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

ILAS 2026,
Virginia Tech, Blacksburg, USA, May 18, 2026

*Based on the joint work [**den Boef et al., ’25**] and [**Benner et al., ’26**]

Outline

Outline

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

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

Improved sequence: $\hat{x}_3 \quad \hat{x}_4 \quad \ldots$

Non-Cycling Mode

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

Cycling Mode

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

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

Reduced Rank Extrapolation and Methodological Extensions

Extrapolation Methods for Fixed-Point Processes

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

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

Improved sequence: $\hat{x}_3 \quad \hat{x}_4 \quad \ldots$

Non-Cycling Mode

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

Cycling Mode

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

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

- 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 $\gamma_1 + \cdots + \gamma_w = 1$.
- 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*

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

  where $\gamma_1 + \cdots + \gamma_w = 1$.
- 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:

$x = Ax + b$	[Mešina, '77], [Eddy, '79](not spelled out), [Sidi, '88], [Sidi, '91]
$Ax = b$	[Sidi & Shapira, '98]

or only consider the fixed-point increment:

$Ax = b$	[Kaniel & Stein, '74]
$x = f(x)$	[Sidi, '20]

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)

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

Residual-Based RRE (new)

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

Target

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

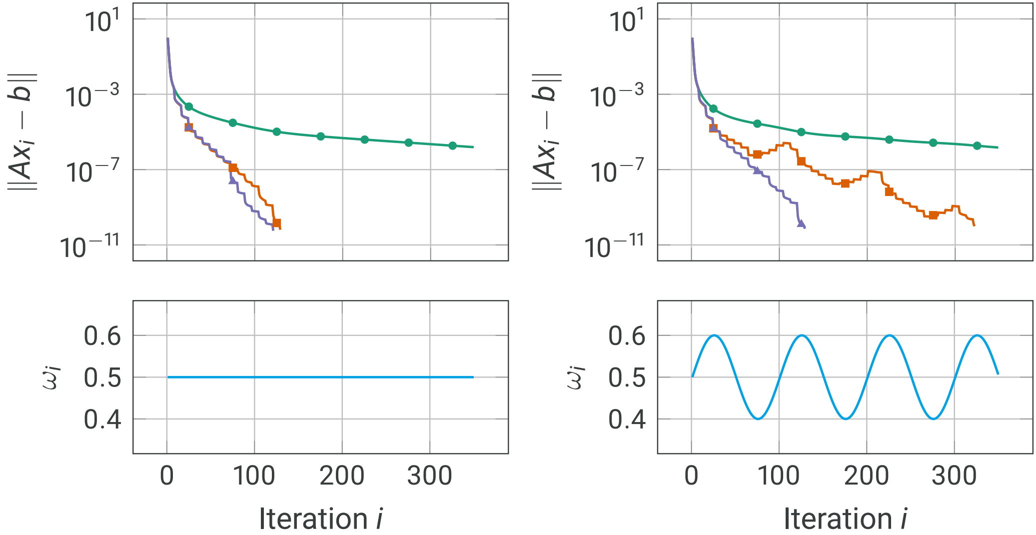


Figure: Comparison of increment-based (—■—) and residual-based (—▲—) RRE formulations applied to nonstationary successive over-relaxation schemes (—●—) for solving $Ax = b$.

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

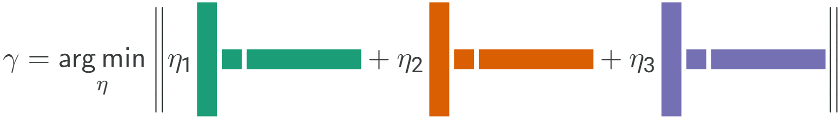
$$\gamma = \underset{\eta \in \mathbb{R}^w}{\arg\min} \left\| \sum_{i=1}^{w} \eta_i (b - Ax_i) \right\| \quad \text{s.t.} \quad \sum_{i=1}^{w} \eta_i = 1 \qquad \Longleftrightarrow \qquad U^{\mathsf{T}} U \alpha = 1 \in \mathbb{R}^w \text{ and } \gamma := \alpha / \|\alpha\|$$

$$U := \left[b - Ax_1 \middle| \dots \middle| b - Ax_w \right] \in \mathbb{R}^{d \times w}$$

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

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

Main idea:

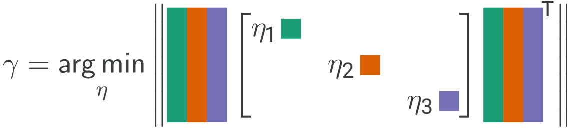


1. Move the arithmetic onto inner factors of dimension $r \ll d$
2. Reduce problem dimension via QR decomposition of outer factors (tall-and-skinny)

Benefits:

Solution of $U^{\mathsf{T}} U \alpha = 1 \in \mathbb{R}^{w}$ 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:



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 $U^{\mathsf{T}} U \alpha = 1 \in \mathbb{R}^{w}$ 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.

Reduced Rank Extrapolation and Methodological Extensions

Low-Rank Matrix Sequences

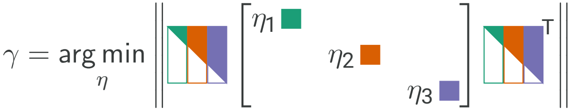
Main idea:

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 $U^\mathsf{T} U \alpha = 1 \in \mathbb{R}^w$ 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:



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 $U^{\mathsf{T}}U\alpha = 1 \in \mathbb{R}^{w}$ 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:

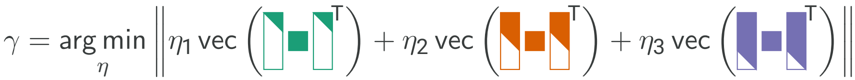
1. Move the arithmetic onto inner factors of dimension $r \ll d$
2. Reduce problem dimension via QR decomposition of outer factors (tall-and-skinny)

Benefits:

Solution of $U^{\mathsf{T}}U\alpha = 1 \in \mathbb{R}^{w}$ 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:



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 $U^{\mathsf{T}} U\alpha = 1 \in \mathbb{R}^{w}$ 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.

Algebraic Riccati Equation

- Underlying equation: algebraic Riccati equation (ARE)

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

- $X \approx Z D Z^{\mathsf{T}}$ and $\mathcal{R}(X) \approx R T R^{\mathsf{T}}$ **[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

$$X_i = Z_i D_i Z_i^{\mathsf{T}}, \qquad \mathscr{R}(X_i) = R_i T R_i^{\mathsf{T}},$$

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

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

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

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

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

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

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

$$\gamma = \underset{\sum_i \eta_i = 1}{\arg\min} \|\eta_i \mathscr{R}(X_i)\| = \underset{\sum_i \eta_i = 1}{\arg\min} \left\| \sum_{i=1}^{w} \eta_i R_i T R_i^{\mathsf{T}} \right\| = \underset{\sum_i \eta_i = 1}{\arg\min} \left\| \sum_{i=1}^{w} \eta_i \widetilde{R}_i T \widetilde{R}_i^{\mathsf{T}} \right\| = \underset{\sum_i \eta_i = 1}{\arg\min} \left\| \sum_{i=1}^{w} \eta_i \operatorname{vec}(\widetilde{R}_i T \widetilde{R}_i^{\mathsf{T}}) \right\|$$

with one thin QR decomposition

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

- 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:

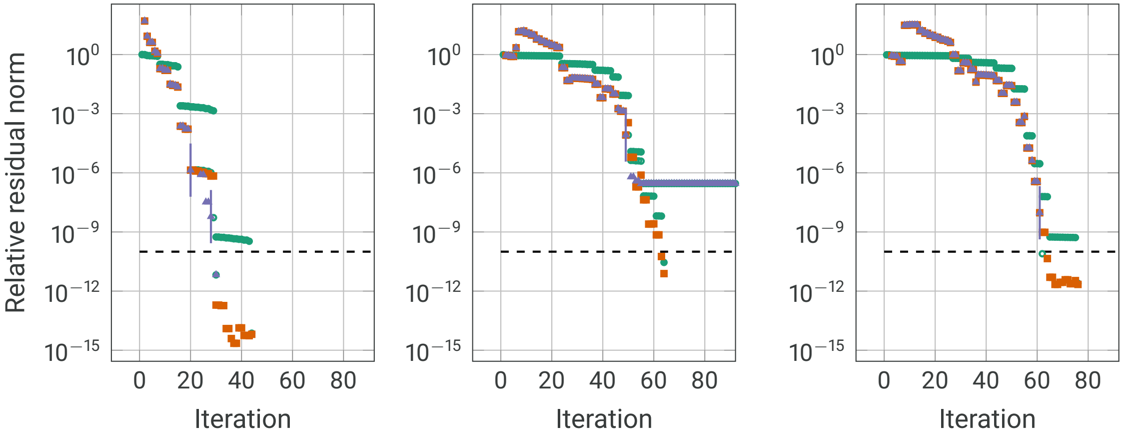
$$\widehat{X} = \sum_{i=1}^{w} \gamma_i X_i = \sum_{i=1}^{w} \tau_i V_i \widetilde{D}_i V_i^{\mathsf{T}}, \qquad \tau_i = \sum_{j=i}^{w} \gamma_j.$$

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

$$\tau_i \geq 0, \qquad i = 1, \ldots, w.$$

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

Nonlinear Toeplitz example [Benner et al., '20]: $d = 10^5$, for varying outputs q; $C \in \mathbb{R}^{q \times d}$.

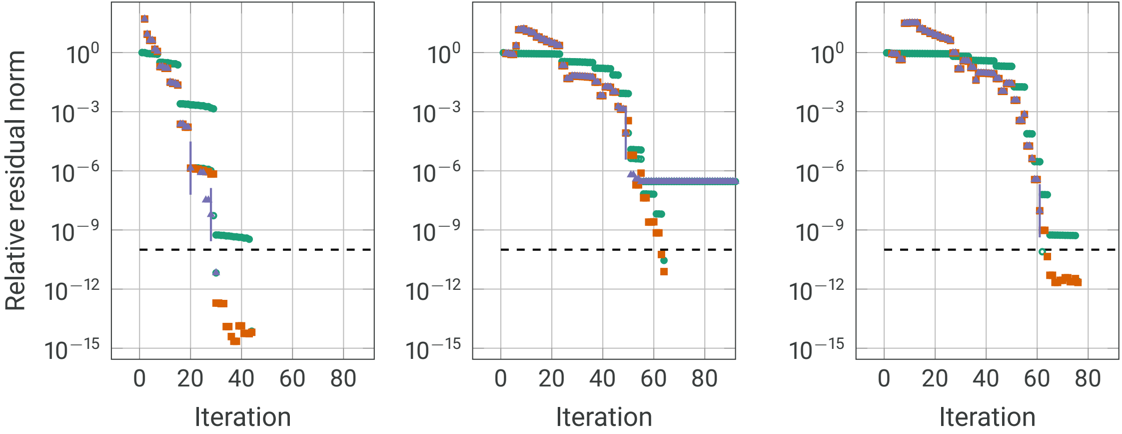


(a) $q = 1$. (b) $q = 20$. (c) $q = 40$.

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

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

Nonlinear Toeplitz example [Benner et al., '20]: $d = 10^5$, for varying outputs q; $C \in \mathbb{R}^{q \times d}$.

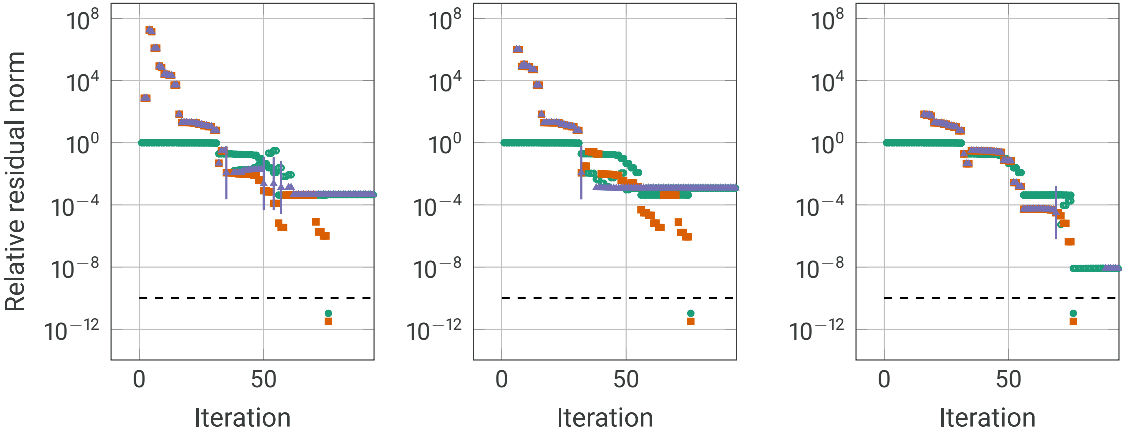


(a) $q = 1$. (b) $q = 20$. (c) $q = 40$.

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

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

Triple Chain mass-spring-damper example [Truhar & Veselić, '09]: $d = 60\,002$, varying w.

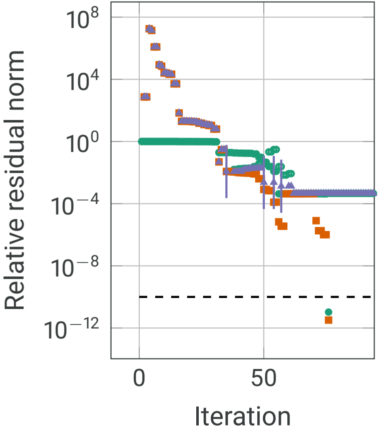


(a) $w = 3$. (b) $w = 6$. (c) $w = 12$.

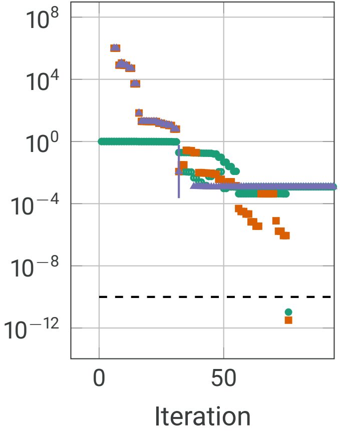
Figure: RADI (●○), non-cycling RRE (■), and cycling RRE (▲).

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

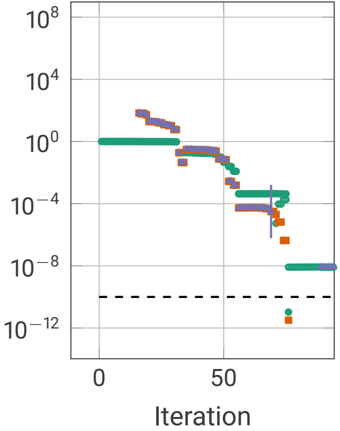
Triple Chain mass-spring-damper example [Truhar & Veselić, '09]: $d = 60\,002$, varying w.



(a) $w = 3$.



(b) $w = 6$.



(c) $w = 12$.

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

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

Outline

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

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

with the Sylvester part and additional low-rank terms

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

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

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

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

$$Y = F \; T \; G^T, \qquad X_k \approx Z_{L,k} \; D_k \; Z_{R,k}^T.$$

- 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 $\tau_{\mathrm{outer}}, \tau_{k,\mathrm{inner}}, \tau_{\mathrm{trunc}} > 0$ and solve

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

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

If a RRE cycle completes,

$$\widehat{X}_i \leftarrow \mathrm{RRE}_w(\widetilde{X}_i, \widetilde{X}_{i-1}, \dots, \widetilde{X}_{i-w}),$$

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

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

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

- Low-rank truncation $\mathcal{T}_{\mathrm{runc}}$ 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 $\tau_{k,\mathrm{inner}}$ *dynamically* s.t. the inner residual is *proportional to* the outer residual **[Shank, Simoncini, & Szyld, '15]**:

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

One Outer Step

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

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

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

If a RRE cycle completes,

$$\widehat{X}_i \leftarrow \text{RRE}_w(\widetilde{X}_i, \widetilde{X}_{i-1}, \ldots, \widetilde{X}_{i-w}),$$

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

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

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

- 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]**:

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

For low-rank iterates

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

increment-based RRE for MSE:

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

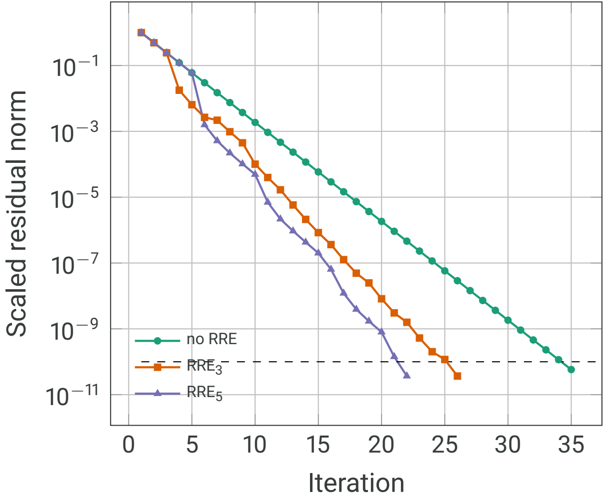
with two thin QR decompositions

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

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

Advection–Diffusion Example

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



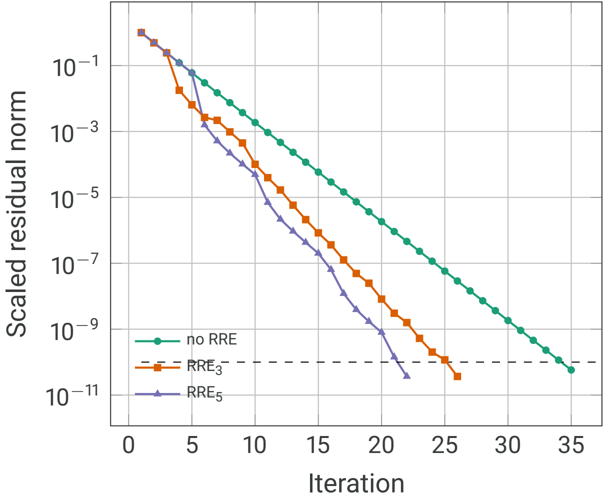
Summary at termination

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 $< 1\%$, $\mathcal{T}_{\text{runc}} \lesssim 15\%$

Advection–Diffusion Example

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



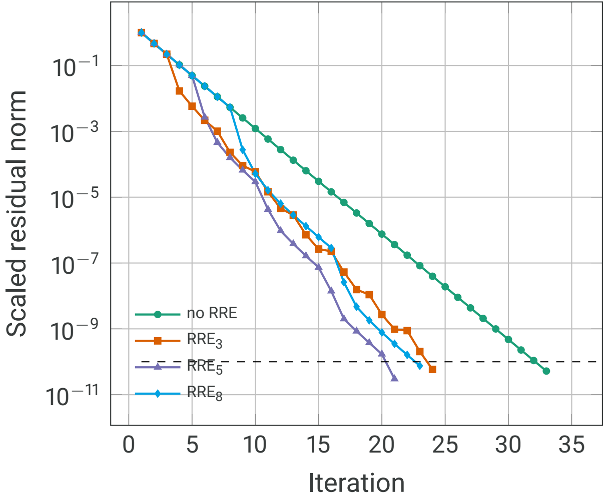
Summary at termination

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 $< 1\%$, $\mathcal{T}_{\text{runc}} \lesssim 15\%$

Multi-term Sylvester Example

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



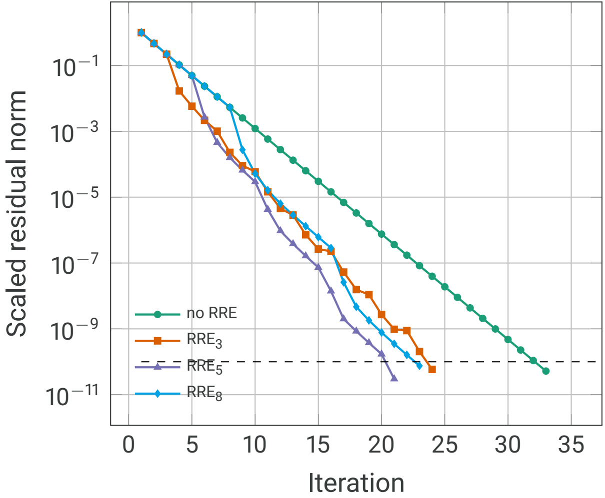
Summary at termination

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

- Only LR-ADI is used as inner solver
- 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

Multi-term Sylvester Example

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

- Only LR-ADI is used as inner solver
- 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

Outline

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:
  - Residual-based RRE for algebraic Riccati equation
  - Increment-based RRE for multi-term Sylvester equation
- Experimental validation: RRE can reduce both memory use and computing time

For more details:

https://arxiv.org/abs/2502.09165 and https://arxiv.org/abs/2603.12979

Further investigation:

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

- Check for acceleration methods for machine learning:

Y. Saad, *Acceleration methods for fixed-point iterations*, Acta Numerica, 34:805–890, 2025.

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:
  - Residual-based RRE for algebraic Riccati equation
  - Increment-based RRE for multi-term Sylvester equation
- Experimental validation: RRE can reduce both memory use and computing time

For more details:

https://arxiv.org/abs/2502.09165 and https://arxiv.org/abs/2603.12979

Further investigation:

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

• Check for acceleration methods for machine learning:

Y. Saad, *Acceleration methods for fixed-point iterations*, Acta Numerica, 34:805–890, 2025.

Bibliographies I

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.

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.

Bibliographies III

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.
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.