Proceedings of the Sixteenth International Conference on
Computational Structures Technology
Edited by: P. Iványi, J. Kruis and B.H.V. Topping
Civil-Comp Conferences, Volume 14, Paper 2.1
Civil-Comp Press, Edinburgh, United Kingdom, 2026
ISSN: 2753-3239, doi: 10.4203/ccc.14.2.1

Deflated Krylov Method in Nonlinear Problems of Mechanics and Stochastic Analysis

M. Béreš[1,2] and S. Sysala[1]

[1] Institute of Geonics, Czech Academy of Sciences, Ostrava, Czechia
[2] Faculty of Electrical Engineering and Computer Science, VSB - Technical University of Ostrava, Czechia

Abstract

This paper focuses on the iterative solution of sequences of large-scale linear systems of equations with similar structure arising in nonlinear mechanics and stochastic analysis. To exploit this similarity, we combine preconditioned Krylov methods with deflation and update the deflation basis from previously computed vectors. The efficiency is illustrated in three different applications: the stochastic Darcy-flow systems, the strip-footing elasto-plastic benchmark solved by quasi-Newton/variable preconditioning (QNVP), and slope stability analysis solved by a continuation Newton-like method within the modified shear strength reduction framework. In all cases, deflation significantly reduces total runtime.

Keywords: nonlinear problems in mechanics, finite element method, Newton method, Krylov method, preconditioners, deflation basis

1 Introduction

Linearized systems of equations often repeatedly appear within solution of nonlinear problems in mechanics. For example, they are considered in each iteration of a Newton-type algorithm and in each time or continuation step. In realistic 3D models, these systems are very large and sparse, and direct solvers are often too expensive in memory and runtime.

We focus on sequences of linear systems with similar algebraic structure. In this setting, a black-box iterative solver does not exploit cross-step information and can be inefficient. Our strategy is to use deflated Krylov methods (DKM), where a low-dimensional deflation space is constructed from selected vectors computed in previous linear solves. We further use inexact inner solves, because full inner accuracy is usually unnecessary and may even degrade total efficiency. This solution scheme is broadly applicable. In this paper, we illustrate its efficiency on three different problems.

The paper is organized as follows. Section 2 summarizes the basic ideas of DKM and extends the method to sequences with varying matrices. Section 3 applies DKM to a stochastic Darcy-flow system. Section 4 applies DKM to a quasi-Newton solution of an elasto-plastic problem. Section 5 combines DKM with continuation and Newton-like techniques for slope stability analysis via the modified shear strength reduction (MSSR) method. Section 6 contains concluding remarks.

All computations were carried out using in-house MATLAB codes.

2 Deflated Krylov method for a sequence of linear systems with different matrices

Consider the following system of linear equations:

$$\text{find } \boldsymbol{s} \in \mathbb{R}^n : \quad \boldsymbol{K}\boldsymbol{s} = \boldsymbol{b}, \quad \boldsymbol{K} \in \mathbb{R}^{n\times n},\ \boldsymbol{b} \in \mathbb{R}^n, \tag{1}$$

where $\boldsymbol{K}$ is a sparse symmetric positive definite (SPD) matrix. Preconditioned Krylov methods (PKM) are natural choices for iterative solutions of (1) because they use sparse matrix-vector products and preconditioning rather than matrix factorizations. In this paper, we use two different Krylov methods, the conjugate gradient (CG) method or the flexible generalized minimal residual (FGMRES) method, depending on conditioning and robustness requirements. Next, PKMs are equipped with a preconditioner $\boldsymbol{M}$ that allows to transform the original system into a better-conditioned problem $\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{s} = \boldsymbol{M}^{-1}\boldsymbol{b}$. For general background, we refer to [1].

The *deflated Krylov method* (DKM) augments PKM by a low-dimensional deflation space [2]. This space is represented by a matrix $\boldsymbol{W} \in \mathbb{R}^{n\times m}$, $m \ll n$, having linearly independent columns. Using this matrix, we split the unknown vector $\boldsymbol{s}$ into two parts:

$$\boldsymbol{s} = \boldsymbol{s}_W + \boldsymbol{s}_{W^\perp},$$

where $\boldsymbol{s}_W$, $\boldsymbol{s}_{W^\perp}$ are the projections of $\boldsymbol{s}$ onto $\boldsymbol{W}$, and its orthogonal complement $\boldsymbol{W}^\perp$, respectively, with respect to the scalar product generated by the matrix $\boldsymbol{K}$. In particular,

$$\boldsymbol{W}^\top \boldsymbol{K} \boldsymbol{s}_W = \boldsymbol{W}^\top \boldsymbol{K} \boldsymbol{s} = \boldsymbol{W}^\top \boldsymbol{b},$$

implying

$$\boldsymbol{s}_W = \boldsymbol{W} \left(\boldsymbol{W}^\top \boldsymbol{K} \boldsymbol{W} \right)^{-1} \boldsymbol{W}^\top \boldsymbol{b}.$$

We see that the calculation $\boldsymbol{s}_W$ requires to solve a coarse problem with the matrix $\boldsymbol{W}^\top \boldsymbol{K} \boldsymbol{W} \in \mathbb{R}^{m \times m}$. Since $m \ll n$, the coarse solve is very cheap. The second part, $\boldsymbol{s}_{W^\perp}$, satisfies

$$\boldsymbol{s}_{W^\perp} = \boldsymbol{s} - \boldsymbol{s}_W = \boldsymbol{P} \boldsymbol{s}, \quad \text{where } \boldsymbol{P} = \boldsymbol{I} - \boldsymbol{W} \left(\boldsymbol{W}^\top \boldsymbol{K} \boldsymbol{W} \right)^{-1} \boldsymbol{W}^\top \boldsymbol{K}.$$

Hence, $\boldsymbol{s}_{W^\perp}$ can be found using a projected version of PKM with respect to the projection matrix $\boldsymbol{P}$.

Inner Krylov iterations for finding $\boldsymbol{s}_{W^\perp}$ are stopped by a relative residual tolerance $\epsilon > 0$. In Sections 4–5, relatively large ϵ is shown to be sufficient when DKM is embedded in Newton-like outer loops.

It is well known that DKM is useful when a small number of eigenvalues of $\boldsymbol{K}$ are close to the origin, or when solving systems with multiple right-hand sides [2]. In this paper, we also illustrate that DKM is efficient for sequences of systems with varying matrices:

$$\text{find } \boldsymbol{s}_i \in \mathbb{R}^n : \quad \boldsymbol{K}_i \boldsymbol{s}_i = \boldsymbol{b}_i, \quad i = 0, 1, 2, \ldots, \tag{2}$$

when the matrices $\boldsymbol{K}_0, \boldsymbol{K}_1, \ldots$ are assumed to be similar. The corresponding deflation spaces are updated iteratively:

$$\boldsymbol{W}_0 = \emptyset, \qquad \boldsymbol{W}_i = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_{m_i}) \in \mathbb{R}^{n \times m_i},$$

where $0 \le m_1 \le m_2 \le \cdots \le m_i \ll n$, and the column vectors $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{m_i}$ depend on previously computed solutions $\boldsymbol{s}_0, \boldsymbol{s}_1, \ldots, \boldsymbol{s}_{i-1}$ of (2). It is convenient to choose an SPD reference matrix $\boldsymbol{K}_{el}$ and keep the basis vectors $\boldsymbol{K}_{el}$-orthonormal:

$$\boldsymbol{w}_i^\top \boldsymbol{K}_{el} \boldsymbol{w}_j = \delta_{ij}.$$

If $\boldsymbol{K}_i = \boldsymbol{K}_{el}$ for any i, then $\boldsymbol{W}_i^\top \boldsymbol{K}_i \boldsymbol{W}_i$ is diagonal and the corresponding coarse solves are especially cheap. The particular construction of $\boldsymbol{W}_1, \boldsymbol{W}_2, \ldots$ and the choices of $\boldsymbol{K}_{el}$ and PKM are problem dependent, as illustrated in the following sections.

3 Deflation efficiency on stochastic Darcy-flow systems

First, we apply DKM to the stochastic finite element method used to solve elliptic partial differential equations. Here, the sequence (2) arises during reduced-basis construction for the stochastic finite element discretization. Each system $\boldsymbol{K}_i \boldsymbol{s}_i = \boldsymbol{b}_i$

corresponds to one deterministic counterpart of the stochastic problem. The index i represents a set of randomly generated material parameters, and thus the deterministic systems have naturally similar structures.

For illustration, we consider the stochastic stationary Darcy flow problem

$$-\operatorname{div}_x\big(k(x,Z)\nabla_x u(x,Z)\big) = 0$$

studied in [3] and solved in the square domain $D = \langle 0,1\rangle^2$. Here, u denotes the unknown pore pressure field, which is fixed to 0 on the left side and to 1 on the right side of the unit square, while the top and bottom sides are impermeable. Next, k denotes an uncertain permeability field. It is assumed that k is piecewise constant in subdomains $D_1, D_2 \ldots, D_5 \subset D$ and has the form

$$k(x,Z_m) = \exp(\sigma Z_m + \mu) \quad \text{in } D_m,\ m = 1,2,\ldots,5,$$

with independent standard Gaussian variables Z_m, $\sigma = 0.3$, and $\mu = 0$. The domain partition is obtained on a uniform 100×100 grid by thresholding a sampled Gaussian random field, which yields the five subdomains; see Figure 1.

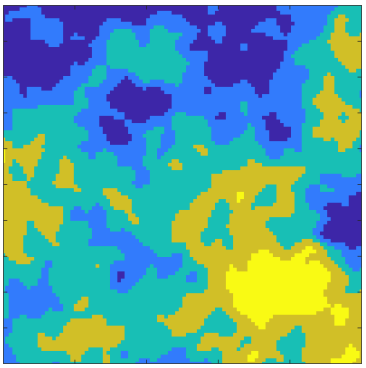

Figure 1: Example of the domain partition used in the reported stochastic Darcy-flow benchmark.

Within this benchmark, 43 inner linear systems is solved in sequence during the reduced-basis construction. Since the corresponding matrices $\boldsymbol{K}_i$ are uniformly SPD, the CG Krylov method is chosen. Next, three preconditioners are compared: additive Schwarz with 30 vertical subdomains, incomplete Cholesky without fill, and diagonal (Jacobi). The inner accuracy is set $\epsilon = 10^{-6}$.

In the deflated CG (DCG) method, the columns of the matrix $\boldsymbol{W}_i$, $i = 1,2,\ldots$, are constructed by $\boldsymbol{K}_{el}$-orthogonalization of the solutions $\boldsymbol{s}_0, \boldsymbol{s}_1, \ldots, \boldsymbol{s}_{i-1}$ defined in (2). Here, $\boldsymbol{K}_{el} = \boldsymbol{K}_0$ is the finite-element stiffness matrix associated with the mean permeability field.

Table 1 reports cumulative DCG savings (in runtime) versus CG over the whole 43-system sequence. We see that the savings are significant for all investigated preconditioners.

Table 1: Cumulative DCG savings [%] versus CG for the representative stochastic Darcy-flow run with 43 solved inner systems (data from Table 6.4 in [3]).

Schwarz	**ichol(0)**	**Diagonal**
80.2%	81.4%	82.5%

The numbers of iterations are compared in Figure 2. The DCG curves decrease along the sequence as the deflation space grows.

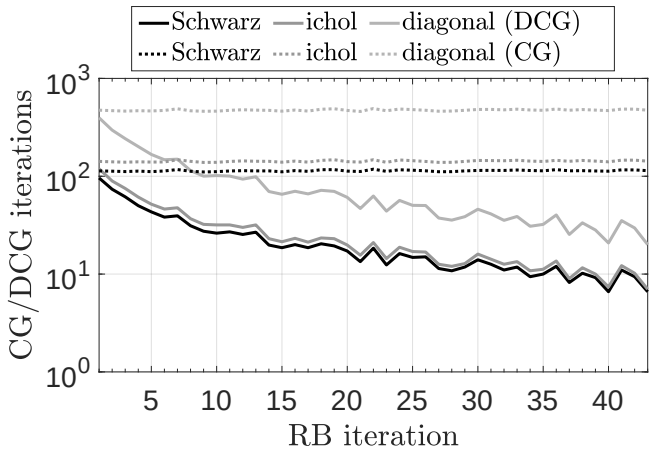

Figure 2: Iteration counts in the 43 consecutive inner systems for all tested preconditioners in the same representative run.

This numerical experiment confirms that, for sequences of related SPD systems, deflation removes a dominant part of Krylov iterations. This behavior is consistent with the two other applications analyzed in the next sections.

4 Quasi-Newton methods in elasto-plasticity

Newton-like methods are a frequent choice for solving elasto-plastic problems. A standard variant of the Newton method uses tangent stiffness matrices and has local superlinear convergence. However, the corresponding linearized systems can be ill-conditioned, for example, in perfect plasticity or plasticity with softening. This may cause numerical difficulties. In addition, repeated tangent assembly may be expensive and also technically difficult, as elasto-plastic constitutive operators are often implicit.

Approximating tangent matrices leads to methods of a quasi-Newton type. Their local convergence is usually slower (often only linear) but more robust. We show that these methods can be highly competitive in total runtime with the standard Newton method when they are combined with DKM. To this end, we consider the so-called *quasi-Newton/variable preconditioning* (QNVP) method, established in [4, 5] and recently extended in [6, 7] to nonlinear elasticity and elasto-plasticity.

Consider the following system of nonlinear equations:

$$\text{find } \boldsymbol{u} \in \mathbb{R}^n : \quad F(\boldsymbol{u}) = \boldsymbol{0}, \tag{3}$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$ is monotone, Lipschitz continuous, and semismooth. These properties are typical for a broad class of elasto-plastic problems with associated flow rules. Hence, the tangent stiffness matrix $\nabla F(\boldsymbol{v})$ exists for almost all $\boldsymbol{v} \in \mathbb{R}^n$ and is symmetric positive semidefinite.

QNVP iterations for the problem (3) read

$$\boldsymbol{u}_{i+1} = \boldsymbol{u}_i + \omega_i \boldsymbol{s}_i, \qquad \boldsymbol{K}_i \boldsymbol{s}_i = -F(\boldsymbol{u}_i), \qquad i = 0, 1, 2, \ldots. \tag{4}$$

Here, $\boldsymbol{u}_0 \in \mathbb{R}^n$ is given, $\boldsymbol{K}_i \in \mathbb{R}^{n \times n}$ is a SPD matrix such that there exist positive constants $0 < m_i \leq M_i$ satisfying the following bounds:

$$m_i \boldsymbol{v}^\top \boldsymbol{K}_i \boldsymbol{v} \leq \boldsymbol{v}^\top (\nabla F(\boldsymbol{u}_i)) \boldsymbol{v} \leq M_i \boldsymbol{v}^\top \boldsymbol{K}_i \boldsymbol{v} \quad \forall \boldsymbol{v} \in \mathbb{R}^n. \tag{5}$$

Finally, we set $\omega_i = 2/(M_i + m_i)$. This algorithm has linear convergence under standard assumptions saying that the solution u exists, $\boldsymbol{u}_0$ is sufficiently close to $\boldsymbol{u}$, and $\nabla F(\boldsymbol{v})$ is positive definite in a neighborhood of $\boldsymbol{u}$, see [7]. Additionally, if $m_i, M_i \to 1$ then the convergence is superlinear. The Newton method is recovered by $\boldsymbol{K}_i = \nabla F(\boldsymbol{u}_i)$ and $m_i = M_i = 1$. The QNVP method can also be modified by replacing $\omega_i = 2/(M_i + m_i)$ with a line-search strategy, enabling global convergence analysis. For brevity, this modification is not considered in the numerical experiments below.

Two QNVP variants are considered:

- **QNVP-1:** $\boldsymbol{K}_i$ is an elastic matrix with variable moduli, updated to approximate $\nabla F(\boldsymbol{u}_i)$ as closely as possible. For the von Mises model, the variable moduli E_i, ν_i and the corresponding constants m_i, M_i were derived in [7]. An extension to other plastic criteria is possible. If the relative modulus changes are very small, we freeze the matrix to save assembly time [7]. Within the DKM approach proposed in Section 2, we choose $\boldsymbol{K}_{el}$ as the elastic stiffness matrix for given elastic moduli E, ν. The columns of the deflation matrix $\boldsymbol{W}_i$, $i = 1, 2, \ldots$, are constructed by $\boldsymbol{K}_{el}$-orthogonalization of the directions $\boldsymbol{s}_0, \boldsymbol{s}_1, \ldots, \boldsymbol{s}_{i-1}$ defined in (4). The choice of PKM depends on a used plastic model and is thus specified below.

- **QNVP-2:** $\boldsymbol{K}_i = \boldsymbol{K}_{el}$ for all i. We set $\boldsymbol{W}_i = [\boldsymbol{A}_1, \ldots, \boldsymbol{A}_{i-1}]$, where $\boldsymbol{A}_i \in \mathbb{R}^{m_i \times n}$ is a matrix representation of the Krylov subspace generated when computing $\boldsymbol{s}_i$. The deflation matrix $\boldsymbol{W}_i$ is then much larger than for QNVP-1. We can allow it because its columns are $\boldsymbol{K}_{el}$-orthogonal and, consequently, the matrices $\boldsymbol{W}_i^\top \boldsymbol{K}_{el} \boldsymbol{W}_i$ are diagonal and thus directly invertible.

For reference, we also report the Newton method with deflation. We use the same DKM approach as in the case of QNVP-1.

As a numerical illustration, we use the 3D strip-footing benchmark (Figure 3): a $10 \times 10 \times 10$ m soil block with a loaded area 1×1 m, zero normal displacement on vertical faces and bottom, and pressure $\gamma_3 = -450$ Pa in the x_3 direction. The loaded area is positioned at a corner, which is justified by symmetry with respect to the x_1 and x_2 axes. Volume forces are neglected.

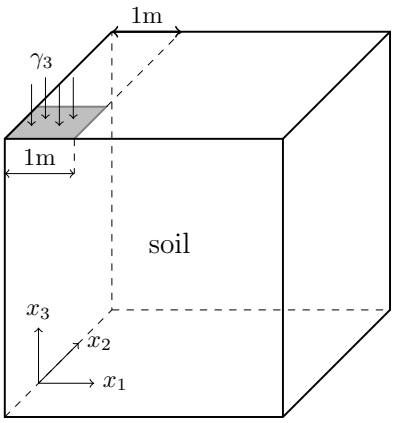

Figure 3: Geometry of the strip-footing problem.

The von Mises elasto-plastic model with isotropic hardening [7] is used with the following material parameters: $E = 10$ MPa, $\nu_0 = 0.48$, $Y = 100$ Pa, and $\alpha \in (0, 1]$. Here, α is a hardening parameter such that the value $\alpha = 1$ corresponds to perfect plasticity. We compare the results for $\alpha = 0.3, 0.5, 0.9$ on a uniform P1 mesh with 5,241,478 unknowns.

For the investigated values of α, the stiffness matrices are uniformly positive definite, therefore, the CG method is chosen within PKM. It is completed with a preconditioner based on a combination of separate displacement decomposition and aggregation-based algebraic multigrid [7]. The inner tolerance (introduced in Section 2) is set to $\epsilon = 10^{-1}$ for QNVP-1 and QNVP-2, and $\epsilon = 10^{-4}$ for Newton. These values are justified by the analysis and experiments in [7].

The outer stopping criterion is

$$\boldsymbol{u}_0 := \mathbf{0}, \qquad \|F(\boldsymbol{u}_i)\| \leq 10^{-K}\|F(\boldsymbol{u}_0)\|, \quad K = 10. \tag{6}$$

This relatively strict outer tolerance is more favorable for Newton (superlinear local rate) than for QNVP (linear local rate).

The Newton, QNVP-1, and QNVP-2 algorithms are compared for different values of the hardening parameter in Table 2. We see that the Newton method is almost insensitive to α, while both QNVP variants are strongly affected by the nonlinearity level. For milder nonlinearity ($\alpha = 0.3, 0.5$), QNVP is faster in runtime despite more outer iterations, which indicates that deflated inner solves are very effective. For stronger nonlinearity ($\alpha = 0.9$), Newton is fastest. However, runtime differences are not so significant as the differences in outer iterations.

Table 2: Comparison of outer (Newton-like) iterations / cumulative DKM iterations / runtime [s] for different values of α. The best runtime for each α is highlighted in bold.

α	**Newton**	**QNVP-1**	**QNVP-2**
0.3	6/305/194.3	12/195/**141.0**	16/192/146.6
0.5	6/313/192.5	15/222/**174.1**	22/221/175.3
0.9	7/460/**257.8**	49/385/442.9	115/531/511.3

5 Shear strength reduction method in slope stability

This section summarizes the use of DKM in 3D slope stability analysis. We follow the finite element formulation, Mohr–Coulomb plasticity, Davis-type modification of the shear strength reduction (MSSR) method, and continuation/Newton-like algorithms from [8].

The MSSR algebraic problem reads

$$\text{given } \lambda \geq \lambda_0, \text{ find } \boldsymbol{u}_\lambda \in \mathbb{R}^n : \quad F_\lambda(\boldsymbol{u}_\lambda) = \boldsymbol{b}, \tag{7}$$

$$\lambda^* = \max\{\lambda \geq \lambda_0 : \text{ (7) has a solution}\}. \tag{8}$$

The parameter λ reduces cohesion c and friction angle ϕ. It also modifies the dilatancy angle ψ to be the parameterized problem associated. The minimal value λ_0 of λ is usually chosen from the interval $(0, 1]$. The aim is to find the critical value λ^* representing the factor of safety (FoS). Equation (7) is a discrete version of the parameterized Mohr–Coulomb elastic-perfectly plastic problem. The mapping $F_\lambda : \mathbb{R}^n \to \mathbb{R}^n$ is monotone, Lipschitz continuous and semismooth, and $\nabla F_\lambda(\boldsymbol{v})$ is symmetric positive semidefinite.

A direct monotone increase of λ is possible but often unstable in practice, as (7) has no solution for $\lambda > \lambda^*$ and the stiffness matrices are ill-conditioned in the vicinity of λ^*. Therefore, we use the indirect continuation method proposed in [8], see Algorithm 1.

Algorithm 1 Indirect continuation for MSSR

Initialization: choose $\lambda_0 > 0$, $\delta\omega_0 > 0$, $\beta > 0$, $k = 0$
solve $F_{\lambda_0}(\boldsymbol{u}_0) = \boldsymbol{b}$; set $\omega_0 = \boldsymbol{b}^\top \boldsymbol{u}_0$
repeat
    set $k = k + 1$
    set $\omega_k = \omega_{k-1} + \delta\omega_{k-1}$
    find $(\boldsymbol{u}_k, \lambda_k)$ from $F_{\lambda_k}(\boldsymbol{u}_k) = \boldsymbol{b}$, $\boldsymbol{b}^\top \boldsymbol{u}_k = \omega_k$
    set $\delta\omega_k = \delta\omega_{k-1}$
    if $k \geq 2$ and $\lambda_k - \lambda_{k-1} \leq \beta(\lambda_{k-1} - \lambda_{k-2})$, set $\delta\omega_k = 2\delta\omega_k$
until stopping criterion is reached
set $\lambda^* = \lambda_k$

We see that increases in λ are controlled indirectly, using a variable ω representing the work of external forces, and the extended system defined in Line 6 of Algorithm 1 is solved instead of the original system (7). In particular, the computed sequence $\{\lambda_k\}$ is increasing and tends to λ^* as $\omega_k \to +\infty$, see [8]. The extended system is solved by Algorithm 2.

Algorithm 2 Newton-like solver for the extended system

1: *Initialization:* choose $\lambda_k^0 \geq 0$, $\boldsymbol{u}_k^0 \in \mathbb{R}^n$ with $\boldsymbol{b}^\top \boldsymbol{u}_k^0 = \omega_k$, $i = 0$
2: **repeat**
3: find $\boldsymbol{v}_k^i, \boldsymbol{z}_k^i \in \mathbb{R}^n$: $\boldsymbol{K}_k^i \boldsymbol{v}_k^i = \boldsymbol{b} - F_{\lambda_k^i}(\boldsymbol{u}_k^i)$, $\boldsymbol{K}_k^i \boldsymbol{z}_k^i = -\boldsymbol{G}_k^i$
4: set $\delta\lambda_k^i = -\boldsymbol{b}^\top \boldsymbol{v}_k^i / \boldsymbol{b}^\top \boldsymbol{z}_k^i$, $\boldsymbol{s}_k^i = \boldsymbol{v}_k^i + \delta\lambda_k^i \boldsymbol{z}_k^i$
5: choose $\alpha_k^i \in (0, 1]$ such that $\|F_{\lambda_k^i + \alpha_k^i \delta\lambda_k^i}(\boldsymbol{u}_k^i + \alpha_k^i \boldsymbol{s}_k^i) - \boldsymbol{b}\| < \|F_{\lambda_k^i}(\boldsymbol{u}_k^i) - \boldsymbol{b}\|$
6: set $\boldsymbol{u}_k^{i+1} = \boldsymbol{u}_k^i + \alpha_k^i \boldsymbol{s}_k^i$, $\lambda_k^{i+1} = \lambda_k^i + \alpha_k^i \delta\lambda_k^i$, $i = i + 1$
7: **until** $\|F_{\lambda_k^{i+1}}(\boldsymbol{u}_k^{i+1}) - \boldsymbol{b}\| \leq 10^{-4}\|\boldsymbol{b}\|$
8: set $\boldsymbol{u}_k = \boldsymbol{u}_k^{i+1}$, $\lambda_k = \lambda_k^{i+1}$

Here

$$\boldsymbol{K}_k^i = (1 - \varrho_k^i)\nabla F_{\lambda_k^i}(\boldsymbol{u}_k^i) + \varrho_k^i \boldsymbol{K}_{el}, \qquad \boldsymbol{G}_k^i = \frac{dF_\lambda(\boldsymbol{u}_k^i)}{d\lambda}\Big|_{\lambda=\lambda_k^i},$$

where $\boldsymbol{K}_{el}$ is the elastic stiffness matrix used to regularize the tangent stiffness matrix $\nabla F_{\lambda_k^i}(\boldsymbol{u}_k^i)$, as it is ill-conditioned for λ_k^i close to λ^*. The regularization parameter $\varrho_k^i \in (0, 1)$ is initiated with $\varrho_k^0 = 10^{-4}$. Then, its value can be increased if the line search in line 5 becomes too restrictive, i.e., if α_k^i is close to zero.

We see that the following systems of linear equations are repeatedly solved in each continuation step (the index k) and in each Newton iteration (the index i):

$$\boldsymbol{K}_k^i \boldsymbol{v}_k^i = \boldsymbol{b} - F_{\lambda_k^i}(\boldsymbol{u}_k^i), \qquad \boldsymbol{K}_k^i \boldsymbol{z}_k^i = -\boldsymbol{G}_k^i.$$

Because $\boldsymbol{K}_k^i$ may be ill-conditioned (despite the regularization), we use FGMRES with a preconditioner based on a separate displacement decomposition and aggregation-based algebraic multigrid.

Deflation matrices $\boldsymbol{W}_k^i$ are constructed by $\boldsymbol{K}_{el}$-orthonormalization of selected history vectors. At the beginning of the continuation step k, we construct $\boldsymbol{W}_k^0$ using continuation solutions $\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{k-1}$ from Algorithm 1. During Newton iterations, the matrices $\boldsymbol{W}_k^i$, $i = 1, 2, \ldots$, are incrementally enriched by the correction vectors $\boldsymbol{v}_k^0, \ldots, \boldsymbol{v}_k^{i-1}$ from Algorithm 2. These correction vectors are discarded for the next continuation step, i.e., they do not influence the construction of $\boldsymbol{W}_{k+1}^0$. We complete the suggested deflation strategy with the following remarks.

- For the initialization solve $F_{\lambda_0}(\boldsymbol{u}_0) = \boldsymbol{b}$, the default deflation basis is empty ($\boldsymbol{W}_0^0 = \emptyset$), because there are no previously computed vectors. The vector $\boldsymbol{u}_0$

is available for deflation only after this solve is completed. If the initialization solve is performed by Newton iterations, the deflation starts from an empty basis and can be activated only after the first correction vectors are computed.

- Vectors $\boldsymbol{z}_k^i$ from Algorithm 2 are not used for the construction of $\boldsymbol{W}_k^i$. In our experiments, the auxiliary system $\boldsymbol{K}_k^i \boldsymbol{z}_k^i = -\boldsymbol{G}_k^i$ is usually solved in one or two FGMRES iterations with the chosen preconditioner. Therefore, including $\boldsymbol{z}_k^i$ in the deflation basis gives only small additional gains, while increasing basis size and projection cost [8].

- Removing the correction vectors $\boldsymbol{v}_k^0, \ldots, \boldsymbol{v}_k^{i-1}$ from the deflation basis for the $(k+1)$-th continuation step is a practical compromise for performance and numerical stability. In exact arithmetic, accumulating all historical Newton corrections would enlarge the deflation space and could only improve Krylov iteration counts. However, in finite precision, a larger coarse space may increase projection/coarse-solve overhead and deteriorate numerical robustness, so a compact evolving basis is preferred in runtime-critical computations [8].

To illustrate the efficiency of the DKM approach in slope stability, we consider the numerical test with a homogeneous 3D slope, see Figure 4. Its inclination is $45°$, height 10 m, and thickness 5 m. Zero normal displacements are prescribed on the vertical faces and at the bottom. The remaining boundary is traction-free. The specific slope weight is $\gamma = 20\,\text{kN/m}^3$. The Mohr-Coulomb model is considered with the following material parameters: $E = 40{,}000\,\text{kPa}$, $\nu = 0.3$, $\phi = 45°$, $\psi = 0°$, and $c = 6\,\text{kPa}$. A uniform P2 mesh with 15,770 elements is used.

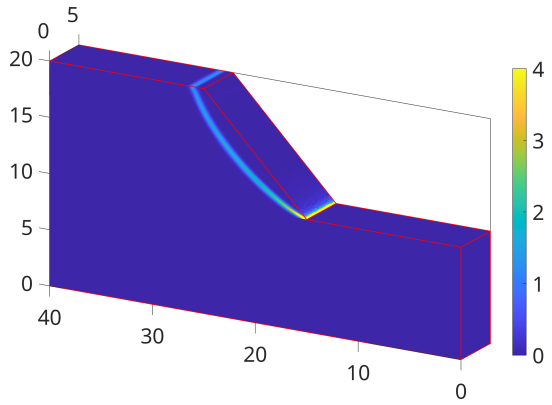

Figure 4: Geometry of the slope and computed failure zones.

The parameter ω in Algorithm 1 is increased up to $\omega_{\max} = 3000$ kNm. The continuation curve $\omega_k \mapsto \lambda_k$ is shown in Figure 5 for $k = 1, 2, \ldots, 15$. The computed factor of safety is $\lambda^* \approx \lambda_{15} \approx 1.3$. The corresponding failure mechanism is depicted in Figure 4.

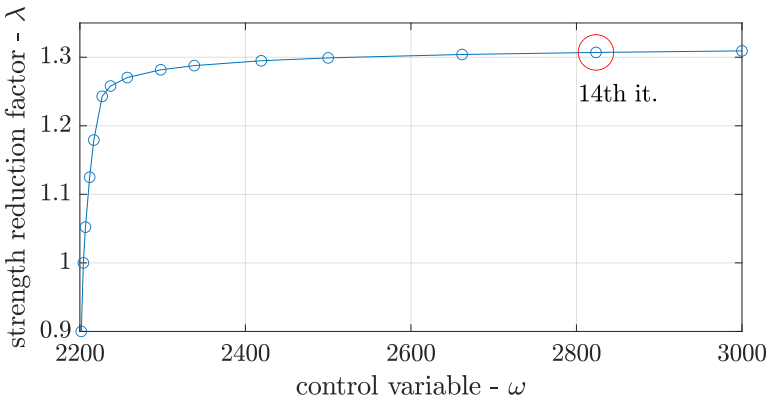

Figure 5: Computed continuation curve for MSSR (unit kNm for ω).

The total number of Newton iterations, the cumulative counts of linearized iterations, and the overall runtime are compared in Table 3 for the FGMRES method with and without deflation and for various inner tolerances ϵ (see Section 2). We also consider a direct solver for linearized systems. The results confirm major gains from deflation: deflated FGMRES is substantially faster than non-deflated FGMRES and the direct baseline. For the best setting ($\epsilon = 10^{-1}$), the non-deflated/deflated ratio is 2.81 in cumulative linear iterations and 1.87 in runtime, while the direct/deflated runtime ratio is 4.13.

Table 3: Solution efficiency vs. inner solver tolerance ϵ for FGMRES with and without deflation (Newton tolerance 10^{-4}). A direct MATLAB solver is also included.

Solver	ϵ	**Newton it.**	**Linear it.**	**Runtime [s]**
Deflated FGMRES	0.2	164	2307	203.54
	10^{-1}	145	2933	195.97
	10^{-2}	164	7908	299.86
	10^{-3}	167	16169	473.29
FGMRES	0.2	346	11389	617.54
	10^{-1}	203	8240	367.02
	10^{-2}	205	26364	809.75
	10^{-3}	154	37214	1034.62
Direct solver	–	164	–	809.37

Notice that fewer total Newton iterations (145) are required for deflated FGMRES with $\epsilon = 10^{-1}$ than for the direct solver baseline (164). This surprising result is partially explained in Figure 6, where the convergence of the Newton method in the 14-th continuation step is compared for deflated FGMRES with different ϵ. We see that $\epsilon = 10^{-1}$ needs 10 Newton iterations versus 43 for $\epsilon = 10^{-3}$. This behavior is not uniform over all continuation steps. In our experiments, it is most visible in selected

steps near the onset of plasticity-dominated response, whereas later steps may show weaker or no such effect. The precise mechanism has not been analyzed yet.

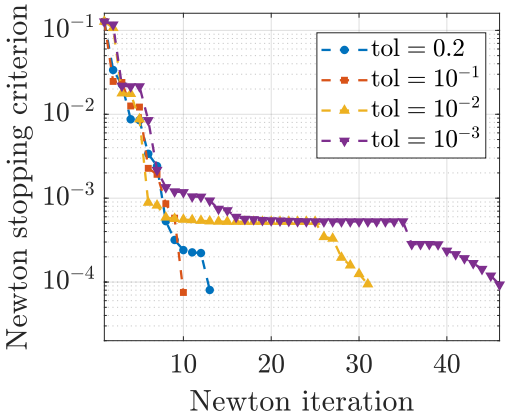

Figure 6: Convergence of the Newton solver in the 14th continuation step for deflated FGMRES and different tolerances ϵ.

6 Conclusion

This paper focused on deflated Krylov methods for sequences of linear systems. We proposed to update the deflation spaces using solutions from previous steps. The efficiency and universality of this idea were shown in three different benchmarks from mechanics and stochastic analysis. Namely, we solved stochastic Darcy-flow systems, the strip-footing elasto-plastic problem by the QNVP method, and the MSSR-based slope stability problem by continuation and Newton-like iterations. The three benchmarks gave the following quantitative results.

For the Darcy-flow benchmark, deflated CG saved more than 80% of cumulative Krylov iterations. For the tested preconditioners, the savings were between 80.2% and 82.5%. Since the Krylov iterations are the dominant cost in this setting, runtime is reduced by about the same percentage.

For the strip-footing benchmark, the Newton and QNVP methods were compared for three different values of the hardening parameter, $\alpha = 0.3, 0.5$ (mild nonlinearity), and $\alpha = 0.9$ (high nonlinearity). The linear-iteration ratios (Newton over QNVP-1) are 1.56, 1.41, and 1.19. The runtime ratios (Newton over QNVP-1) are 1.38, 1.11, and 0.58. Hence, for $\alpha = 0.9$, QNVP-1 still uses fewer cumulative linear iterations (385 vs. 460), but the strong increase in outer iterations (49 vs. 7), together with additional overhead outside linear iterations, leads to higher total runtime.

For the MSSR slope-stability benchmark, the best inner tolerance setting is $\epsilon = 10^{-1}$. For this setting, the non-deflated/deflated ratio is 2.81 in cumulative linear iterations and 1.87 in runtime, and the direct/deflated runtime ratio is 4.13.

In general, these benchmarks indicate that deflation may help in a broader class of problems in which similar linear systems are solved repeatedly in sequence.

Acknowledgements

This work was supported by the Czech Science Foundation under project No. 26-20457S.

References

[1] Yousef Saad. *Numerical Methods for Large Eigenvalue Problems: Revised Edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2011. DOI: 10.1137/1.9781611970739.

[2] Y. Saad, M. Yeung, J. Erhel, and F. Guyomarc'h. A deflated version of the conjugate gradient algorithm. *SIAM Journal on Scientific Computing*, 21(5):1909–1926, 2000. DOI: 10.1137/S1064827598339761.

[3] M. Béreš. *Methods for the Solution of Differential Equations with Uncertainties in Parameters*. Dissertation thesis, VŠB–Technical University of Ostrava, Ostrava, Czech Republic, 2023.

[4] I. Faragó and J. Karátson. *Numerical Solution of Nonlinear Elliptic Problems via Preconditioning Operators: Theory and Applications*, volume 11 of *Advances in Computation*. Nova Science Publishers, New York, 2002.

[5] J. Karátson and I. Faragó. Variable preconditioning via quasi-newton methods for nonlinear problems in hilbert space. *SIAM Journal on Numerical Analysis*, 41(4):1242–1262, 2003. DOI: 10.1137/S0036142901399613.

[6] J. Karátson, S. Sysala, and M. Béreš. Quasi-newton variable preconditioning for nonlinear elasticity systems in 3d. *Numerical Linear Algebra with Applications*, 31(3):e2537, 2024. DOI: 10.1002/nla.2537.

[7] J. Karátson, S. Sysala, and M. Béreš. Quasi-newton iterative solution approaches for nonsmooth elliptic operators with applications to elasto-plasticity. *Computers and Mathematics with Applications*, 178:61–80, 2025. DOI: 10.1016/j.camwa.2024.11.022.

[8] S. Sysala, M. Béreš, S. Bérešová, T. Luber, and Z. Michalec. Advanced continuation and iterative methods for slope stability analysis in 3d. *Computers & Structures*, 315:107842, 2025. DOI: 10.1016/j.compstruc.2025.107842.