On the versatility of Krylov subspaces in modern matrix computations

Valeria Simoncini

Dipartimento di Matematica
Alma Mater Studiorum - Università di Bologna
valeria.simoncini@unibo.it

Algebraic computations. I

Old and new challenges in Scientific Computing

- Solution of block-structured/preconditioned large linear systems,

$$Ax = b \qquad n \times n$$

- Eigensolver requiring spectral transformations

$$Ax = \lambda M x, \qquad \|x\| = 1,$$

- Large scale matrix function evaluations

$$x = \exp(A)v, \quad x = \sqrt{A}v, \quad etc.$$

- Matrix and Tensor equations

$$(A_1 \otimes B_1 \otimes C_1 + \ldots + A_\ell \otimes B_\ell \otimes C_\ell)x = b$$

Algebraic computations. II

Old and new frameworks in Scientific Computing

- Many-dimensional problems (high-dim tensorized form)
  - Algebraic formulations
  - Memory constraints (for data and **solution**)
- Finite Precision computations
  - Rigorous round-off error analysis vs flexibility
  - Accuracy tradeoffs
- Mixed-precision computations
  - High performance machines
  - Computation lightening

Algebraic computations. II

Old and new frameworks in Scientific Computing

- Many-dimensional problems (high-dim tensorized form)
  - Algebraic formulations
  - Memory constraints (for data and **solution**)
- Finite Precision computations
  - Rigorous round-off error analysis vs flexibility
  - Accuracy tradeoffs
- Mixed-precision computations
  - High performance machines
  - Computation lightening

Algebraic computations. II

Old and new frameworks in Scientific Computing

- Many-dimensional problems (high-dim tensorized form)
  - Algebraic formulations
  - Memory constraints (for data and **solution**)
- Finite Precision computations
  - Rigorous round-off error analysis vs flexibility
  - Accuracy tradeoffs
- Mixed-precision computations
  - High performance machines
  - Computation lightening

The framework - iterative methods

- **Inexact** operator $v \to \mathcal{A}_\epsilon(v)$

  where $\mathcal{A}_\epsilon \to \mathcal{A}$ for $\epsilon \to 0$ (ϵ may be tuned)

  (e.g., Preconditioning, Schur complements, spectral transformations, etc.)

- Truncated computations:

  Inner products, matrix and vector sums

Classical nightmare

Accuracy and optimality properties are lost

Goal: Achieve approximation x_m to x within a fixed tolerance, by using $\mathcal{A}_\epsilon$ (and *not* $\mathcal{A}$), with variable ϵ

The framework - iterative methods

- Inexact operator $v \to \mathcal{A}_\epsilon(v)$

  where $\mathcal{A}_\epsilon \to \mathcal{A}$ for $\epsilon \to 0$ (ϵ may be tuned)

  (e.g., Preconditioning, Schur complements, spectral transformations, etc.)

- Truncated computations:

  Inner products, matrix and vector sums

Classical nightmare

Accuracy and optimality properties are lost

Goal: Achieve approximation x_m to x within a fixed tolerance, by using $\mathcal{A}_\epsilon$ (and *not* $\mathcal{A}$), with variable ϵ

The framework - iterative methods

- Inexact operator $v \to \mathcal{A}_\epsilon(v)$

  where $\mathcal{A}_\epsilon \to \mathcal{A}$ for $\epsilon \to 0$ (ϵ may be tuned)

  (e.g., Preconditioning, Schur complements, spectral transformations, etc.)

- Truncated computations:

  Inner products, matrix and vector sums

Classical nightmare

Accuracy and optimality properties are lost

Goal: Achieve approximation x_m to x within a fixed tolerance, by using $\mathcal{A}_\epsilon$ (and *not* $\mathcal{A}$), with variable ϵ

The framework - iterative methods

- Inexact operator $v \to \mathcal{A}_\epsilon(v)$

  where $\mathcal{A}_\epsilon \to \mathcal{A}$ for $\epsilon \to 0$ (ϵ may be tuned)

  (e.g., Preconditioning, Schur complements, spectral transformations, etc.)

- Truncated computations:

  Inner products, matrix and vector sums

Classical nightmare

Accuracy and optimality properties are lost

Goal: Achieve approximation x_m to x within a fixed tolerance, by using $\mathcal{A}_\epsilon$ (and *not* $\mathcal{A}$), with variable ϵ

The important ingredients

- Inexact operator $v \to \mathcal{A}_\epsilon(v)$:

$$y = \mathcal{A}_\epsilon(v) = \mathcal{A}v + w, \qquad \|w\| = \epsilon(v)$$

- Incremental approximation: growing subspace, with basis $V_m = [v_1, \ldots, v_m]$,

$$x_m = V_m y_m = \sum_{i=1}^{m} v_i (y_m)_i$$

$\Rightarrow$ The whole of y_m may change at each iteration, but

crucial property

The components of y_m have a decaying pattern

The important ingredients

- Inexact operator $v \to \mathcal{A}_\epsilon(v)$:

$$y = \mathcal{A}_\epsilon(v) = \mathcal{A}v + w, \qquad \|w\| = \epsilon(v)$$

- Incremental approximation: growing subspace, with basis $V_m = [v_1, \ldots, v_m]$,

$$x_m = V_m y_m = \sum_{i=1}^{m} v_i (y_m)_i$$

$\Rightarrow$ The whole of y_m may change at each iteration, but

crucial property

The components of y_m have a decaying pattern

The exact approach. Application of an operator.

To focus our attention: $\mathcal{A} = A$.

$$\mathcal{K}_m = \mathrm{span}\{v, Av, \ldots, A^{m-1}v\} \quad \text{Krylov subspace}$$

♣ $V_m = [v_1, \ldots, v_m]$, orth basis, obtained with Arnoldi (Gram-Schmidt) process

$$v_1 = \frac{v}{\|v\|}, \quad \hat{v} = Av_m - \sum_{i=1}^{m} v_i (v_i^T A v_m), \quad v_{m+1} = \frac{\hat{v}}{\|\hat{v}\|}$$

⇒ Arnoldi relation:

$$AV_m = V_{m+1}\underline{H}_m \qquad v = V_{m+1}e_1\beta \qquad \underline{H}_m = \begin{bmatrix} H_m \\ h_{m+1,m}e_m^T \end{bmatrix}$$

System: $x_m \in \mathcal{K}_m \quad \Rightarrow \quad x_m = V_m y_m \qquad (x_0 = 0)$

Eigenpb: (θ, y) eigenpair of $H_m \quad \Rightarrow \quad (\theta, V_m y)$ Ritz pair for (λ, x)

The exact approach. Application of an operator.

To focus our attention: $\mathcal{A} = A$.

$$\mathcal{K}_m = \mathrm{span}\{v, Av, \ldots, A^{m-1}v\} \quad \text{Krylov subspace}$$

♣ $V_m = [v_1, \ldots, v_m]$, orth basis, obtained with Arnoldi (Gram-Schmidt) process

$$v_1 = \frac{v}{\|v\|}, \quad \hat{v} = Av_m - \sum_{i=1}^{m} v_i (v_i^T A v_m), \;\; v_{m+1} = \frac{\hat{v}}{\|\hat{v}\|}$$

⇒ Arnoldi relation:

$$AV_m = V_{m+1}\underline{H}_m \qquad v = V_{m+1}e_1\beta \qquad \underline{H}_m = \begin{bmatrix} H_m \\ h_{m+1,m}e_m^T \end{bmatrix}$$

System: $x_m \in \mathcal{K}_m \quad \Rightarrow \quad x_m = V_m y_m \qquad (x_0 = 0)$

Eigenpb: (θ, y) eigenpair of $H_m \quad \Rightarrow \quad (\theta, V_m y)$ Ritz pair for (λ, x)

The inexact key relation

$\mathcal{A}$ is not available

$$\mathcal{A} = A \quad \to \quad \mathcal{A}_\epsilon \approx A$$

e.g., $\mathcal{A}_\epsilon v := \mathcal{A}v + f, \qquad \|f\| = \epsilon$

$$AV_m = V_{m+1}\underline{H}_m + \underbrace{F_m}_{[f_1, f_2, \ldots, f_m]} \qquad F_m \text{ error matrix, } \|f_j\| = O(\epsilon_j)$$

How large is F_m allowed to be?

system:

$$\begin{aligned} r_m &= b - AV_m y_m = b - V_{m+1}\underline{H}_m y_m - F_m y_m \\ &= \underbrace{V_{m+1}(e_1\beta - \underline{H}_m y_m)}_{\text{computed residual} =: \tilde{r}_m} - F_m y_m \end{aligned}$$

eigenproblem: $\quad (\theta, V_m y)$

$$r_m = \theta V_m y - AV_m y = v_{m+1} h_{m+1,m} e_m^T y - F_m y$$

The inexact key relation

$\mathcal{A}$ is **not** available

$$\mathcal{A} = A \quad \rightarrow \quad \mathcal{A}_\epsilon \approx A$$

e.g., $\mathcal{A}_\epsilon v := \mathcal{A} v + f, \qquad \|f\| = \epsilon$

$$AV_m = V_{m+1}\underline{H}_m + \underbrace{F_m}_{[f_1, f_2, \ldots, f_m]} \qquad F_m \text{ error matrix}, \ \|f_j\| = O(\epsilon_j)$$

How large is F_m allowed to be?

system:

$$\begin{aligned} r_m &= b - AV_m y_m = b - V_{m+1}\underline{H}_m y_m - F_m y_m \\ &= \underbrace{V_{m+1}(e_1\beta - \underline{H}_m y_m)}_{\text{computed residual} =: \tilde{r}_m} - F_m y_m \end{aligned}$$

eigenproblem: $\qquad (\theta, V_m y)$

$$r_m = \theta V_m y - AV_m y = v_{m+1} h_{m+1,m} e_m^T y - F_m y$$

The inexact key relation

$\mathcal{A}$ is not available

$$\mathcal{A} = A \quad \rightarrow \quad \mathcal{A}_\epsilon \approx A$$

e.g., $\mathcal{A}_\epsilon v := \mathcal{A} v + f, \qquad \|f\| = \epsilon$

$$AV_m = V_{m+1}\underline{H}_m + \underbrace{F_m}_{[f_1, f_2, \ldots, f_m]} \qquad F_m \text{ error matrix, } \|f_j\| = O(\epsilon_j)$$

How large is F_m allowed to be?

system:

$$\begin{aligned} r_m &= b - AV_m y_m = b - V_{m+1}\underline{H}_m y_m - F_m y_m \\ &= \underbrace{V_{m+1}(e_1\beta - \underline{H}_m y_m)}_{\text{computed residual} =: \tilde{r}_m} - F_m y_m \end{aligned}$$

eigenproblem: $(\theta, V_m y)$

$$r_m = \theta V_m y - AV_m y = v_{m+1} h_{m+1,m} e_m^T y - F_m y$$

A dynamic setting

$$\boxed{\text{true (unobservable) residual} = \text{computable residual} \; - F_m y}$$

$$F_m y = [f_1, f_2, \ldots, f_m] \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_m \end{bmatrix} = \sum_{i=1}^{m} f_i \eta_i$$

- ⋄ The terms $f_i \eta_i$ need to be small:

$$\|f_i \eta_i\| < \frac{1}{m}\epsilon \quad \forall i \quad \Rightarrow \quad \|F_m y\| < \epsilon$$

- ⋄ If η_i small $\quad \Rightarrow \quad f_i$ is allowed to be large

A dynamic setting

$$\boxed{\boxed{\text{true (unobservable) residual} = \text{computable residual } - F_m y}}$$

$$F_m y = [f_1, f_2, \ldots, f_m] \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_m \end{bmatrix} = \sum_{i=1}^{m} f_i \eta_i$$

◇ The terms $f_i \eta_i$ need to be small:

$$\|f_i \eta_i\| < \frac{1}{m}\epsilon \quad \forall i \quad \Rightarrow \quad \|F_m y\| < \epsilon$$

◇ If η_i small $\quad \Rightarrow \quad$ f_i is allowed to be large

Linear systems: The solution pattern

$y_m = [\eta_1; \eta_2; \dots; \eta_m]$ depends on the chosen method, e.g.

- GMRES: $y_m = \mathrm{argmin}_y \|e_1\beta - \underline{H}_m y\|$,

$$|\eta_i| \le \frac{1}{\sigma_{\min}(\underline{H}_m)} \|\tilde{r}_{i-1}\|$$

$\tilde{r}_{i-1}$: GMRES computed residual at iteration $i-1$.

Simoncini & Szyld, '03 (see also Sleijpen & van den Eshof, '04, Bouras-Frayssé '05)

Analogous result for Galerkin methods (e.g. FOM)

Relaxing the inexactness in A

$A \cdot v_i$ not performed exactly $\quad \Rightarrow \quad (A + E_i) \cdot v_i$

True (unobservable) vs. computed residuals:

$$r_m = b - AV_m y_m = V_{m+1}(e_1\beta - \underline{H}_m y_m) - F_m y_m$$

GMRES: If (Similar result for FOM)

$$\|E_i\| \le \frac{\sigma_{\min}(\underline{H}_m)}{m} \frac{1}{\|\tilde{r}_{i-1}\|} \varepsilon \quad i = 1, \ldots, m$$

then $\|F_m y_m\| \le \varepsilon \quad \Rightarrow \quad \|r_m - V_{m+1}(e_1\beta - \underline{H}_m y_m)\| \le \varepsilon$

$\tilde{r}_{i-1}$: GMRES computed residual at iteration $i - 1$

An example: Schur complement

$$\underbrace{B^T S^{-1} B}_{A} x = b \qquad\qquad y_i \leftarrow B^T S^{-1} B v_i$$

Inexact matrix-vector product:

$$\begin{cases} \text{Solve } S w_i = B v_i \\ \text{Compute } y_i = B^T w_i \end{cases} \quad \overset{\text{Inexact}}{\Rightarrow} \quad \begin{cases} \text{Approx solve } S w_i = B v_i \quad \Rightarrow \widehat{w}_i \\ \text{Compute } \widehat{y}_i = B^T \widehat{w}_i \end{cases}$$

$w_i = \widehat{w}_i + \epsilon_i$ $\qquad$ ϵ_i error in inner solution $\qquad$ so that

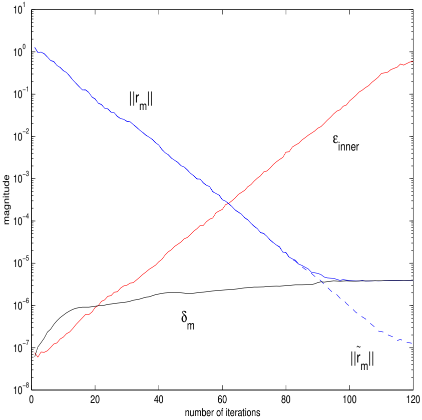
$$A v_i \quad \rightarrow \quad B^T \widehat{w}_i = \underbrace{B^T w_i}_{A v_i} - \underbrace{B^T \epsilon_i}_{-E_i v_i} = (A + E_i) v_i$$

Numerical experiment

$$\underbrace{B^T S^{-1} B}_{A} x = b \qquad \text{at each it. } i \text{ solve} \quad S w_i = B v_i$$

Inexact FOM

$\delta_m = \|r_m - (b - V_{m+1}\underline{H}_m y_m)\|$



Different problems. Similar setting.

Approximating the evaluation of a matrix function
Given $V_m \in \mathbb{R}^{n\times m}$ whose columns are an orthogonal basis of some approximation space, $0 \neq t \in \mathbb{R}$,

$$f(tA)v \approx \mathbf{u}_m := V_m f(tH_m)e_1,$$

"Residual" evaluation:

$$r_m(t) := |h_{m+1,m}\mathbf{e}_m^T f(tH_m)\mathbf{e}_1|, \qquad h_{m+1,m} = v_{m+1}^\top A V_m$$

If $u(t) = f(tA)v$ is the solution to the differential equation $u^{(d)} = Au$ for some derivative d, then

$$\mathbf{r}_m(t) = A\mathbf{u}_m - \mathbf{u}_m^{(d)} = AV_m f(tH_m)\mathbf{e}_1 - \mathbf{u}_m^{(d)} = \ldots = \mathbf{v}_{m+1}h_{m+1,m}\mathbf{e}_m^T f(tH_m)\mathbf{e}_1$$

Distance between exact and computable residuals: for $F_m = [\mathbf{f}_1, \ldots, \mathbf{f}_m]$,

$$|\|\mathbf{r}_m\| - r_m| \leq \|[\mathbf{f}_1, \ldots, \mathbf{f}_m]f(tH_m)\mathbf{e}_1\| \leq \sum_{j=1}^{m} \|\mathbf{f}_j\|\, |\mathbf{e}_j^T f(tH_m)\mathbf{e}_1|$$

Proof of element-wise decay of $f(tH_m)\mathbf{e}_1$ in Pozza-Simoncini, BIT '19

Different problems. Similar setting.

Approximating the evaluation of a matrix function
Given $V_m \in \mathbb{R}^{n\times m}$ whose columns are an orthogonal basis of some approximation space, $0 \neq t \in \mathbb{R}$,

$$f(tA)v \approx \mathbf{u}_m := V_m f(tH_m)e_1,$$

"Residual" evaluation:

$$r_m(t) := |h_{m+1,m}\mathbf{e}_m^T f(tH_m)\mathbf{e}_1|, \qquad h_{m+1,m} = v_{m+1}^\top A V_m$$

If $u(t) = f(tA)v$ is the solution to the differential equation $u^{(d)} = Au$ for some derivative d, then

$$\mathbf{r}_m(t) = A\mathbf{u}_m - \mathbf{u}_m^{(d)} = AV_m f(tH_m)\mathbf{e}_1 - \mathbf{u}_m^{(d)} = \ldots = \mathbf{v}_{m+1}h_{m+1,m}\mathbf{e}_m^T f(tH_m)\mathbf{e}_1$$

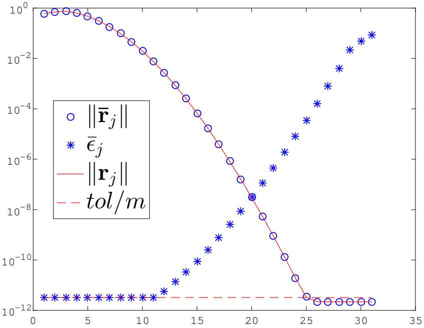
Distance between exact and computable residuals: for $F_m = [\mathbf{f}_1, \ldots, \mathbf{f}_m]$,

$$|\|\mathbf{r}_m\| - r_m| \leq \|[\mathbf{f}_1, \ldots, \mathbf{f}_m] f(tH_m)\mathbf{e}_1\| \leq \sum_{j=1}^m \|\mathbf{f}_j\|\, |\mathbf{e}_j^T f(tH_m)\mathbf{e}_1|$$

Proof of element-wise decay of $f(tH_m)\mathbf{e}_1$ in Pozza-Simoncini, BIT '19

An example. Matrix pde225 from the Matrix Market repository

Approximation of $e^{-A}\mathbf{v}$ with $\mathbf{v} = 1$ (normalized)



* Residual norm $\|\mathbf{r}_j\|$ with constant accuracy $\epsilon_j = tol/m$,
* residual norm $\|\bar{\mathbf{r}}_j\|$ with a variable strategy for the perturbation $\bar{\epsilon}_j$ as the inexact Arnoldi method proceeds

Multiterm linear matrix equation. 1

$$A_1 \boldsymbol{X} B_1 + A_2 \boldsymbol{X} B_2 + \ldots + A_\ell \boldsymbol{X} B_\ell = C$$

$A_i \in \mathbb{R}^{n\times n}$, $B_i \in \mathbb{R}^{m\times m}$, $\boldsymbol{X}$ unknown matrix

Possibly large dimensions, structured coefficient matrices

Kronecker formulation $\boxed{\left(B_1^\top \otimes A_1 + \ldots + B_\ell^\top \otimes A_\ell\right) \boldsymbol{x} = c \;\Leftrightarrow\; \mathcal{A}\boldsymbol{x} = c}$

$\mathcal{A} \in \mathbb{R}^{nm\times nm}$. ... Iterative methods

(Benner, Breiten, Bouhamidi, Chehab, Damm, Grasedyck, Jbilou, Kressner, Matthies, Nagy, Onwunta, Raydan, Stoll, Tobler, Wedderburn, Zander, ...)

♣ Same framework for multiple Kronecker terms, e.g.,

$$\left(A_1 \otimes B_1 \otimes C_1 + \ldots + A_\ell \otimes B_\ell \otimes C_\ell\right) \boldsymbol{x} = d$$

Multiterm linear matrix equation. 1

$$A_1 \boldsymbol{X} B_1 + A_2 \boldsymbol{X} B_2 + \ldots + A_\ell \boldsymbol{X} B_\ell = C$$

$A_i \in \mathbb{R}^{n\times n}$, $B_i \in \mathbb{R}^{m\times m}$, $\boldsymbol{X}$ unknown matrix

Possibly large dimensions, structured coefficient matrices

Kronecker formulation $\boxed{\left(B_1^\top \otimes A_1 + \ldots + B_\ell^\top \otimes A_\ell\right) \boldsymbol{x} = c \;\Leftrightarrow\; \mathcal{A}\boldsymbol{x} = c}$

$\mathcal{A} \in \mathbb{R}^{nm\times nm}$. ... Iterative methods

(Benner, Breiten, Bouhamidi, Chehab, Damm, Grasedyck, Jbilou, Kressner, Matthies, Nagy, Onwunta, Raydan, Stoll, Tobler, Wedderburn, Zander, ...)

♣ Same framework for multiple Kronecker terms, e.g.,

$$(A_1 \otimes B_1 \otimes C_1 + \ldots + A_\ell \otimes B_\ell \otimes C_\ell)\, \boldsymbol{x} = d$$

Multiterm linear matrix equation. 1

$$A_1 \boldsymbol{X} B_1 + A_2 \boldsymbol{X} B_2 + \ldots + A_\ell \boldsymbol{X} B_\ell = C$$

$A_i \in \mathbb{R}^{n\times n}$, $B_i \in \mathbb{R}^{m\times m}$, $\boldsymbol{X}$ unknown matrix

Possibly large dimensions, structured coefficient matrices

Kronecker formulation $\boxed{\left(B_1^\top \otimes A_1 + \ldots + B_\ell^\top \otimes A_\ell\right) \boldsymbol{x} = c \;\Leftrightarrow\; \mathcal{A}\boldsymbol{x} = c}$

$\mathcal{A} \in \mathbb{R}^{nm\times nm}$. ... Iterative methods

(Benner, Breiten, Bouhamidi, Chehab, Damm, Grasedyck, Jbilou, Kressner, Matthies, Nagy, Onwunta, Raydan, Stoll, Tobler, Wedderburn, Zander, ...)

♣ Same framework for multiple Kronecker terms, e.g.,

$$(A_1 \otimes B_1 \otimes C_1 + \ldots + A_\ell \otimes B_\ell \otimes C_\ell)\, \boldsymbol{x} = d$$

Multiterm linear matrix equation. 2

Iterative methods: matrix-matrix multiplications and rank truncation

$$A_1 \boldsymbol{X} B_1 + A_2 \boldsymbol{X} B_2 + \ldots + A_\ell \boldsymbol{X} B_\ell = C$$

Alternatives to Kronecker form:

- Fixed point iterations (an "evergreen"...)
- Projection-type methods $\Rightarrow$ low rank approximation
- Ad-hoc problem-dependent procedures
- etc.

♣ Many discretized problems now take this form
(SPDEs, parameter-dep PDEs, space-time PDEs, etc.)

Currently a very active area of research

Multiterm linear matrix equation. 2

Iterative methods: matrix-matrix multiplications and rank truncation

$$A_1 \boldsymbol{X} B_1 + A_2 \boldsymbol{X} B_2 + \ldots + A_\ell \boldsymbol{X} B_\ell = C$$

Alternatives to Kronecker form:

- Fixed point iterations (an "evergreen"...)
- Projection-type methods $\Rightarrow$ low rank approximation
- Ad-hoc problem-dependent procedures
- etc.

♣ Many discretized problems now take this form
(SPDEs, parameter-dep PDEs, space-time PDEs, etc.)

Currently a very active area of research

Iterative methods: matrix-matrix products and rank truncation

$$A_1 X B_1 + A_2 X B_2 + \ldots + A_\ell X B_\ell = C, \qquad C \text{ low rank}$$

Kronecker formulation in disguise:

$$\left(B_1^\top \otimes A_1 + \ldots + B_\ell^\top \otimes A_\ell\right) \boldsymbol{x} = c$$

Conjugate Gradients: Use X instead of $\boldsymbol{x}$, where $\boldsymbol{x} = \mathrm{vec}(X)$,

Matrix-oriented "thinking". Update:

$$x_{k+1} = x_k + \alpha_k p_k \quad \Rightarrow \quad X_{k+1} = X_k + \alpha_k P_k$$

Matrix-oriented "thinking". Truncate:

$$X_k = U_k U_k^T \quad \rightarrow \quad \widetilde{X}_{k+1} = \widetilde{U}_{k+1} \widetilde{U}_{k+1}^T \quad \rightarrow \quad X_{k+1} = \mathrm{trunc}(\widetilde{X}_{k+1}) = U_{k+1} U_{k+1}^T$$

(here P_k also low rank) $\mathrm{trunc}(\widetilde{X}_{k+1})$ acts on the SVD of X_{k+1}

Iterative methods: matrix-matrix products and rank truncation

$$A_1XB_1 + A_2XB_2 + \ldots + A_\ell XB_\ell = C, \qquad C \text{ low rank}$$

Kronecker formulation in disguise:

$$\left(B_1^\top \otimes A_1 + \ldots + B_\ell^\top \otimes A_\ell\right) \boldsymbol{x} = c$$

Conjugate Gradients: Use X instead of $\boldsymbol{x}$, where $\boldsymbol{x} = \mathrm{vec}(X)$,

Matrix-oriented "thinking". Update:

$$x_{k+1} = x_k + \alpha_k p_k \quad \Rightarrow \quad X_{k+1} = X_k + \alpha_k P_k$$

Matrix-oriented "thinking". Truncate:

$$X_k = U_k U_k^T \quad \to \quad \widetilde{X}_{k+1} = \widetilde{U}_{k+1}\widetilde{U}_{k+1}^T \quad \to \quad X_{k+1} = \mathrm{trunc}(\widetilde{X}_{k+1}) = U_{k+1}U_{k+1}^T$$

(here P_k also low rank) $\quad$ $\mathrm{trunc}(\widetilde{X}_{k+1})$ acts on the SVD of X_{k+1}

Iterative methods: matrix-matrix products and rank truncation

$$A_1XB_1 + A_2XB_2 + \ldots + A_\ell XB_\ell = C, \qquad C \text{ low rank}$$

Kronecker formulation in disguise:

$$\left(B_1^\top \otimes A_1 + \ldots + B_\ell^\top \otimes A_\ell\right) \boldsymbol{x} = c$$

Conjugate Gradients: Use X instead of $\boldsymbol{x}$, where $\boldsymbol{x} = \mathrm{vec}(X)$,

Matrix-oriented "thinking". Update:

$$x_{k+1} = x_k + \alpha_k p_k \quad \Rightarrow \quad X_{k+1} = X_k + \alpha_k P_k$$

Matrix-oriented "thinking". Truncate:

$$X_k = U_k U_k^T \quad \to \quad \widetilde{X}_{k+1} = \widetilde{U}_{k+1}\widetilde{U}_{k+1}^T \quad \to \quad X_{k+1} = \mathrm{trunc}(\widetilde{X}_{k+1}) = U_{k+1}U_{k+1}^T$$

(here P_k also low rank) $\qquad \mathrm{trunc}(\widetilde{X}_{k+1})$ acts on the SVD of X_{k+1}

Truncated matrix-oriented CG (TCG) for Kronecker form

Input: $\mathcal{A}(\boldsymbol{X}) = A_1 \boldsymbol{X} B_1 + A_2 \boldsymbol{X} B_2 + \ldots + A_\ell \boldsymbol{X} B_\ell$, right-hand side $C \in \mathbb{R}^{n\times n}$ in low-rank format. Truncation operator trunc.
Output: Matrix $X \in \mathbb{R}^{n\times n}$ in low-rank format s.t. $||\mathcal{A}(X) - C||_F / ||C||_F \le tol$

1. $X_0 = 0,\ R_0 = C,\ P_0 = R_0,\ Q_0 = \mathcal{A}(P_0)$
2. $\xi_0 = \langle P_0, Q_0\rangle,\ k = 0$ $\qquad \langle X, Y\rangle = \mathrm{tr}(X^\top Y)$
3. While $||R_k||_F > tol$
4. $\quad \omega_k = \langle R_k, P_k\rangle / \xi_k$
5. $\quad X_{k+1} = X_k + \omega_k P_k,$ $\qquad X_{k+1} \leftarrow \mathrm{trunc}(X_{k+1})$
6. $\quad R_{k+1} = C - \mathcal{A}(X_{k+1}),$ $\qquad$ Optionally: $R_{k+1} \leftarrow \mathrm{trunc}(R_{k+1})$
7. $\quad \beta_k = -\langle R_{k+1}, Q_k\rangle / \xi_k$
8. $\quad P_{k+1} = R_{k+1} + \beta_k P_k,$ $\qquad P_{k+1} \leftarrow \mathrm{trunc}(P_{k+1})$
9. $\quad Q_{k+1} = \mathcal{A}(P_{k+1}),$ $\qquad$ Optionally: $Q_{k+1} \leftarrow \mathrm{trunc}(Q_{k+1})$
10. $\quad \xi_{k+1} = \langle P_{k+1}, Q_{k+1}\rangle$
11. $\quad k = k + 1$
12. end while

♣ Iterates kept in factored form! $\qquad$ Kressner and Tobler, 2011
(truncation by tolerance and/or max rank)

Truncated matrix-oriented CG (TCG) for Kronecker form

Input: $\mathcal{A}(\boldsymbol{X}) = A_1\boldsymbol{X}B_1 + A_2\boldsymbol{X}B_2 + \ldots + A_\ell\boldsymbol{X}B_\ell$, right-hand side $C \in \mathbb{R}^{n\times n}$ in low-rank format. Truncation operator trunc.
Output: Matrix $X \in \mathbb{R}^{n\times n}$ in low-rank format s.t. $||\mathcal{A}(X) - C||_F/||C||_F \le tol$

1. $X_0 = 0,\ R_0 = C,\ P_0 = R_0,\ Q_0 = \mathcal{A}(P_0)$
2. $\xi_0 = \langle P_0, Q_0\rangle,\ k = 0$ $\qquad\qquad \langle X, Y\rangle = \mathrm{tr}(X^\top Y)$
3. While $||R_k||_F > tol$
4. $\quad \omega_k = \langle R_k, P_k\rangle/\xi_k$
5. $\quad X_{k+1} = X_k + \omega_k P_k,$ $\qquad X_{k+1} \leftarrow \mathrm{trunc}(X_{k+1})$
6. $\quad R_{k+1} = C - \mathcal{A}(X_{k+1}),$ $\qquad$ Optionally: $R_{k+1} \leftarrow \mathrm{trunc}(R_{k+1})$
7. $\quad \beta_k = -\langle R_{k+1}, Q_k\rangle/\xi_k$
8. $\quad P_{k+1} = R_{k+1} + \beta_k P_k,$ $\qquad P_{k+1} \leftarrow \mathrm{trunc}(P_{k+1})$
9. $\quad Q_{k+1} = \mathcal{A}(P_{k+1}),$ $\qquad$ Optionally: $Q_{k+1} \leftarrow \mathrm{trunc}(Q_{k+1})$
10. $\quad \xi_{k+1} = \langle P_{k+1}, Q_{k+1}\rangle$
11. $\quad k = k + 1$
12. end while

♣ Iterates kept in factored form! $\qquad$ Kressner and Tobler, 2011
(truncation by tolerance and/or max rank)

A very general reference strategy

This setting can accommodate various strategies:

- Rank and accuracy flexibility in (rhs) data
- Multiprecision and other memory conservative computations
- HPC implementations
- Fault tolerance implementations

Effect of truncation

Let $x_k = \mathrm{vec}(X_k)$ (and similarly for the other variables). Truncation can be written as

$$x^{(k+1)} = x_{ex}^{(k+1)} + \boldsymbol{e}_X^{(k+1)}, \qquad p^{(k+1)} = p_{ex}^{(k+1)} + \boldsymbol{e}_P^{(k+1)}$$

($\boldsymbol{e}_X^{(k+1)}, \boldsymbol{e}_P^{(k+1)}$ local truncation errors)

TH: Let $\Delta_k = \max\{\|\boldsymbol{e}_P^{(k)}\|, \|\boldsymbol{e}_X^{(k)}\|, \|\boldsymbol{e}_P^{(k+1)}\|, \|\boldsymbol{e}_X^{(k+1)}\|\}$ and also $\delta_k = \min\{\|\boldsymbol{e}_P^{(k)}\|, \|\boldsymbol{e}_X^{(k)}\|, \|\boldsymbol{e}_P^{(k+1)}\|, \|\boldsymbol{e}_X^{(k+1)}\|\}$. Then there exists $\eta \in [0,1]$ such that

$$\eta \frac{1}{\|\mathcal{A}^{-1}\|} \frac{\delta_k}{\|r^{(k+1)}\|} \le \frac{|r^{(k+1)})^\top p^{(k)}|}{\|r^{(k+1)}\|\|p^{(k)}\|} \le \|\mathcal{A}\| \frac{\Delta_k}{\|r^{(k+1)}\|},$$

and

$$\beta_k = -\frac{(r_{ex}^{(k+1)})^\top \mathcal{A} p^{(k)} - (\mathcal{A}\boldsymbol{e}_X^{(k+1)})^\top \mathcal{A} p^{(k)}}{(p^{(k)})^\top \mathcal{A} p^{(k)}}.$$

Moreover, with $\gamma = \|\mathcal{A}p^{(k)}\| + (2|\beta_{k-1}| + |\beta_{k-1}\alpha_k|)\|\mathcal{A}p^{(k-1)}\| + \|r^{(k+1)}\|)/\|r^{(k)}\|$

$$\frac{|r^{(k+1)})^\top r^{(k)}|}{\|r^{(k+1)}\|\|r^{(k)}\|} \le \gamma \frac{\Delta_k}{\|r^{(k+1)}\|}$$

Effect of truncation

Let $x_k = \mathrm{vec}(X_k)$ (and similarly for the other variables). Truncation can be written as

$$x^{(k+1)} = x_{ex}^{(k+1)} + \boldsymbol{e}_X^{(k+1)}, \qquad p^{(k+1)} = p_{ex}^{(k+1)} + \boldsymbol{e}_P^{(k+1)}$$

($\boldsymbol{e}_X^{(k+1)}, \boldsymbol{e}_P^{(k+1)}$ local truncation errors)

TH: Let $\Delta_k = \max\{\|\boldsymbol{e}_P^{(k)}\|, \|\boldsymbol{e}_X^{(k)}\|, \|\boldsymbol{e}_P^{(k+1)}\|, \|\boldsymbol{e}_X^{(k+1)}\|\}$ and also $\delta_k = \min\{\|\boldsymbol{e}_P^{(k)}\|, \|\boldsymbol{e}_X^{(k)}\|, \|\boldsymbol{e}_P^{(k+1)}\|, \|\boldsymbol{e}_X^{(k+1)}\|\}$. Then there exists $\eta \in [0, 1]$ such that

$$\eta \frac{1}{\|\mathcal{A}^{-1}\|} \frac{\delta_k}{\|r^{(k+1)}\|} \le \frac{|(r^{(k+1)})^\top p^{(k)}|}{\|r^{(k+1)}\|\|p^{(k)}\|} \le \|\mathcal{A}\| \frac{\Delta_k}{\|r^{(k+1)}\|},$$

and

$$\beta_k = -\frac{(r_{ex}^{(k+1)})^\top \mathcal{A} p^{(k)} - (\mathcal{A}\boldsymbol{e}_X^{(k+1)})^\top \mathcal{A} p^{(k)}}{(p^{(k)})^\top \mathcal{A} p^{(k)}}.$$

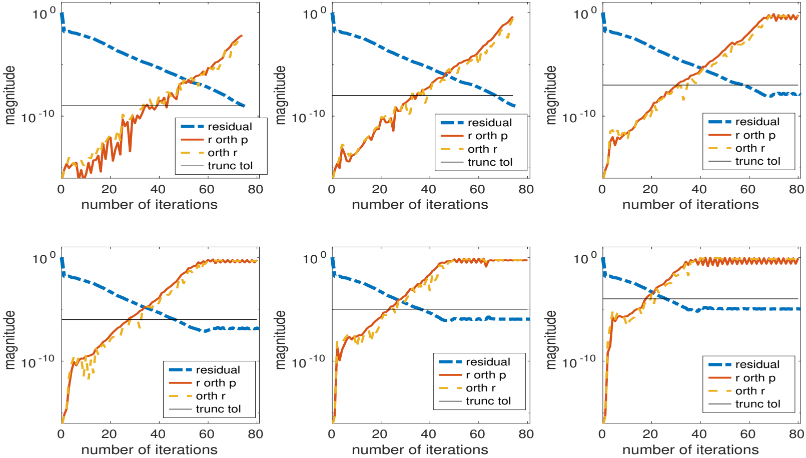
Moreover, with $\gamma = \|\mathcal{A}p^{(k)}\| + (2|\beta_{k-1}| + |\beta_{k-1}\alpha_k|)\|\mathcal{A}p^{(k-1)}\| + \|r^{(k+1)}\|)/\|r^{(k)}\|$

$$\frac{|(r^{(k+1)})^\top r^{(k)}|}{\|r^{(k+1)}\|\|r^{(k)}\|} \le \gamma \frac{\Delta_k}{\|r^{(k+1)}\|}$$

An example: $AX + XA + MXM = c_1 c_1^\top$

A: 2D Laplace operator, M =pentadiag$(-0.5, -1, \underline{3.2}, -1, -0.5)$, c_1 random entries

Truncated CG residual norm (blue line) for different truncation values



Also reported: Loss of orthogonality (cosine of the angles) between consecutive residuals and residual and directions

Conclusions

- Krylov-based approaches are very flexible
- Relaxation properties are versitile wrto problem
- Relaxation properties often arise in disguise
- Handling inexactness – instead of preventing it – is extremely useful in practice

Visit: www.dm.unibo.it/~simoncin

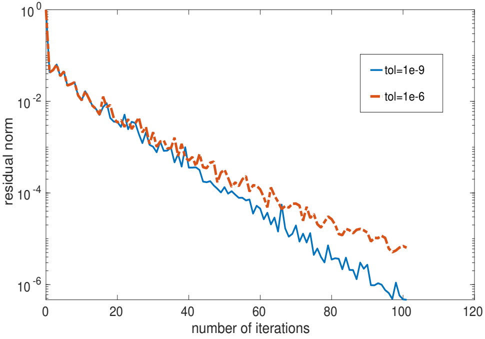
Email address: valeria.simoncini@unibo.it

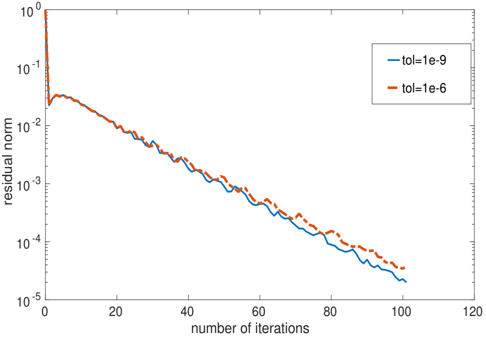
Another example. The tough problems may remain so.

$A = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ with $\lambda_i = \lambda_1 + \frac{(i-1)}{(n-1)}(\lambda_n - \lambda_1)\rho^{n-i}$, $\lambda_1 = 0.1, \quad \lambda_n = 100$

M: diagonal matrix with elements logarithmically distributed in $[10^{-2}, 10^0]$

Convergence history of TCG for two truncation tolerances:





Left: $\rho = 0.4$ Right: $\rho = 0.8$

Different problems. Similar setting. 2

Large scale Lyapunov equation (also for Sylvester eqn):

$$A\boldsymbol{X} + \boldsymbol{X}A^\top + BB^\top = 0$$

Projection-type methods

Given a low dimensional approximation space $\mathcal{K}$,

$$\mathbf{X} \approx X_m = V_m Y V_m^T \qquad \mathrm{col}(X_m) \in \mathcal{K}$$

Galerkin condition: $\quad R := AX_m + X_m A^\top + BB^\top \quad \perp \quad \mathcal{K}$

$$V_m^\top R V_m = 0 \qquad\qquad \mathcal{K} = \mathrm{Range}(V_m)$$

Proofs of element-wise decay in Y:

- Standard Krylov (Simoncini '15)
- Rational Krylov (Pozza-Simoncini '19, see also Freitag-Kürschner '20)

Different problems. Similar setting. 2

Large scale Lyapunov equation (also for Sylvester eqn):

$$A\boldsymbol{X} + \boldsymbol{X}A^\top + BB^\top = 0$$

Projection-type methods

Given a low dimensional approximation space $\mathcal{K}$,

$$\mathbf{X} \approx X_m = V_m Y V_m^T \qquad \mathrm{col}(X_m) \in \mathcal{K}$$

Galerkin condition: $\quad R := AX_m + X_m A^\top + BB^\top \quad \perp \quad \mathcal{K}$

$$V_m^\top R V_m = 0 \qquad\qquad \mathcal{K} = \mathrm{Range}(V_m)$$

Proofs of element-wise decay in Y:

- Standard Krylov (Simoncini '15)
- Rational Krylov (Pozza-Simoncini '19, see also Freitag-Kürschner '20)