A subspace–conjugate gradient method for linear matrix equations

Davide Palitta, Martina Iannacito, Valeria Simoncini

The problem

Consider a **multiterm Sylvester equation**

$$\boldsymbol{A}_1\boldsymbol{X}\boldsymbol{B}_1 + \cdots + \boldsymbol{A}_m\boldsymbol{X}\boldsymbol{B}_m = \boldsymbol{C} \qquad (1)$$

that appears in **Control Theory**, **Signal Processing**, **Physical Simulation**, **Uncertainty Quantification**

Finding the solution $\boldsymbol{X}$ of (1) is

- for $m \leq 2$: **easy** (projection methods, ADI, Riemannian optimization methods, etc...)
- for $m > 2$: **more challenging** (matrix–oriented Krylov methods, ad–hoc projection methods, Riemannian optimization)

Matrix–oriented Conjugate Gradient

Assumptions

- $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ are symmetric matrices of size $(n \times n)$
- Right–hand side $\boldsymbol{C} = C_1C_2^\top$ has low rank (s_C) .
- $\mathcal{L}(\boldsymbol{X}) = \sum_{i=1}^m \boldsymbol{A}_i\boldsymbol{X}\boldsymbol{B}_i$ is SPD w.r.t. $\langle \boldsymbol{X}, \boldsymbol{Y}\rangle = \mathrm{trace}(\boldsymbol{X}^\top\boldsymbol{Y})$.

Matrix–oriented CG is equivalent to standard CG, since

$$\sum_{i=1}^m \boldsymbol{A}_i\boldsymbol{X}\boldsymbol{B}_i = \boldsymbol{C} \iff \Big(\sum_{i=1}^m \boldsymbol{B}_i^\top \otimes \boldsymbol{A}_i\Big)\mathrm{vec}(\boldsymbol{X}) = \mathrm{vec}(\boldsymbol{C}).$$

Given an initial guess $\boldsymbol{X}_0 \in \mathbb{R}^{n\times n}$, the matrix–oriented CG iterates are

$$\begin{aligned}
\boldsymbol{X}_{k+1} &= \boldsymbol{X}_k + \alpha_k\boldsymbol{P}_k && \text{where } \alpha_k \in \mathbb{R} && \textit{(solution)}\\
\boldsymbol{R}_{k+1} &= \boldsymbol{C} - \mathcal{L}(\boldsymbol{X}_{k+1}) && && \textit{(residual)}\\
\boldsymbol{P}_{k+1} &= \boldsymbol{R}_{k+1} + \beta_k\boldsymbol{P}_k && \text{where } \beta_k \in \mathbb{R} && \textit{(direction)}
\end{aligned}$$

These matrices are in *factored form*, e.g., $\boldsymbol{P}_k = P_kP_k^\top$, and throughout the iterations, blocks get larger accumulating redundant information.

Low rank truncation ➡
- delayed or stagnating convergence
- challenging to control the rank

Why matrix coefficients?

Matrix–oriented CG

$\alpha_k \in \mathbb{R}$ satisfies the local orthogonality condition

$$\mathrm{vec}(\boldsymbol{R}_k) \perp \mathrm{range}(P_k),$$

where $\mathrm{range}(P_k)$ is a subspace of $\mathbb{R}^{n^2}$ of dimension 1.
The j-th column of $\boldsymbol{X}_{k+1}$ is equal to the j-th column of $\boldsymbol{X}_k$ updated by

$$\boldsymbol{u} = \alpha_k(p_{j,1}\boldsymbol{p}_1 + \cdots + p_{j,s_k}\boldsymbol{p}_{s_k})$$

where $P_k = [\boldsymbol{p}_1, \ldots, \boldsymbol{p}_{s_k}]$ and $P_k(i,j) = p_{ij}$.

Subspace–CG

$\boldsymbol{\alpha}_k \in \mathbb{R}^{s_k\times s_k}$ satisfies the local orthogonality condition

$$\mathrm{vec}(\boldsymbol{R}_k) \perp \mathrm{range}(P_k \otimes P_k) \subseteq \mathbb{R}^{n^2},$$

where $\mathrm{range}(P_k \otimes P_k)$ is a subspace of $\mathbb{R}^{n^2}$ of dimension s_k^2.
The j-th column of $\boldsymbol{X}_{k+1}$ is equal to the j-th column of $\boldsymbol{X}_k$ updated by

$$\Big(\sum_{i=1}^{s_k} \alpha_{1,i}\,p_{j,i}\Big)\boldsymbol{p}_1 + \cdots + \Big(\sum_{i=1}^{s_k} \alpha_{s_k,i}\,p_{j,i}\Big)\boldsymbol{p}_{s_k}$$

where $\boldsymbol{\alpha}_k(i,j) = \alpha_{i,j}$.

Subspace–Conjugate Gradient

Key idea

Replace α, β scalars by $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ matrices

As in matrix–oriented CG, define $\Phi : \mathbb{R}^{n\times n} \to \mathbb{R}$ such that

$$\Phi(\boldsymbol{X}) = \frac{1}{2}\langle \mathcal{L}(\boldsymbol{X}), \boldsymbol{X}\rangle - \langle \boldsymbol{C}, \boldsymbol{X}\rangle,$$

so that $\boldsymbol{X}^\star$, the exact solution of (1), satisfies $\boldsymbol{X}^\star = \arg\min_{\boldsymbol{X}\in\mathbb{R}^{n\times n}} \Phi(\boldsymbol{X})$.
The new iterate for the solution is

$$\boxed{\boldsymbol{X}_{k+1} = \boldsymbol{X}_k + P_k\boldsymbol{\alpha}_kP_k^\top \quad \text{where} \quad \boldsymbol{\alpha}_k \in \mathbb{R}^{s_k\times s_k}.}$$

Let $\phi : \mathbb{R}^{s_k\times s_k} \to \mathbb{R}$ be $\phi(\boldsymbol{\alpha}) = \Phi(\boldsymbol{X}_k + P_k\boldsymbol{\alpha}P_k^\top)$, then

$$\boldsymbol{\alpha}_k = \arg\min_{\boldsymbol{\alpha}\in\mathbb{R}^{s_k\times s_k}} \phi(\boldsymbol{\alpha}) \qquad (2)$$

Proposition

The matrix $\boldsymbol{\alpha}_k \in \mathbb{R}^{s_k\times s_k}$ minimizer of (2) is the unique solution of

$$P_k^\top\mathcal{L}(P_k\boldsymbol{\alpha}P_k^\top)P_k = P_k^\top\boldsymbol{R}_kP_k. \qquad (3)$$

$\boldsymbol{\alpha}_k$ solution of (3) is equivalent to imposing the **local orthogonality condition**

$$\mathrm{vec}(\boldsymbol{R}_{k+1}) \perp \mathrm{range}(P_k \otimes P_k). \qquad (4)$$

The new iterate for the direction matrix is

$$\boxed{\boldsymbol{P}_{k+1} = \boldsymbol{R}_{k+1} + P_k\boldsymbol{\beta}_kP_k^\top \quad \text{where} \quad \boldsymbol{\beta}_k \in \mathbb{R}^{s_k\times s_k}.}$$

Proposition

The matrix $\boldsymbol{\beta}_k \in \mathbb{R}^{s_k\times s_k}$ is obtained imposing the **$\mathcal{L}$–orthogonality condition**

$$\mathrm{vec}\big(\mathcal{L}(\boldsymbol{P}_{k+1})\big) \perp \mathrm{range}(P_k \otimes P_k). \qquad (5)$$

$\boldsymbol{\beta}_k$ satisfying (5) is the unique solution of

$$P_k^\top\mathcal{L}(P_k\boldsymbol{\beta}P_k^\top)P_k = -P_k^\top\mathcal{L}(\boldsymbol{R}_{k+1})P_k. \qquad (6)$$

Notice that (3) and (6) are smaller size multiterm Sylvester equations, that is

$$\sum_{i=1}^m \widetilde{\boldsymbol{A}}_i\boldsymbol{\alpha}\widetilde{\boldsymbol{B}}_i = P_k^\top\boldsymbol{R}_kP_k \quad \text{and} \quad \sum_{i=1}^m \widetilde{\boldsymbol{A}}_i\boldsymbol{\beta}\widetilde{\boldsymbol{B}}_i = -P_k^\top\mathcal{L}(\boldsymbol{R}_{k+1})P_k$$

where $\widetilde{\boldsymbol{A}}_i = P_k^\top\boldsymbol{A}_iP_k$ and $\widetilde{\boldsymbol{B}}_i = P_k^\top\boldsymbol{B}_iP_k$ have size $(s_k \times s_k)$.

Numerical example

Consider the stationary diffusion equation $-\nabla\cdot(\kappa\nabla u) = 0$ in $(0,1)\times(0,1)$ with Dirichlet boundary conditions and semiseparable diffusion coefficient:

$$\kappa(x,y) = \sum_{j=0}^m \delta_j\kappa_{x,j}(x)\kappa_{y,j}(y) = 1 + \sum_{j=1}^{m_k-1}\frac{10^j}{j!}x^jy^j.$$

The resulting multiterm linear equation is

$$\sum_{j=1}^{m_k}\delta_j\Big(\boldsymbol{A}_{j,x}\boldsymbol{X}\boldsymbol{D}_{j,y} + \boldsymbol{D}_{j,x}\boldsymbol{X}\boldsymbol{A}_{j,y}\Big) = \boldsymbol{C},$$

where $\boldsymbol{C}$ has rank 4, $m_k = 4$ and for a total of 8 terms.
Performances of the Subspace–CG (Ss–CG), the Riemannian–nonlinear CG (R–NLCG) [1] and the matrix–oriented truncated preconditioned CG (TPCG) [2, 3] are compared.

n	Precond. type	`maxrank`	R–NLCG	TPCG	Ss–CG determ.	Ss–CG rand.
10000	$\mathcal{P}_1$	20	– (100)	– (100)	– (100)	– (100)
	$\mathcal{P}_1$	40	– (100)	– (100)	1.08 (5)	**0.92** (5)
	$\mathcal{P}_1$	60	– (100)	– (100)	2.47 (5)	**2.34** (5)
	$\mathcal{P}_2$	20	**11.25** (36)	11.42 (38)	– (100)	– (100)
	$\mathcal{P}_2$	40	*42.97 (36)	**15.54** (33)	– (100)	– (100)
	$\mathcal{P}_2$	60	*98.62 (35)	32.39 (28)	9.59 (5)	**8.37** (5)
102400	$\mathcal{P}_1$	20	– (100)	– (100)	– (100)	– (100)
	$\mathcal{P}_1$	40	†	– (100)	18.17 (6)	**8.74** (6)
	$\mathcal{P}_1$	60	†	– (100)	23.50 (5)	**16.93** (5)
	$\mathcal{P}_2$	20	**183.44** (41)	– (100)	– (100)	– (100)
	$\mathcal{P}_2$	40	†	446.94 (47)	– (100)	– (100)
	$\mathcal{P}_2$	60	†	884.20 (26)	115.73 (3)	**101.91** (3)

– no convergence   * Lower final residual norm than other methods   † Out of Memory

Table 1: Running time in seconds, and in parenthesis the number of iterations. Stopping tolerance `tol` $= 5\cdot 10^{-6}$. Truncation tolerance `tolrank`$= 10^{-12}$. $\mathcal{P}_1$: one–term precond, $\mathcal{P}_2$: two–term precond, expensive.

References

[1] I. Bioli, D. Kressner, and L. Robol. Preconditioned Low–Rank Riemannian Optimization for Symmetric Positive Definite Linear Matrix Equations. *SIAM J. Sci. Comput.*, 47(2):A1091–A1116, 2025.

[2] D. Kressner and C. Tobler. Low–Rank Tensor Krylov Subspace Methods for Parametrized Linear Systems. *SIAM J. Matrix Anal. & Appl.*, 32(4):1288–1316, 2011.

[3] V. Simoncini and Y. Hao. Analysis of the Truncated Conjugate Gradient Method for Linear Matrix Equations. *SIAM J. Matrix Anal. & Appl.*, 44(1):359–381, 2023.

Further details

D. Palitta, M. Iannacito, and V. Simoncini. A Subspace–Conjugate Gradient Method for Linear Matrix Equations. *SIAM J. Matrix Anal. & Appl.*, 46(4):2197–2225, 2025.

Finanziato dall'Unione europea
NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA