



MAX PLANCK INSTITUTE
FOR DYNAMICS OF COMPLEX
TECHNICAL SYSTEMS
MAGDEBURG



COMPUTATIONAL METHODS IN
SYSTEMS AND CONTROL THEORY

Reduced Rank Extrapolation for Low-Rank Matrix Equations

Xiaobo Liu

MPI Magdeburg, Germany

**Numerical Analysis and Optimization workshop
BIT, Beijing, September 20, 2026**

Based on joint work P. Benner et al. (arXiv:2603.12979) and P. den Boef et al. (Numer. Algorithms, 2026).



1. Reduced Rank Extrapolation

2. Methodological Extensions

3. Algebraic Riccati Equation

4. Multi-Term Sylvester Equation

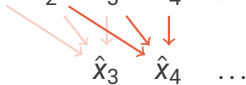
5. Conclusions



- **Non-cycling** mode generates new extrapolated sequence $\hat{x}_w, \hat{x}_{w+1}, \dots$ (window size w).

Original sequence: $x_1 \quad x_2 \quad x_3 \quad x_4 \quad \dots$

Improved sequence:



Non-Cycling Mode

```
for  $i \leftarrow 1, 2, \dots$  do
   $x_{i+1} \leftarrow f(x_i)$ 
  if  $i \geq w$  then
     $\hat{x}_i \leftarrow \text{extr.}(x_{i-w+1}, \dots, x_i, x_{i+1})$ 
    if  $\hat{x}_i$  converged then break
  end
end
```

Cycling Mode

```
for  $i \leftarrow 1, 2, \dots$  do
   $x_{i+1} \leftarrow f(x_i)$ 
  if should start new cycle then
     $x_{i+1} \leftarrow \text{extr.}(x_{i-w+1}, \dots, x_i, x_{i+1})$ 
  end
  if  $x_{i+1}$  converged then break
end
```

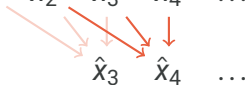
- **Cycling** mode **restarts** using the **extrapolated iterate**, to generate, e.g., $x_1, x_2, \hat{x}_3, x_4, \dots$



- **Non-cycling** mode generates new extrapolated sequence $\hat{x}_w, \hat{x}_{w+1}, \dots$ (window size w).

Original sequence: $x_1 \quad x_2 \quad x_3 \quad x_4 \quad \dots$

Improved sequence:



Non-Cycling Mode

```
for  $i \leftarrow 1, 2, \dots$  do
   $x_{i+1} \leftarrow f(x_i)$ 
  if  $i \geq w$  then
     $\hat{x}_i \leftarrow \text{extr.}(x_{i-w+1}, \dots, x_i, x_{i+1})$ 
    if  $\hat{x}_i$  converged then break
  end
end
```

Cycling Mode

```
for  $i \leftarrow 1, 2, \dots$  do
   $x_{i+1} \leftarrow f(x_i)$ 
  if should start new cycle then
     $x_{i+1} \leftarrow \text{extr.}(x_{i-w+1}, \dots, x_i, x_{i+1})$ 
  end
  if  $x_{i+1}$  converged then break
end
```

- **Cycling** mode **restarts** using the **extrapolated iterate**, to generate, e.g., $x_1, x_2, \hat{x}_3, x_4, \dots$



- Objective: solve $Ax = b$.
- Iterative scheme: splitting $A = I - (I - A)$,

$$x_{i+1} \leftarrow f(x_i) := (I - A)x_i + b.$$

- **RRE**: find $\hat{x} := \sum_{i=1}^w \gamma_i x_i$ that minimizes the **increment** of the *fixed-point iteration*

$$\|f(\hat{x}) - \hat{x}\|,$$

where $\gamma_1 + \dots + \gamma_w = 1$.

- **Fixed-point increment** coincides with **residual** of underlying equation:

$$f(x) - x = b - Ax$$



- Objective: solve $Ax = b$.
- Under a more general splitting $A = M - N$,

$$x_{i+1} \leftarrow f(x_i) := M^{-1}(Nx_i + b).$$

- **RRE**: find $\hat{x} := \sum_{i=1}^w \gamma_i x_i$ that minimizes the increment of the *fixed-point iteration*

$$\begin{aligned} \|f(\hat{x}) - \hat{x}\| &= \left\| \sum_{i=1}^w \gamma_i (f(x_i) - x_i) \right\| \text{ by linearity} \\ &= \left\| \sum_{i=1}^w \gamma_i (x_{i+1} - x_i) \right\|, \end{aligned}$$

where $\gamma_1 + \dots + \gamma_w = 1$.

- Two residual notions now: fixed-point increment vs residual of underlying equation

$$f(x) - x = M^{-1}(b - Ax)$$



Most derivations of RRE read off the iteration scheme from an underlying equation where **both notions of the residual coincide**:

$$x = Ax + b \quad [\text{Mešina, '77}], [\text{Eddy, '79}] (\text{not spelled out}), [\text{Sidi, '88}], [\text{Sidi, '91}]$$

$$Ax = b \quad [\text{Sidi \& Shapira, '98}]$$

or only consider the **fixed-point increment**:

$$Ax = b \quad [\text{Kaniel \& Stein, '74}]$$

$$x = f(x) \quad [\text{Sidi, '20}]$$



1. Reduced Rank Extrapolation

2. Methodological Extensions

3. Algebraic Riccati Equation

4. Multi-Term Sylvester Equation

5. Conclusions



Consider **nonstationary** splittings $A = M_i - N_i$ for solving $Ax = b$:

$$x_{i+1} \leftarrow f_i(x_i) := M_i^{-1}(N_i x_i + b).$$

Two different formulations for the weight γ in the RRE extrapolant $\hat{x} := \sum_{i=1}^w \gamma_i x_i$:

Increment-Based RRE**(classic)**

$$\gamma = \arg \min_{g \in \mathbb{R}^w} \left\| \sum_{i=1}^w \eta_i (x_{i+1} - x_i) \right\|,$$
$$\text{s.t. } \sum_{i=1}^w \eta_i = 1.$$

Residual-Based RRE**(new)**

$$\gamma = \arg \min_{\eta \in \mathbb{R}^w} \left\| \sum_{i=1}^w \eta_i (b - Ax_i) \right\|,$$
$$\text{s.t. } \sum_{i=1}^w \eta_i = 1.$$

Target

Extend both RRE formulations to **nonstationary** f_i with **low-rank matrix sequence** $\{X_i\}$.

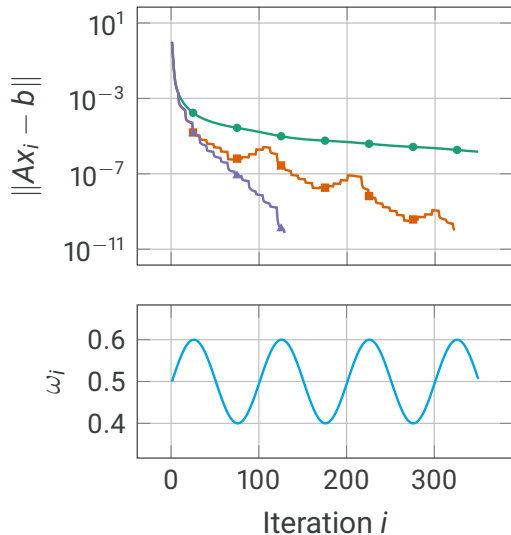
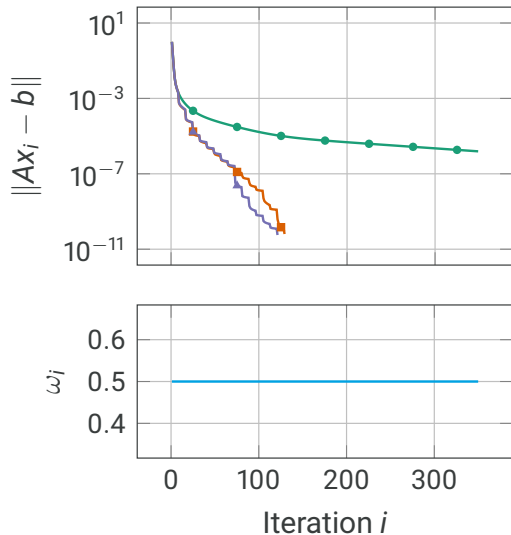


Figure: Comparison of increment-based (—■—) and residual-based (—▲—) RRE formulations applied to **nonstationary successive over-relaxation** schemes (—●—) for solving $Ax = b$.



- RRE for vector sequences $x_i \in \mathbb{R}^d$: via QR & method of Lagrange multipliers

$$\gamma = \arg \min_{\eta \in \mathbb{R}^w} \left\| \sum_{i=1}^w \eta_i (b - Ax_i) \right\| \iff U^T U \alpha = \mathbf{1} \in \mathbb{R}^w \text{ and } \gamma := \alpha / \|\alpha\|$$

$$\text{s.t. } \sum_{i=1}^w \eta_i = 1 \quad U := [b - Ax_1 \mid \dots \mid b - Ax_w] \in \mathbb{R}^{d \times w}$$

- RRE for low-rank *matrix* sequences $X_i = \begin{bmatrix} \square & \square \end{bmatrix}$ with residuals $\mathcal{R}(X_i) = \begin{bmatrix} \square & \square \end{bmatrix}$:

- Assembling $X_i \in \mathbb{R}^{d \times d}$ is **prohibitively expensive**, despite inner factor having size r
- Applying $\text{vec}(\cdot)$ to X_i , $U \in \mathbb{R}^{d^2 \times w}$ is **even more expensive** to assemble



- RRE for vector sequences $x_i \in \mathbb{R}^d$: via QR & method of Lagrange multipliers

$$\gamma = \arg \min_{\eta \in \mathbb{R}^w} \left\| \sum_{i=1}^w \eta_i (b - Ax_i) \right\| \iff U^T U \alpha = 1 \in \mathbb{R}^w \text{ and } \gamma := \alpha / \|\alpha\|$$

$$\text{s.t. } \sum_{i=1}^w \eta_i = 1 \quad U := \begin{bmatrix} b - Ax_1 & \dots & b - Ax_w \end{bmatrix} \in \mathbb{R}^{d \times w}$$

- RRE for low-rank *matrix* sequences $X_i = \begin{bmatrix} \square & \square \end{bmatrix}$ with residuals $\mathcal{R}(X_i) = \begin{bmatrix} \square & \square \end{bmatrix}$:

- Assembling $X_i \in \mathbb{R}^{d \times d}$ is **prohibitively expensive**, despite inner factor having size r
- Applying $\text{vec}(\cdot)$ to X_i , $U \in \mathbb{R}^{d^2 \times w}$ is **even more expensive** to assemble



- RRE for vector sequences $x_i \in \mathbb{R}^d$: via QR & method of Lagrange multipliers

$$\gamma = \arg \min_{\eta \in \mathbb{R}^w} \left\| \sum_{i=1}^w \eta_i (b - Ax_i) \right\| \iff U^T U \alpha = 1 \in \mathbb{R}^w \text{ and } \gamma := \alpha / \|\alpha\|$$

$$\text{s.t. } \sum_{i=1}^w \eta_i = 1 \quad U := \begin{bmatrix} b - Ax_1 & \dots & b - Ax_w \end{bmatrix} \in \mathbb{R}^{d \times w}$$

- RRE for low-rank *matrix* sequences $X_i = \begin{bmatrix} \square & \square \end{bmatrix}$ with residuals $\mathcal{R}(X_i) = \begin{bmatrix} \square & \square \end{bmatrix}$:

- Assembling $X_i \in \mathbb{R}^{d \times d}$ is **prohibitively expensive**, despite inner factor having size r
- Applying $\text{vec}(\cdot)$ to X_i , $U \in \mathbb{R}^{d^2 \times w}$ is **even more expensive** to assemble



Main idea:

$$\gamma = \arg \min_{\eta} \left\| \eta_1 \begin{bmatrix} \text{green} \end{bmatrix} + \eta_2 \begin{bmatrix} \text{orange} \end{bmatrix} + \eta_3 \begin{bmatrix} \text{purple} \end{bmatrix} \right\|$$

1. Move the arithmetic onto inner factors of dimension $r \ll d$
2. Reduce problem dimension via QR decomposition of outer factors (tall-and-skinny)

Benefits:

Solution of $U^T U \alpha = 1 \in \mathbb{R}^w$ now requires $U \in \mathbb{R}^{(wr)^2 \times w}$ where $(wr)^2 \ll d^2$.

Recall that $\gamma := \alpha / \|\alpha\|$, and that // may represent increments or residuals.



Main idea:

$$\gamma = \arg \min_{\eta} \left\| \begin{bmatrix} \text{green} & \text{orange} & \text{purple} \end{bmatrix} \begin{bmatrix} \eta_1 \text{ green} & & \\ & \eta_2 \text{ orange} & \\ & & \eta_3 \text{ purple} \end{bmatrix} \begin{bmatrix} \text{green} \\ \text{orange} \\ \text{purple} \end{bmatrix}^T \right\|$$

1. Move the arithmetic onto inner factors of dimension $r \ll d$
2. Reduce problem dimension via QR decomposition of outer factors (tall-and-skinny)

Benefits:

Solution of $U^T U \alpha = 1 \in \mathbb{R}^w$ now requires $U \in \mathbb{R}^{(wr)^2 \times w}$ where $(wr)^2 \ll d^2$.

Recall that $\gamma := \alpha / \|\alpha\|$, and that // may represent increments or residuals.



Main idea:

$$\gamma = \arg \min_{\eta} \left\| \left[\begin{array}{c} Q \\ \begin{array}{|c|} \hline \text{diagonal} \end{array} \end{array} \right] \left[\begin{array}{c} \eta_1 \text{ (green)} \\ \eta_2 \text{ (orange)} \\ \eta_3 \text{ (purple)} \end{array} \right] \left[\begin{array}{c} \begin{array}{|c|} \hline \text{diagonal} \end{array} \\ Q^T \end{array} \right] \right\|$$

1. Move the arithmetic onto inner factors of dimension $r \ll d$
2. Reduce problem dimension via QR decomposition of outer factors (tall-and-skinny)

Benefits:

Solution of $U^T U \alpha = 1 \in \mathbb{R}^w$ now requires $U \in \mathbb{R}^{(wr)^2 \times w}$ where $(wr)^2 \ll d^2$.

Recall that $\gamma := \alpha / \|\alpha\|$, and that // may represent increments or residuals.



Main idea:

$$\gamma = \arg \min_{\eta} \left\| \begin{bmatrix} \text{Green} & \text{Orange} & \text{Purple} \\ \text{Green} & \text{Orange} & \text{Purple} \\ \text{Green} & \text{Orange} & \text{Purple} \end{bmatrix} \begin{bmatrix} \eta_1 \text{ Green} & & \\ & \eta_2 \text{ Orange} & \\ & & \eta_3 \text{ Purple} \end{bmatrix} \begin{bmatrix} \text{Green} & \text{Orange} & \text{Purple} \\ \text{Green} & \text{Orange} & \text{Purple} \\ \text{Green} & \text{Orange} & \text{Purple} \end{bmatrix}^T \right\|$$

1. Move the arithmetic onto inner factors of dimension $r \ll d$
2. Reduce problem dimension via QR decomposition of outer factors (tall-and-skinny)

Benefits:

Solution of $U^T U \alpha = 1 \in \mathbb{R}^w$ now requires $U \in \mathbb{R}^{(wr)^2 \times w}$ where $(wr)^2 \ll d^2$.

Recall that $\gamma := \alpha / \|\alpha\|$, and that // may represent increments or residuals.



Main idea:

$$\gamma = \arg \min_{\eta} \left\| \eta_1 \begin{array}{c} \text{green rectangle} \\ \text{green square} \\ \text{green rectangle}^T \end{array} + \eta_2 \begin{array}{c} \text{orange rectangle} \\ \text{orange square} \\ \text{orange rectangle}^T \end{array} + \eta_3 \begin{array}{c} \text{purple rectangle} \\ \text{purple square} \\ \text{purple rectangle}^T \end{array} \right\|$$

1. Move the arithmetic onto inner factors of dimension $r \ll d$
2. Reduce problem dimension via QR decomposition of outer factors (tall-and-skinny)

Benefits:

Solution of $U^T U \alpha = 1 \in \mathbb{R}^w$ now requires $U \in \mathbb{R}^{(wr)^2 \times w}$ where $(wr)^2 \ll d^2$.

Recall that $\gamma := \alpha / \|\alpha\|$, and that // may represent increments or residuals.



Main idea:

$$\gamma = \arg \min_{\eta} \left\| \eta_1 \text{vec} \left(\begin{array}{|c|} \hline \text{green triangle} \\ \hline \end{array} \begin{array}{|c|} \hline \text{green square} \\ \hline \end{array} \begin{array}{|c|} \hline \text{green triangle}^T \\ \hline \end{array} \right) + \eta_2 \text{vec} \left(\begin{array}{|c|} \hline \text{orange triangle} \\ \hline \end{array} \begin{array}{|c|} \hline \text{orange square} \\ \hline \end{array} \begin{array}{|c|} \hline \text{orange triangle}^T \\ \hline \end{array} \right) + \eta_3 \text{vec} \left(\begin{array}{|c|} \hline \text{purple triangle} \\ \hline \end{array} \begin{array}{|c|} \hline \text{purple square} \\ \hline \end{array} \begin{array}{|c|} \hline \text{purple triangle}^T \\ \hline \end{array} \right) \right\|$$

1. Move the arithmetic onto inner factors of dimension $r \ll d$
2. Reduce problem dimension via QR decomposition of outer factors (tall-and-skinny)

Benefits:

Solution of $U^T U \alpha = 1 \in \mathbb{R}^w$ now requires $U \in \mathbb{R}^{(wr)^2 \times w}$ where $(wr)^2 \ll d^2$.

Recall that $\gamma := \alpha / \|\alpha\|$, and that // may represent **increments** or **residuals**.

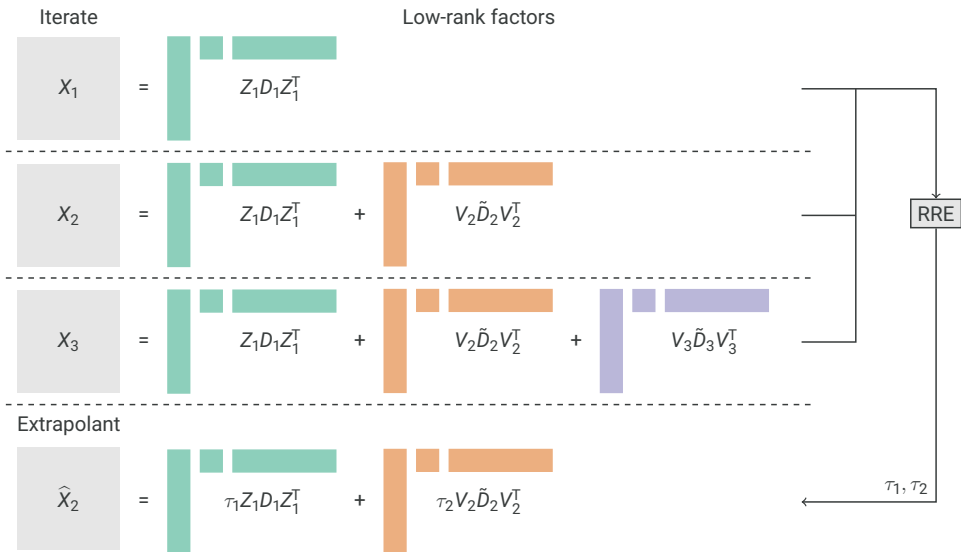


Figure: Increment-based RRE for low-rank matrix sequences, where $\tau_i := \gamma_i + \dots + \gamma_n$.

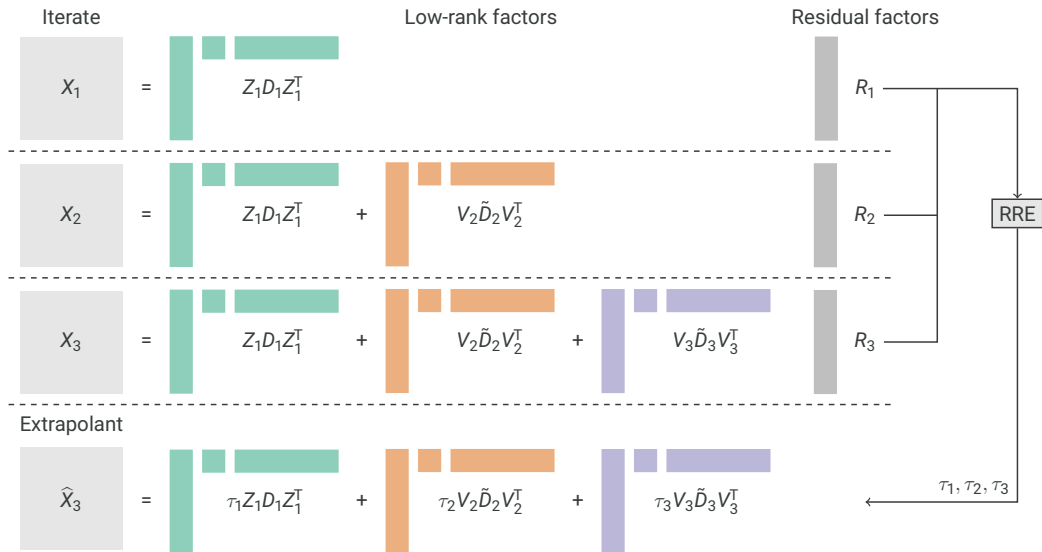


Figure: Residual-based RRE for low-rank matrix sequences, where $\tau_i := \gamma_i + \dots + \gamma_n$.



1. Reduced Rank Extrapolation
2. Methodological Extensions
- 3. Algebraic Riccati Equation**
4. Multi-Term Sylvester Equation
5. Conclusions



- Underlying equation: algebraic Riccati equation (ARE)

$$\begin{bmatrix} C^T \\ C \end{bmatrix} + A^T X E + E^T X A - E^T X \begin{bmatrix} B \\ H^{-1} B^T \end{bmatrix} X E = \begin{bmatrix} 0 \end{bmatrix} \in \mathbb{R}^{d \times d}$$

- $X \approx \begin{bmatrix} Z \\ D \end{bmatrix} \begin{bmatrix} Z^T \end{bmatrix}$ and $\mathcal{R}(X) \approx \begin{bmatrix} R \\ T \end{bmatrix} \begin{bmatrix} R^T \end{bmatrix}$ [Benner & Bujanović, '16]

- Appears in various problems in control theory [Jungers, '17]
- Implementation derived from M-M.E.S.S (The Matrix Equation Sparse Solvers library) [Saak, Köhler, & Benner, '25]



Riccati Alternating Directions Implicit (RADI) naturally combines the two extensions: nonstationary iteration and low-rank matrix iterates [Benner et al., '18]:

One RADI Step

Given

$$X_i = Z_i D_i Z_i^T, \quad \mathcal{R}(X_i) = R_i T R_i^T,$$

select a **shift** $\sigma_i \in \mathbb{R}_{<0}$ and compute

$$V_{i+1} \leftarrow \sqrt{-2\sigma_i} (A^T - E^T X_i B H^{-1} B^T + \sigma_i E^T)^{-1} R_i T$$

$$\tilde{Y}_{i+1} \leftarrow T - \frac{1}{2\sigma_i} (V_{i+1}^T B) H^{-1} (V_{i+1}^T B)^T \quad \text{and} \quad \tilde{D}_{i+1} \leftarrow (\tilde{Y}_{i+1})^{-1}$$

Assemble low-rank factors of X_{i+1} and outer residual factor:

$$Z_{i+1} \leftarrow [Z_i \quad V_{i+1}] \quad \text{and} \quad D_{i+1} \leftarrow \begin{bmatrix} D_i & \\ & \tilde{D}_{i+1} \end{bmatrix}$$

$$R_{i+1} \leftarrow R_i + \sqrt{-2\sigma_i} E^T V_{i+1} \tilde{D}_{i+1}.$$

- $X_1 = 0$ and (A, E) stable $\rightsquigarrow$
 $R_1 = C^T$, $T = I$, and $\text{rank}(\mathcal{R}(X_i)) \leq \text{rank}(C)$.

- Fixed-point process of the form

$$X_{i+1} = F_i(X_i),$$

where the map changes with the **shift** σ_i .

- X_i and $\mathcal{R}(X_i)$ remain in low-rank factored form: **increments of the low-rank factors** are computed.



Residual-based RRE:

$$\gamma = \arg \min_{\sum_i \eta_i = 1} \|\eta_i \mathcal{R}(X_i)\| = \arg \min_{\sum_i \eta_i = 1} \left\| \sum_{i=1}^w \eta_i R_i T R_i^T \right\| = \arg \min_{\sum_i \eta_i = 1} \left\| \sum_{i=1}^w \eta_i \tilde{R}_i \tilde{T} \tilde{R}_i^T \right\| = \arg \min_{\sum_i \eta_i = 1} \left\| \sum_{i=1}^w \eta_i \text{vec}(\tilde{R}_i \tilde{T} \tilde{R}_i^T) \right\|$$

with one thin QR decomposition

$$[R_1 \cdots R_w] = \tilde{Q}[\tilde{R}_1 \cdots \tilde{R}_w].$$

- The residual factors R_i, T are already computed by RADI.
- The QR factorization of $[R_1 \cdots R_w]$ dominates the extra cost of RRE ($\mathcal{O}(d(wr)^2)$ flops).
- Residual-based RRE needs w iterates rather than $w + 1$, not relying on increments $X_{i+1} - X_i$.



The **RRE extrapolant** can be assembled from RADI rank updates:

$$\hat{X} = \sum_{i=1}^w \gamma_i X_i = \sum_{i=1}^w \tau_i V_i \tilde{D}_i V_i^T, \quad \tau_i = \sum_{j=i}^w \gamma_j.$$

For AREs, the target is the stabilizing *positive semidefinite* solution. Thus a simple sufficient condition for $\hat{X} \succeq 0$ is

$$\tau_i \geq 0, \quad i = 1, \dots, w.$$

- For **cycling-mode** RRE, the extrapolates $\hat{X}$ used as new initializations: computing the weights γ requires residual factorization $\mathcal{R}(\hat{X}) = \hat{R}T\hat{R}^T$.



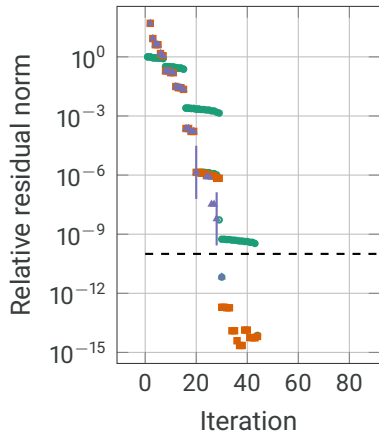
The **RRE extrapolant** can be assembled from RADI rank updates:

$$\hat{X} = \sum_{i=1}^w \gamma_i X_i = \sum_{i=1}^w \tau_i V_i \tilde{D}_i V_i^T, \quad \tau_i = \sum_{j=i}^w \gamma_j.$$

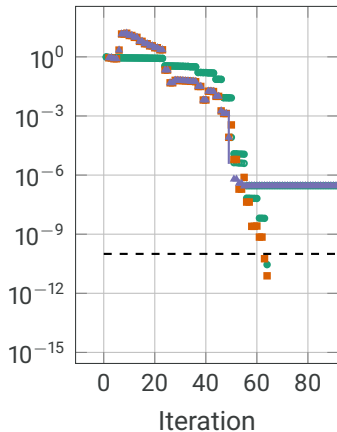
For AREs, the target is the stabilizing *positive semidefinite* solution. Thus a simple sufficient condition for $\hat{X} \succeq 0$ is

$$\tau_i \geq 0, \quad i = 1, \dots, w.$$

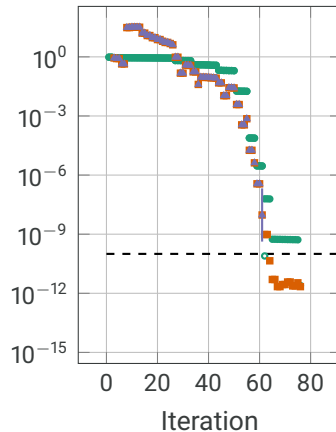
- For **cycling-mode** RRE, the extrapolates $\hat{X}$ used as new initializations: computing the weights γ requires residual factorization $\mathcal{R}(\hat{X}) = \hat{R}T\hat{R}^T$.



(a) $q = 1$.



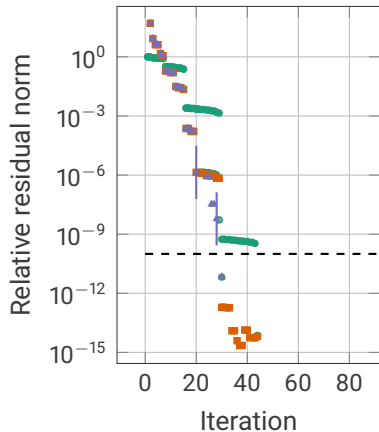
(b) $q = 20$.



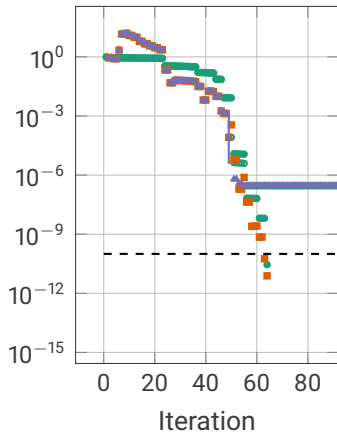
(c) $q = 40$.

Figure: RADl ($\bullet$), non-cycling RRE ($\blacksquare$), and cycling RRE ($\blacktriangle$).

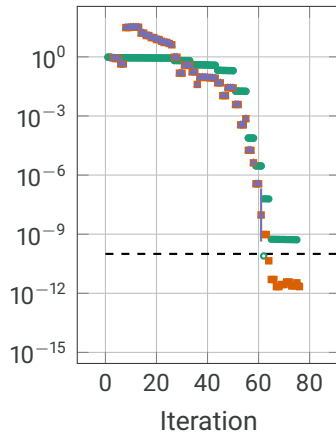
- Stagnation of cycling RRE likely due to non-PSD extrapolant (as initializer) residual.



(a) $q = 1$.



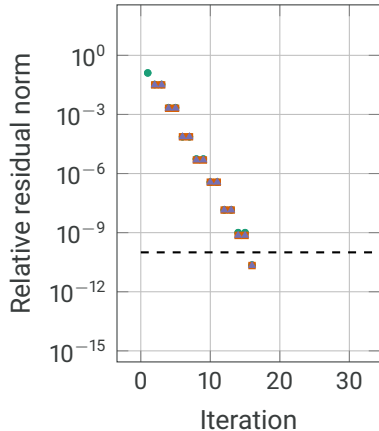
(b) $q = 20$.



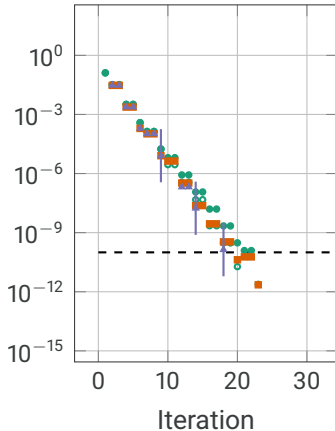
(c) $q = 40$.

Figure: RADl (●○), non-cycling RRE (■), and cycling RRE (▲).

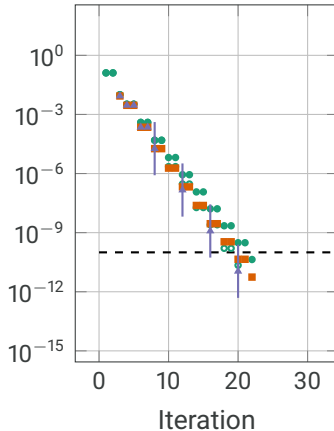
- Stagnation of cycling RRE likely due to non-PSD extrapolant (as initializer) residual.



(a) $q = 1$.



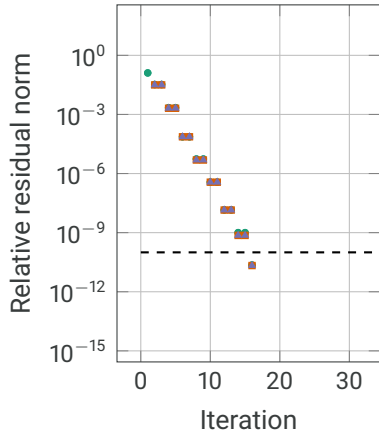
(b) $q = 20$.



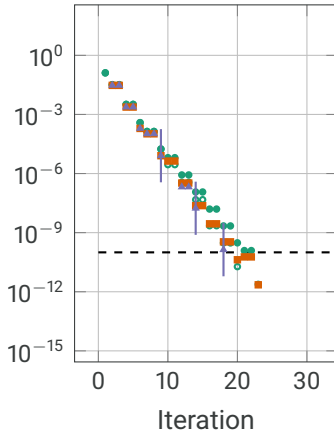
(c) $q = 40$.

Figure: ADI (●○), non-cycling RRE (■), and cycling RRE (▲).

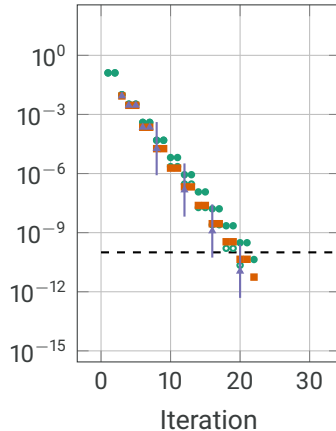
- No stagnation in the Lyapunov case & more benefits of RRE with larger output q .



(a) $q = 1$.



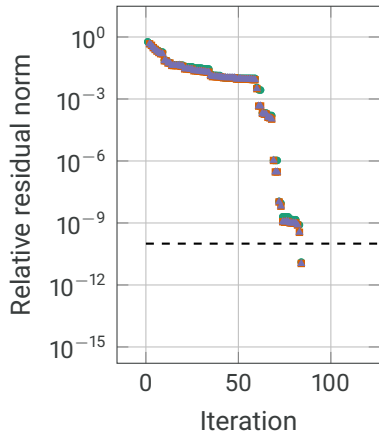
(b) $q = 20$.



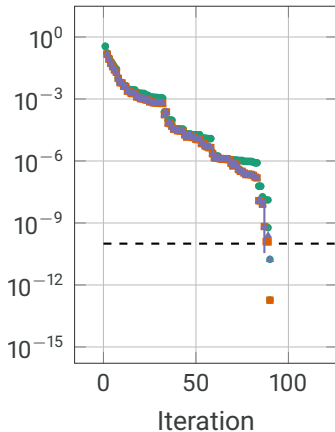
(c) $q = 40$.

Figure: ADI ($\bullet\circ$), non-cycling RRE ($\blacksquare$), and cycling RRE ($\blacktriangle$).

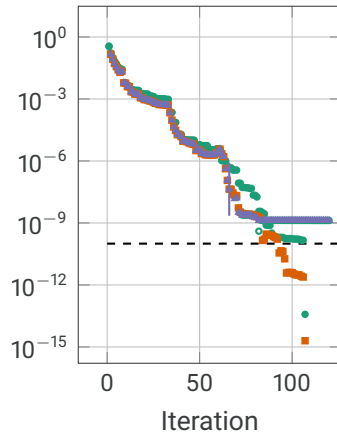
- No stagnation in the Lyapunov case & more benefits of RRE with larger output q .



(a) ADI for Controllability ALE



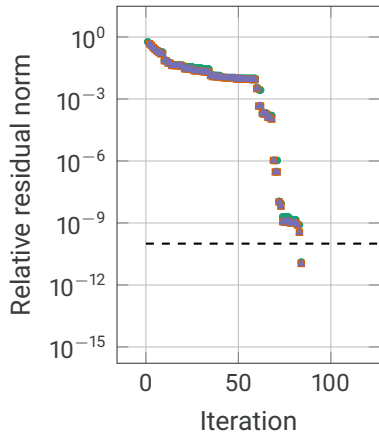
(b) ADI for Observability ALE



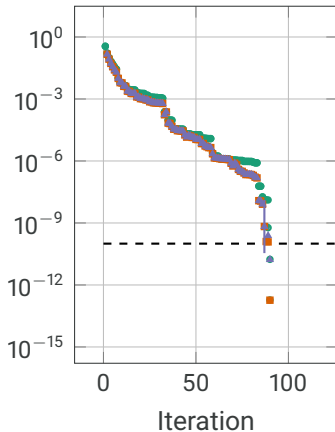
(c) RADI for ARE

Figure: (R)ADI ($\bullet$), non-cycling RRE ($\blacksquare$), and cycling RRE ($\blacktriangle$).

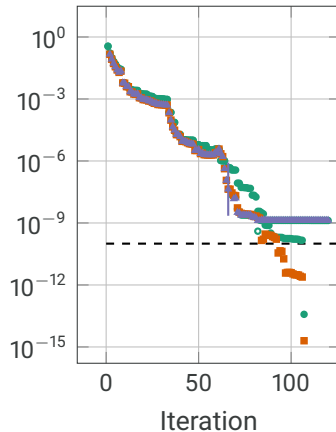
- Little acceleration for ALEs (ADI iterates near optimal?); (non-cycling) benefits for ARE.



(a) ADI for Controllability ALE



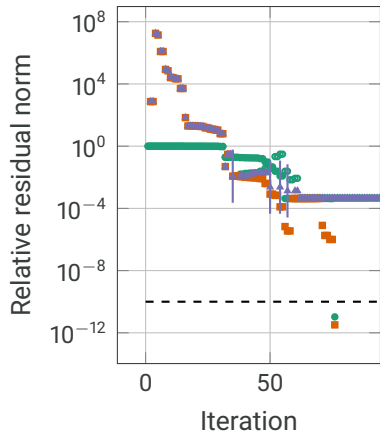
(b) ADI for Observability ALE



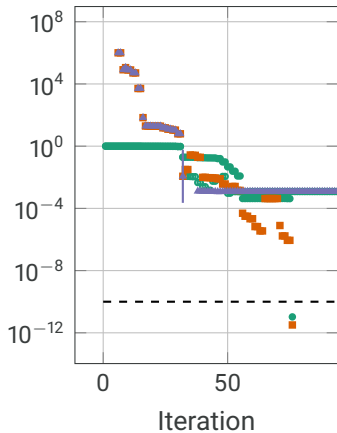
(c) RADI for ARE

Figure: (R)ADI ($\bullet$), non-cycling RRE ($\blacksquare$), and cycling RRE ($\blacktriangle$).

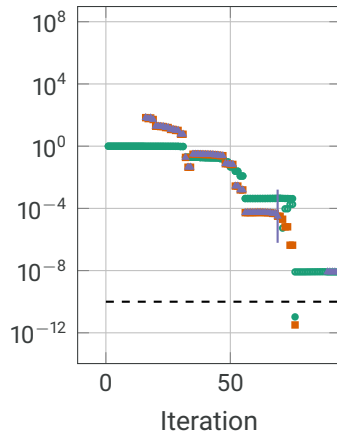
- Little acceleration for ALEs (ADI iterates near optimal?); (non-cycling) benefits for ARE.



(a) $w = 3$.



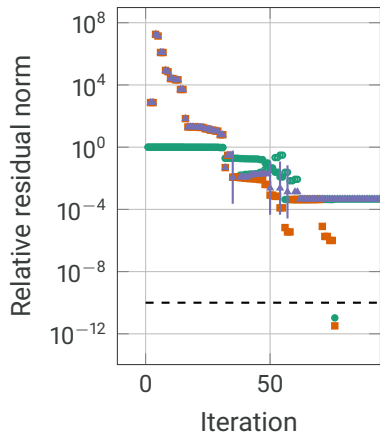
(b) $w = 6$.



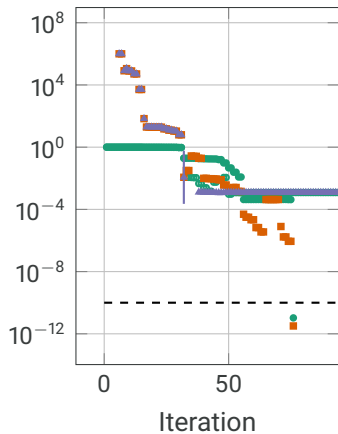
(c) $w = 12$.

Figure: RADl (●○), non-cycling RRE (■), and cycling RRE (▲).

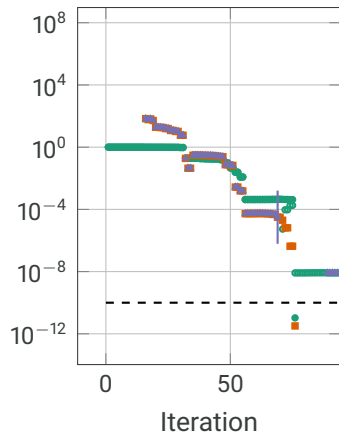
- Stagnation in cycling RRE. Faster convergence in non-cycling RRE for larger w .



(a) $w = 3$.



(b) $w = 6$.



(c) $w = 12$.

Figure: RADl (●○), non-cycling RRE (■), and cycling RRE (▲).

- Stagnation in cycling RRE. Faster convergence in non-cycling RRE for larger w .



1. Reduced Rank Extrapolation
2. Methodological Extensions
3. Algebraic Riccati Equation
- 4. Multi-Term Sylvester Equation**
5. Conclusions



- Underlying equation: multi-term Sylvester equation (MSE)

$$\mathcal{A}(X) = \mathcal{L}(X) + \Pi(X) = -Y, \quad X, Y \in \mathbb{R}^{n \times m}$$

with the **Sylvester** part and additional **low-rank** terms

$$\mathcal{L}(X) = AX + XB, \quad \Pi(X) = \sum_{j=1}^{\ell} N_j X H_j.$$

- If $\rho(\mathcal{L}^{-1}\Pi) < 1$, the splitting yields a fixed-point process

$$X_{k+1} = \mathcal{L}^{-1}(-Y - \Pi(X_k)) \iff AX_{k+1} + X_{k+1}B = -Y - \Pi(X_k).$$

- Large-scale setting (now two low-rank bases are required)

$$Y = \begin{bmatrix} F \\ T \end{bmatrix} G^T, \quad X_k \approx \begin{bmatrix} Z_{L,k} \\ D_k \end{bmatrix} \begin{bmatrix} Z_{R,k}^T \end{bmatrix}.$$

- The symmetric case recovers **Lyapunov-plus-positive** equations; key applications in **control** and **model reduction** [Benner & Damm, '11], [Kleinman, '69].



One Outer Step

Choose tolerances $\tau_{\text{outer}}, \tau_{k,\text{inner}}, \tau_{\text{trunc}} > 0$ and solve

$AX_i + X_iB = -F_iT_iG_i^T$ approxly. to $\tau_{k,\text{inner}}$ for $Z_{L,i}D_iZ_{R,i}^T$,

$$\tilde{X}_i = Z_{L,i}D_iZ_{R,i}^T \leftarrow \mathcal{T}_{\text{runc}}(Z_{L,i}D_iZ_{R,i}^T, \tau_{\text{trunc}}).$$

If a RRE cycle completes,

$$\hat{X}_i \leftarrow \text{RRE}_w(\tilde{X}_i, \tilde{X}_{i-1}, \dots, \tilde{X}_{i-w}),$$

$$X_i = Z_{L,i}D_iZ_{R,i}^T \leftarrow \mathcal{T}_{\text{runc}}(\hat{X}_i, \tau_{\text{trunc}}).$$

Estimate $\|\mathcal{R}(X_i)\|$ and stop if $\|\mathcal{R}(X_i)\| < \tau_{\text{outer}}\|FTG^T\|$.

$$F_iT_iG_i^T = \mathcal{T}_{\text{runc}}\left(FTG^T + \sum_{j=1}^{\ell} N_jZ_{L,i}D_iZ_{R,i}^TH_j, \tau_{\text{trunc}}\right).$$

- Low-rank truncation $\mathcal{T}_{\text{runc}}$ applied to the Sylvester solve, RRE extrapolant, and next low-rank right-hand side.
- Either low-rank ADI or extended Krylov subspace methods as the inner low-rank Sylvester solver, provided by M-M.E.S.S. [Saak, Köhler, & Benner, '25].
- Estimate norm of the residual $\mathcal{R}(X_i) := \mathcal{A}(X) + Y$ by SVD via matrix-free iterative methods.
- Choose $\tau_{k,\text{inner}}$ dynamically s.t. the inner residual is proportional to the outer residual [Shank, Simoncini, & Szyld, '15]:

$$\|AX_i + X_iB + F_iT_iG_i^T\| \leq \delta\|\mathcal{R}(X_i)\|, \delta = 10^{-3}.$$



One Outer Step

Choose tolerances $\tau_{\text{outer}}, \tau_{k,\text{inner}}, \tau_{\text{trunc}} > 0$ and solve

$AX_i + X_iB = -F_iT_iG_i^T$ approxly. to $\tau_{k,\text{inner}}$ for $Z_{L,i}D_iZ_{R,i}^T$,

$$\tilde{X}_i = Z_{L,i}D_iZ_{R,i}^T \leftarrow \mathcal{T}_{\text{trunc}}(Z_{L,i}D_iZ_{R,i}^T, \tau_{\text{trunc}}).$$

If a RRE cycle completes,

$$\hat{X}_i \leftarrow \text{RRE}_w(\tilde{X}_i, \tilde{X}_{i-1}, \dots, \tilde{X}_{i-w}),$$

$$X_i = Z_{L,i}D_iZ_{R,i}^T \leftarrow \mathcal{T}_{\text{trunc}}(\hat{X}_i, \tau_{\text{trunc}}).$$

Estimate $\|\mathcal{R}(X_i)\|$ and stop if $\|\mathcal{R}(X_i)\| < \tau_{\text{outer}}\|FTG^T\|$.

$$F_iT_iG_i^T = \mathcal{T}_{\text{trunc}}\left(FTG^T + \sum_{j=1}^{\ell} N_jZ_{L,i}D_iZ_{R,i}^TH_j, \tau_{\text{trunc}}\right).$$

- **Low-rank truncation** $\mathcal{T}_{\text{trunc}}$ applied to the Sylvester solve, RRE extrapolant, and next low-rank right-hand side.
- Either **low-rank ADI** or **extended Krylov subspace methods** as the inner low-rank Sylvester solver, provided by M-M.E.S.S. [Saak, Köhler, & Benner, '25].
- Estimate norm of the residual $\mathcal{R}(X_i) := \mathcal{A}(X) + Y$ by SVD via matrix-free iterative methods.
- Choose $\tau_{k,\text{inner}}$ *dynamically* s.t. the inner residual is *proportional* to the outer residual [Shank, Simoncini, & Szyld, '15]:

$$\|AX_i + X_iB + F_iT_iG_i^T\| \leq \delta\|\mathcal{R}(X_i)\|, \quad \delta = 10^{-3}.$$



For low-rank iterates

$$\tilde{X}_i = Z_{L,i} D_i Z_{R,i}^T, \quad z_i = \text{rank}(\tilde{X}_i) \ll \min(n, m),$$

increment-based RRE for MSE:

$$\gamma = \arg \min_{\sum_i \eta_i = 1} \left\| \sum_{i=1}^w \eta_i (\tilde{X}_{i+1} - \tilde{X}_i) \right\| = \arg \min_{\sum_i \eta_i = 1} \left\| \sum_{i=1}^w \eta_i \left(R_{L,i+1} D_{i+1} R_{R,i+1}^T - R_{L,i} D_i R_{R,i}^T \right) \right\|$$

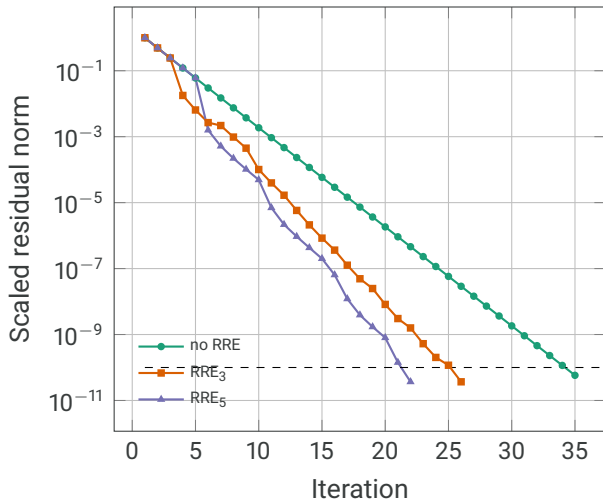
with two thin QR decompositions

$$[Z_{L,1} \cdots Z_{L,w+1}] = Q_L [R_{L,1} \cdots R_{L,w+1}], \quad [Z_{R,1} \cdots Z_{R,w+1}] = Q_R [R_{R,1} \cdots R_{R,w+1}].$$

■ $\text{rank}(\tilde{X}_i) \leq \sum_i z_i \rightsquigarrow$ rank truncation $\mathcal{T}_{\text{runc}}$ right after extrapolation.



Lyapunov example from [Shank, Simoncini, & Szyld, '15]: $n = m = 22500$, $\tau_{\text{outer}} = 10^{-10}$.



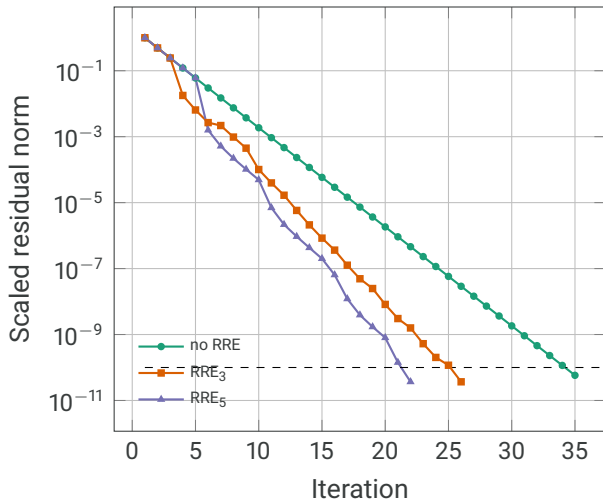
Summary at termination

setting	inner	iter	rank(X)	res	time [s]
no RRE	ADI	34	265	5.8×10^{-11}	90.9
RRE ₃	ADI	25	265	3.7×10^{-11}	72.5
RRE ₅	ADI	21	261	3.7×10^{-11}	56.1
no RRE	EKSM	34	265	5.7×10^{-11}	160.7
RRE ₃	EKSM	25	267	3.0×10^{-11}	133.2
RRE ₅	EKSM	20	247	1.8×10^{-11}	98.6

- RRE speeds up the iteration and reduces the iter steps
- EKSM as inner solver not change iters, but increases total runtime
- Total runtime proportions: RRE $< 1\%$, $\tau_{\text{runc}} \lesssim 15\%$



Lyapunov example from [Shank, Simoncini, & Szyld, '15]: $n = m = 22500$, $\tau_{\text{outer}} = 10^{-10}$.



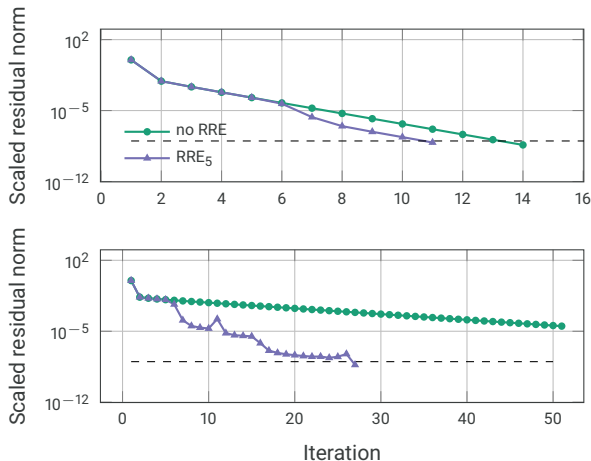
Summary at termination

setting	inner	iter	rank(X)	res	time [s]
no RRE	ADI	34	265	5.8×10^{-11}	90.9
RRE ₃	ADI	25	265	3.7×10^{-11}	72.5
RRE ₅	ADI	21	261	3.7×10^{-11}	56.1
no RRE	EKSM	34	265	5.7×10^{-11}	160.7
RRE ₃	EKSM	25	267	3.0×10^{-11}	133.2
RRE ₅	EKSM	20	247	1.8×10^{-11}	98.6

- RRE speeds up the iteration and reduces the iter steps
- **EKSM** as inner solver not change iters, but increases total runtime
- Total runtime proportions: RRE $< 1\%$, $\mathcal{T}_{\text{runc}} \lesssim 15\%$



Lyapunov example constructed from [Bai & Skoogh, '06]: $n = m = 22650$, $\tau_{\text{outer}} = 10^{-8}$.



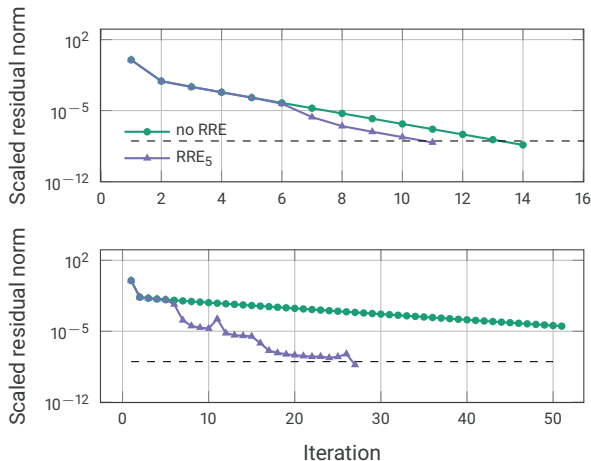
Summary at termination

setting	β	iter	rank(X)	res	time [s]
no RRE	0.50	13	132	4.1×10^{-9}	27.0
RRE ₃	0.50	11	123	8.3×10^{-9}	22.9
RRE ₅	0.50	10	115	7.1×10^{-9}	19.0
no RRE	0.85	50	66	3.2×10^{-5}	89.7
RRE ₃	0.85	47	139	8.9×10^{-9}	103.9
RRE ₅	0.85	26	122	4.9×10^{-9}	60.9

- Only LR-ADI is used as inner solver
- Top: $\beta = 0.5$; bottom: $\beta = 0.85$:
larger $\beta \rightsquigarrow$ larger spectral radius
- RRE accelerates convergence, more
beneficial for the harder problem
 $\beta = 0.85$



Lyapunov example constructed from [Bai & Skoogh, '06]: $n = m = 22650$, $\tau_{\text{outer}} = 10^{-8}$.



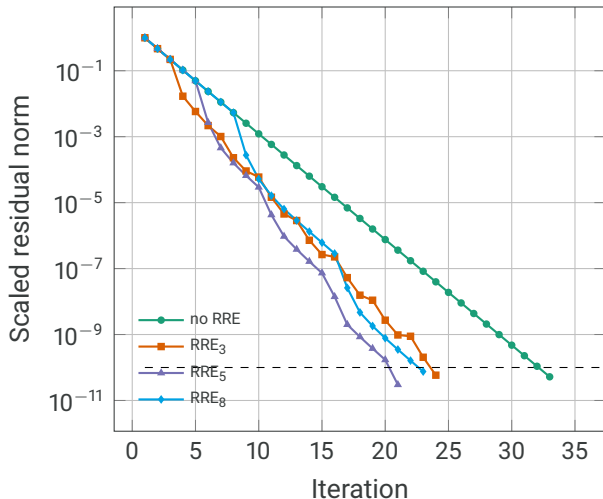
Summary at termination

setting	β	iter	rank(X)	res	time [s]
no RRE	0.50	13	132	4.1×10^{-9}	27.0
RRE ₃	0.50	11	123	8.3×10^{-9}	22.9
RRE ₅	0.50	10	115	7.1×10^{-9}	19.0
no RRE	0.85	50	66	3.2×10^{-5}	89.7
RRE ₃	0.85	47	139	8.9×10^{-9}	103.9
RRE ₅	0.85	26	122	4.9×10^{-9}	60.9

- Only **LR-ADI** is used as inner solver
- Top: $\beta = 0.5$; bottom: $\beta = 0.85$:
larger $\beta \rightsquigarrow$ larger spectral radius
- RRE accelerates convergence, more beneficial for the harder problem
 $\beta = 0.85$



Adapted from [Shank, Simoncini, & Szyld, '15]: $n = 22500$, $m = 8100$, $\tau_{\text{outer}} = 10^{-10}$.



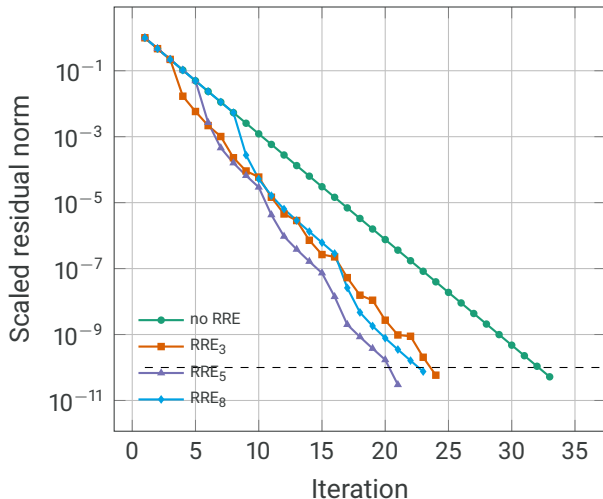
Summary at termination

setting	iter	rank X	res	time [s]
no RRE	32	230	5.2×10^{-11}	96.1
RRE ₃	23	208	5.8×10^{-11}	72.1
RRE ₅	20	208	3.0×10^{-11}	62.9
RRE ₈	22	222	7.5×10^{-11}	66.7

- RRE speeds up the iteration and reduces the iter steps
- Keeping **increasing** w does not further accelerate convergence
- Optimal w requires spectral info of the iteration map



Adapted from [Shank, Simoncini, & Szyld, '15]: $n = 22500$, $m = 8100$, $\tau_{\text{outer}} = 10^{-10}$.



Summary at termination

setting	iter	rank X	res	time [s]
no RRE	32	230	5.2×10^{-11}	96.1
RRE ₃	23	208	5.8×10^{-11}	72.1
RRE ₅	20	208	3.0×10^{-11}	62.9
RRE ₈	22	222	7.5×10^{-11}	66.7

- RRE speeds up the iteration and reduces the iter steps
- Keeping **increasing w** does not further accelerate convergence
- Optimal w requires spectral info of the iteration map



1. Reduced Rank Extrapolation
2. Methodological Extensions
3. Algebraic Riccati Equation
4. Multi-Term Sylvester Equation
- 5. Conclusions**



- Two extensions of RRE, applicable to sequence of low-rank matrices:
 1. Nonstationary fixed-point iterations
 2. Low-rank matrix sequences
- Demonstration of two formulations of RRE:
 - **Increment-based** RRE for multi-term Sylvester equation (conventional for fixed splitting)
 - **Residual-based** RRE for algebraic Riccati equation (shift-independent measure for RADI)
- Experimental validation: RRE can reduce computing time for various problems

Further investigation:

- Non-PSD residuals of the extrapolant of RADI + cycling RRE leading to stagnation
- Analysis of numerical behavior of RRE

Check the manuscripts:

<https://arxiv.org/abs/2603.12979> and <https://doi.org/10.1007/s11075-026-02413-9>

Slides available on <https://xiaobo-liu.github.io/talks>

Thanks for your attention!



- Two extensions of RRE, applicable to sequence of low-rank matrices:
 1. Nonstationary fixed-point iterations
 2. Low-rank matrix sequences
- Demonstration of two formulations of RRE:
 - **Increment-based** RRE for multi-term Sylvester equation (conventional for fixed splitting)
 - **Residual-based** RRE for algebraic Riccati equation (shift-independent measure for RADI)
- Experimental validation: RRE can reduce computing time for various problems

Further investigation:

- Non-PSD residuals of the extrapolant of RADI + cycling RRE leading to stagnation
- Analysis of numerical behavior of RRE

Check the manuscripts:

<https://arxiv.org/abs/2603.12979> and <https://doi.org/10.1007/s11075-026-02413-9>

Slides available on <https://xiaobo-liu.github.io/talks>

Thanks for your attention!



- Two extensions of RRE, applicable to sequence of low-rank matrices:
 1. Nonstationary fixed-point iterations
 2. Low-rank matrix sequences
- Demonstration of two formulations of RRE:
 - **Increment-based** RRE for multi-term Sylvester equation (conventional for fixed splitting)
 - **Residual-based** RRE for algebraic Riccati equation (shift-independent measure for RADI)
- Experimental validation: RRE can reduce computing time for various problems

Further investigation:

- Non-PSD residuals of the extrapolant of RADI + cycling RRE leading to stagnation
- Analysis of numerical behavior of RRE

Check the manuscripts:

<https://arxiv.org/abs/2603.12979> and <https://doi.org/10.1007/s11075-026-02413-9>

Slides available on <https://xiaobo-liu.github.io/talks>

Thanks for your attention!



Zhaojun Bai and Daniel Skoogh.

A projection method for model reduction of bilinear dynamical systems.

Linear Algebra Appl., 415(2–3):406–425, 2006.



Peter Benner and Zvonimir Bujanović.

On the solution of large-scale algebraic riccati equations by using low-dimensional invariant subspaces.

Linear Algebra Appl., 488:430–459, 2016.



Peter Benner, Zvonimir Bujanović, Patrick Kürschner, and Jens Saak.

RADI: a low-rank ADI-type algorithm for large scale algebraic Riccati equations.

Numer. Math., 138:301–330, 2018.



Peter Benner, Zvonimir Bujanović, Patrick Kürschner, and Jens Saak.

A numerical comparison of different solvers for large-scale, continuous-time algebraic Riccati equations and LQR problems.

SIAM J. Sci. Comput., 42(2):A957–A996, 2020.



Peter Benner and Tobias Damm.

Lyapunov equations, energy functionals, and model order reduction of bilinear and stochastic systems.

SIAM J. Control. Optim., 49(2):686–711, 2011.



Peter Benner and Jens Saak.

Linear-quadratic regulator design for optimal cooling of steel profiles.

Technical Report SFB393/05-05, Sonderforschungsbereich 393 *Parallele Numerische Simulation für Physik und Kontinuumsmechanik*, TU Chemnitz, D-09107 Chemnitz (Germany), 2005.



Robert P. Eddy.

Extrapolating to the limit of a vector sequence.

In *Information linkage between applied mathematics and industry*, pages 387–396. Elsevier, 1979.



Marc Jungers.

Historical perspectives of the Riccati equations.

IFAC-PapersOnLine, 50(1):9535–9546, 2017.



S. Kaniel and J. Stein.

Least-square acceleration of iterative methods for linear equations.

J. Opt. Th. Appl., 14:431–437, 1974.



David L. Kleinman.

On the stability of linear stochastic systems.

IEEE Trans. Autom. Control, 14(4):429–430, 1969.



Marián Mešina.

Convergence acceleration for the iterative solution of the equations $X = AX + f$.

Comp. Meth. Appl. Mech. Eng., 10(2):165–173, 1977.



Christian Moosmann, Evgenii B. Rudnyi, Andreas Greiner, and Jan G. Korvink.

Model order reduction for linear convective thermal flow.

In Proceedings of 10th International Workshops on THERMal INvestigations of ICs and Systems, 2004.



Jens Saak and Martin Köhler and Peter Benner.

M-M.E.S.S. – The Matrix Equations Sparse Solvers library.



Stephen D. Shank, Valeria Simoncini and Daniel B. Szyld.

Efficient low-rank solution of generalized Lyapunov equations.

Numer. Math., 134:327–342, 2015.



Avram Sidi.

Extrapolation vs. projection methods for linear systems of equations.

J. Comput. Appl. Math., 22(1):71–88, 1988.



Avram Sidi.

Efficient implementation of minimal polynomial and reduced rank extrapolation methods.

J. Comput. Appl. Math., 36(3):305–337, 1991.



Avram Sidi.

Minimal polynomial and reduced rank extrapolation methods are related.

Adv. Comput. Math., 43:151–170, 2017.



Avram Sidi.

A convergence study for reduced rank extrapolation on nonlinear systems.

Numer. Algorithms, 84:957–982, 2020.



Avram Sidi and Yair Shapira.

Upper bounds for convergence rates of acceleration methods with initial iterations.

Numer. Algorithms, 18(2):113–132, 1998.



Ninoslav Truhar and Krešimir Veselić.

An efficient method for estimating the optimal dampers' viscosity for linear vibrating systems using Lyapunov equation.

SIAM J. Matrix Anal. Appl., 31(1):18–39, 2009.