CSMA 2026
17ème Colloque National en Calcul des Structures
18-22 Mai 2026, Presqu'île de Giens (Var)

Linearly Preconditioned Substructured RASPEN for Structural Mechanics

E.M. Ettaouchi[1,2], N. Tardieu[1], L. Giraud[3], C. Kruse[2]

[1] *EDF R&D, Bd Gaspard Monge, Palaiseau, 91120, France. el-mehdi.ettaouchi@edf.fr, nicolas.tardieu@edf.fr*
[2] *Cerfacs, 42 Av Gaspard Coriolis, Toulouse, 31100, France. carola.kruse@cerfacs.fr*
[3] *Inria, 200 Av. de la Vieille Tour 33405 Talence. luc.giraud@inria.fr*

Abstract — This paper introduces an efficient nonlinear domain decomposition approach for structural mechanics based on the SRASPEN method. By explicitly constructing the Jacobian, the method leverages classical linear preconditioners and Krylov subspace recycling to improve scalability and reduce computational cost. Numerical tests on a elasto-plastic beam show significant convergence gains over standard SRASPEN.
Mots clés — Domain Decomposition, Newton Method, Krylov Method.

1 Introduction

In many physical applications, nonlinear partial differential equations (PDEs) pose significant challenges for classical solvers, including second-order methods like Newton's method. To accelerate convergence, nonlinear preconditioning techniques have been developed, inspired by their linear counterparts [1, 2]. Nonlinear preconditioning modifies the original system into a more favorable system, improving both the nonlinear behavior and the conditioning of the Jacobian, facilitating faster convergence. The pioneering ASPIN method [3] introduced nonlinear preconditioning via domain decomposition. RASPEN [4] improved upon ASPIN by using a restricted additive Schwarz strategy, known for better robustness and sharper convergence.

Substructured RASPEN (SRAPSEN) [5] enhances RASPEN by reducing the problem to a substructured space, focusing on interface unknowns. This leads to smaller linear systems and thus reduced memory usage and computational cost. Convergence analysis as well as implementation details are provided in [6].

The SRASPEN approach shows very efficient on small numbers of subdomains but, as is well known, the need of a second level preconditioner becomes obvious as this number increases. Two-level RASPEN-like is an active topic of research and analogous strategies to those used in linear multigrid methods to construct an effective coarse subspace [7, 8] have been investigated. The coarse problem can be integrated into nonlinear solvers in two main ways: through nonlinear subspace iteration, where the coarse subspace is treated as a subdomain and iteratively refined; or via the Full Approximation Scheme (FAS) [9], a nonlinear extension of classical multigrid techniques. In any case, it comes at the expense of solving a new non-linear problem. Although it is much smaller than the original one, it must be solved at each iteration, which significantly increases the computational cost.

To circumvent this drawback, we propose in this paper evaluating the use of a classical linear preconditioner in conjunction with recycling techniques or algebraic coarse corrections, which may improve both scalability and numerical efficiency.

The paper is structured as follows: the first section provides a detailed introduction to the SRASPEN method; the second section presents an efficient strategy for constructing the iteration matrix, enabling the use of linear preconditioners; finally, the last section discusses implementation aspects and provides numerical illustrations.

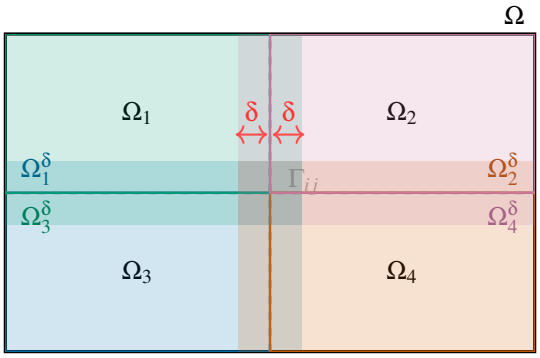

Figure 1: Nonoverlapping subdomains $\{\Omega_i\}$ (solid lines) and their overlapping extensions $\{\Omega_i^\delta\}$ (transparent halos). Interfaces Γ_{ij} shown dashed.

2 The SRASPEN nonlinear solver

The RASPEN (Restricted Additive Schwarz Preconditioned Exact Newton) method is a nonlinear domain decomposition solver designed to solve large-scale nonlinear systems arising from the discretization of partial differential equations (PDEs). It combines Newton's method with a domain decomposition strategy using the Restricted Additive Schwarz (RAS) method as a nonlinear preconditioner.

Consider a nonlinear system of equations $F(u) = 0, \quad u \in \mathbb{R}^n$ where $F : \mathbb{R}^n \to \mathbb{R}^n$ is a nonlinear operator resulting from the discretization of a PDE.

2.1 Nonlinear additive Schwarz methods

Relying on classical concept and notation of linear domain decomposition, we define the nonlinear additive Schwarz preconditioner. For this, we introduce first the concept of subdomain solution denoted by $G_i(u)$. It consist in a nonlinear solve performed on a subdomain Ω_i and thus returning the local solution u_i. $G_i(u)$ is fully defined by:

$$R_i F\left(P_i G_i(u) + (I_n - P_i R_i)u\right) = 0_{\mathbb{R}^{n_i}}, \tag{1}$$

with $R_i = P_i^T$ being the Boolean restriction operator form the global domain to the overlapping subdomain Ω_i. A precise analysis of G_i indicates that it does not depend on all values of u but rather on the ghost nodes of the subdomain Ω_i. Thus, G_i is considered to be a function from $\mathbb{R}^{n_{\Gamma_i}}$ to $\mathbb{R}^{n_i}$ and we reformulate its definition in Equation (1) as follows:

$$R_i F\left(P_i G_i\left(R_{\Gamma_i} u\right) + (I_n - P_i R_i)u\right) = 0_{\mathbb{R}^{n_i}},$$

with R_{Γ_i} being the Boolean restriction operator form the global domain to the boundary Γ_i of Ω_i. Let us define the subdomain correction $C_i(u)$ in the subsequent manner:

$$C_i(u) = R_i u - G_i\left(R_{\Gamma_i} u\right),$$

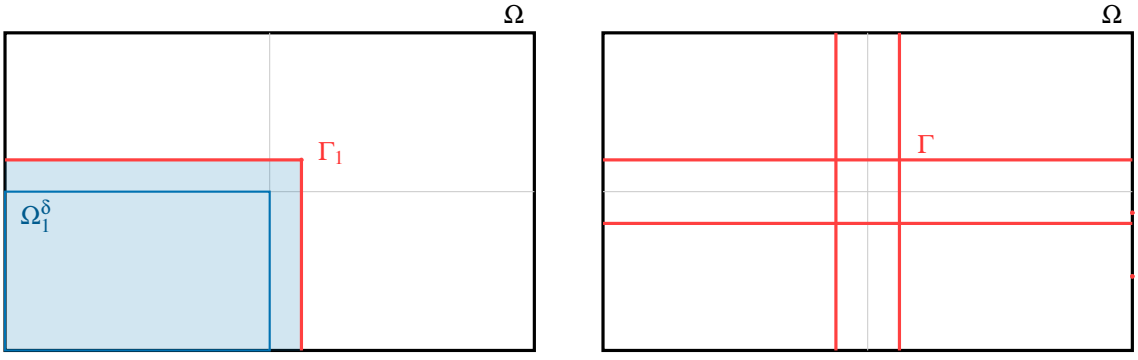

Figure 2: Local interface Γ_i^{loc} (left) and substructure $\Gamma = \bigcup_i \Gamma_i^{\text{loc}}$ (right)

so that the nonlinear subspace iteration now writes:

$$R_i F\left(u - P_i C_i\left(u\right)\right) = 0_{\mathbb{R}^{n_i}}. \tag{2}$$

Thus, the restricted Schwarz preconditioner $\mathcal{P}$, defined in [4], can be written as follows:

$$\mathcal{P}\left(u\right) = u - \sum_{i\in[\![1,N]\!]} \widehat{P}_i C_i\left(u\right) = \sum_{i\in[\![1,N]\!]} \widehat{P}_i G_i\left(R_{\Gamma_i} u\right), \tag{3}$$

with $\widehat{P}_i = \tilde{P}_i \tilde{R}_i P_i$ and $\tilde{R}_i = \tilde{P}_i^T$ being the Boolean restriction operator form the global domain to the non-overlapping subdomain Ω_i. The operator $\widehat{P}_i$ corresponds to prolongating values of an overlapping subdomain to the global domain while setting the overlap values to zero. When a subdomain is subjected to the solution values as boundary conditions, we find the exact values on its interior, i.e., $\forall i \in [\![1,N]\!],\ G_i\left(R_{\Gamma_i} u^*\right) = R_i u^* \Leftrightarrow C_i\left(u^*\right) = 0_{\mathbb{R}^{n_i}}$. Hence, u^* is a fixed point of the preconditioner $\mathcal{P}$ with:

$$\mathcal{P}\left(u^*\right) = u^*. \tag{4}$$

The action of the preconditioner can be used as solver through the following fixed point iteration:

$$u_{j+1} = \mathcal{P}\left(u_j\right). \tag{5}$$

However, this is not a robust solver as it often diverges or converges slowly. In the upcoming section, we introduce an accelerated version of the iteration (5) through the Newton method as an outer solver.

2.2 Preconditioned Newton methods

Another way to find the solution u^* is to solve Equation (4) by the Newton method, which leads us to define the preconditioned function $\mathcal{F}$ as its corresponding residual:

$$\mathcal{F}\left(u\right) = u - \mathcal{P}\left(u\right) = \sum_{i\in[\![1,N]\!]} \widehat{P}_i C_i\left(u\right). \tag{6}$$

Denoting by $\mathcal{J}$ the Jacobian associated with $\mathcal{F}$, its expression would be given as follows:

$$\mathcal{J}\left(u\right) = I_n - d\mathcal{P}\left(u\right) = \sum_{i\in[\![1,N]\!]} \widehat{P}_i dC_i\left(u\right).$$

$d\mathcal{P}$ and dC_i denotes respectively the preconditioner and the subdomain correction derivatives. Differentiating Equation (2) with respect to u results in the following expression:

$$dC_i\left(u\right) = \left(R_i J\left(u\right) P_i\right)^{-1} R_i J\left(u\right). \tag{7}$$

This leads us to the final Jacobian of the preconditioned nonlinear system:

$$\mathcal{J}\left(u\right) = \sum_{i\in[\![1,N]\!]} \widehat{P}_i \left(R_i J\left(u - P_i C_i\left(u\right)\right) P_i\right)^{-1} R_i J\left(u - P_i C_i\left(u\right)\right). \tag{8}$$

Now, that the exact Jacobian of the preconditioned function is known and computable, we can define its corresponding Newton method iteration as follows:

$$u_{j+1} = u_j - \left(\mathcal{J}\left(u_j\right)\right)^{-1} \mathcal{F}\left(u_j\right). \tag{9}$$

The method based on the iteration (9) is called *Restricted Additive Schwarz Preconditioned Exact Newton* (RASPEN). A more optimized version of this method is called substructured RASPEN (SRASPEN) [5] which works only on the substructure. Its associated preconditioned function that we denote $\mathcal{F}_\Gamma : \mathbb{R}^{n_\Gamma} \longrightarrow \mathbb{R}^{n_\Gamma}$ is defined from the original function as

$$\mathcal{F}_\Gamma\left(v\right) = R_\Gamma \mathcal{F}\left(P_\Gamma v\right), \forall v \in \mathbb{R}^{n_\Gamma}. \tag{10}$$

Its corresponding Jacobian denoted $\mathcal{J}_\Gamma$ obtained by differentiating Equation (10) yielding the following expression:

$$\mathcal{J}_\Gamma(v) = R_\Gamma \mathcal{J}(P_\Gamma v) P_\Gamma.$$

Hence, the Newton iteration derived from this substructured version of RASPEN is given by

$$v_{j+1} = v_j - \left(\mathcal{J}_\Gamma(v_j)\right)^{-1} \mathcal{F}_\Gamma(v_j). \tag{11}$$

In [5], the authors proved the equivalence of the SRASPEN iterates (11) and those of RASPEN on the substructure when initialized with the same values:

$$v_0 = R_\Gamma u_0 \implies v_j = R_\Gamma u_j, \quad \forall j \in \mathbb{N}. \tag{12}$$

This result is mainly based on the following property of the preconditioned Jacobian which can be easily deduced from differentiating Equation (3):

$$\left(I_n - \mathcal{J}(u)\right)\left(I_n - P_\Gamma R_\Gamma\right) = 0_{\mathbb{R}^{n\times n}}, \quad \forall u \in U. \tag{13}$$

The previous equation has a simple interpretation: it says that the preconditioned Jacobian has only non-zero off-diagonal terms on the ghosts nodes for any position $u \in U$.

The substructured approach significantly reduces the cost of solving the global linear system at each Newton nonlinear iteration, as $P_\Gamma v$ is a prolongation by zeros outside the substructure. However, this leads to a poor starting point for the nonlinear local problems, resulting in additional steps that increase computational effort. These extra steps may offset the savings achieved in solving the nonlinear systems. To address this issue, an inexpensive adjustment can be incorporated into the method to recover the RASPEN values within the interior of the subdomains (see [6]).

2.3 Limitations and prospects

As mentioned in the introduction, while the SRASPEN method demonstrates strong performance with a limited number of subdomains, its scalability is hindered as this number increases, necessitating a second-level preconditioner, which significantly raise computational costs. To address this limitation, we explore in the sequel the use of classical linear preconditioners combined with recycling techniques.

3 Explicit construction of the Jacobian matrix

As with many iterative methods, we do not work with an explicit matrix; instead, we work with an operator whose effect on a vector is known. This is clearly the case for RASPEN. Nevertheless, based on numerous experiments, we have noticed that, through preconditioned by nonlinear solvers, the solution to equation (9) can lead to hundreds of iterations of the outer Krylov method. This guided us towards the idea that building the explicit matrix A_{SRAS} could be computationally feasible and beneficial. This is topic of what follows.

3.1 Expression of the Jacobian matrix

To simplify the notation and emphasize the connection with the linear case, we adopt the following notation:

$$\begin{aligned} A &= J(u - P_i C_i(u)) \\ A_i &= R_i J(u - P_i C_i(u)) P_i \end{aligned}$$

Then, (8) can be rewritten in the more common form:

$$\mathcal{J}(u) = A_{\text{RAS}} = \sum_{i=1}^{n} \tilde{R}_i^T A_i^{-1} R_i A$$

We then use the following trick:

$$\begin{aligned} A_{\text{RAS}} &= \sum_{i=1}^{n} \tilde{R}_i^T A_i^{-1} R_i A \left(I_n + R_i^T R_i - R_i^T R_i\right), \\ &= \sum_{i=1}^{n} \tilde{R}_i^T \underbrace{A_i^{-1} R_i A R_i^T}_{I_{n_i}} R_i + \tilde{R}_i^T A_i^{-1} R_i A \left(I_n - R_i^T R_i\right), \\ &= I_n + \sum_{i=1}^{n} \tilde{R}_i^T A_i^{-1} R_i A \left(I_n - R_i^T R_i\right). \end{aligned}$$

We shall now have a closer look at the effect of the $R_i A \left(I_n - R_i^T R_i\right)$ operator on a given global vector u. First $u_1 = \left(I_n - R_i^T R_i\right) u$ has the same shape as u but with the internal dofs in Ω_i set to zero. Then $b_1 = A u_1$ is the associated residual in the whole domain Ω. Namely the internal forces in Ω_i are recovered. Then b_1 is restricted in Ω_i by $R_i b_1$.
Given that R_{Γ_i} is the trace of Ω_i, the effect of $R_i A \left(I_n - R_i^T R_i\right) u$ is equivalently expressed as:

$$R_i A \left(I_n - R_i^T R_i\right) = R_i A R_{\Gamma_i}^T R_{\Gamma_i}.$$

We now introduce the rectangular matrix $A_{\Gamma_i} = R_i A R_{\Gamma_i}$ of size $n_i \times n_{\Gamma_i}$, so that the Jacobian $\mathcal{J}(u)$ now expresses:

$$A_{\text{RAS}} = I_n + \sum_{i=1}^{n} \tilde{R}_i^T A_i^{-1} A_{\Gamma_i} R_{\Gamma_i} \tag{14}$$

This expression reveals the structure of the RASPEN preconditioned Jacobian. It has a block structure, with the identity on the diagonal blocks of each subdomain, and non-zero off-diagonal blocks between subdomain sharing at least one node. It also emphasizes the role of the substructure in connecting the subdomains.

3.2 The substructured matrix

In order to further exploit the properties of the substructure, we define the substructured Jacobian matrix as follows:

$$A_{\text{SRAS}} = R_\Gamma A_{\text{RAS}} R_\Gamma^T.$$

Let us now show how to exploit the substructure approach regarding the solution x of the linearized system for a right-hand side b.

$$\begin{aligned} A_{\text{RAS}} x = b, \implies & A_{\text{RAS}} R_\Gamma^T x^{(s)} + A_{\text{RAS}} \left(I_n - R_\Gamma^T R_\Gamma\right) x = b, \\ \implies & \left(I_n - R_\Gamma^T R_\Gamma\right) x = b - A_{\text{RAS}} R_\Gamma^T A_{\text{SRAS}}^{-1} b^{(s)}, \\ \implies & x = R_\Gamma^T x^{(s)} + \left(I_n - R_\Gamma^T R_\Gamma\right) x = b + R_\Gamma^T A_{\text{SRAS}}^{-1} b^{(s)} - A_{\text{RAS}} R_\Gamma^T A_{\text{SRAS}}^{-1} b^{(s)}, \\ \implies & x = b + \left(I_n - A_{\text{RAS}}\right) R_\Gamma^T A_{\text{SRAS}}^{-1} b^{(s)}, \end{aligned}$$

$$\boxed{x = \left(I_n + \left(I_n - A_{\text{RAS}}\right) R_\Gamma^T A_{\text{SRAS}}^{-1} R_\Gamma\right) b.} \tag{15}$$

Hence, the inverse of A_{RAS} is $\left(I_n + \left(I_n - A_{\text{RAS}}\right) R_\Gamma^T A_{\text{SRAS}}^{-1} R_\Gamma\right)$. It includes the solution of a linear system involving A_{SRAS}, whose relatively small size is expected to yield numerical efficiency. This is the topic of the following section.

3.3 The block sparsity pattern of A_{SRAS}

We now study the profile of the substructured Jacobian matrix. To this end, we introduce the notion of inner local interface as the traces of the substructure in the local subdomains *i.e.* $\Gamma_i' = \Omega_i' \bigcap \Gamma$. Let us now define the intersections between the outer local interfaces $\left(\Gamma_i\right)_{1 \le i \le N}$ and the inner local interfaces $\left(\Gamma_i'\right)_{1 \le i \le N}$,

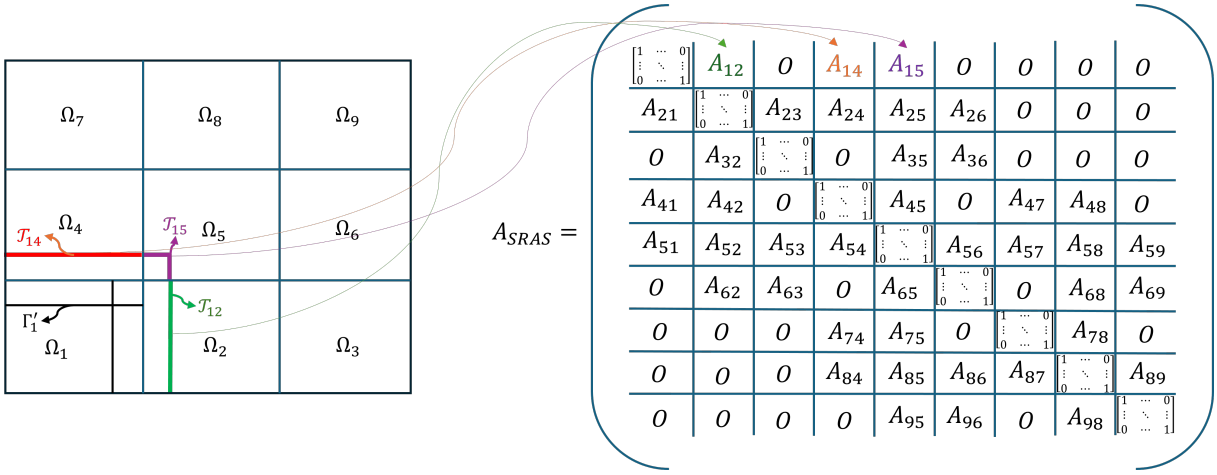

Figure 3: Jacobian sparsity pattern

denoted $\mathcal{T}_{i,j} = \Gamma_i' \cap \Gamma_j$. With this in place, we define the neighbors indexing for the i-th subdomain as follows:

$$(i_k)_{k\in[\![0,r_i]\!]} \subset [\![1,N]\!], \quad \text{such that,} \quad \mathcal{T}_{i,i_k} \neq \emptyset, \quad \forall k \in [\![0,r_i]\!].$$

This means that only the r_i subdomains $(i_k)_{k\in[\![0,r_i]\!]}$ whose internal ghost dofs $\left(\Gamma_{i_k}'\right)_{k\in[\![0,r_i]\!]}$, influence the solution of the subdomain i. This result helps figuring out the sparsity pattern of A_{SRAS}, which is illustrated in Figure 3. It is also of major importance to emphasize that the off-diagonal blocks themselves are also sparse.

3.4 Building procedure

The efficiency of the proposed method strongly depends on the efficiency of the construction process of the matrix A_{SRAS}. Based on previous insights, we designed a naturally parallel algorithm in which all off-diagonal blocks are assembled concurrently. Specifically, we compute only the necessary portion of the inverse A_i^{-1} required for the matrix-matrix product $A_i^{-1}A_{\Gamma_i}R_{\Gamma_i}$, as used in equation (14).

Regarding the term A_{15} of Figure 3, these entries are the DOFs of the ghosts of the subdomain 1 that reside in subdomain 5. As can be noticed, only a small portion of the inverse will be computed, using multiple right-hand sides. These features are provided by the MUMPS solver [10] and they lead to tremendous gains in our situation.

4 Numerical experiments

4.1 Implementation

We have implemented the SRASPEN method in code_aster [11] mainly using the petsc4py library [12]. It is available in the code since version 17.3.16. The nonlinear local problems are solved using the Trust-Region Newton method [13]. Regarding the global problem, it is solved on the substructure by a classical Newton method without line-search. The Jacobian of the first nonlinear iteration of each load step is explicitly built using the proposed strategy and used for linearly preconditioning the successive solves of the linearized system. The MUMPS direct solver is used for the solution of local linearized systems.

We now turn our attention to the solution of (15), which lies at the core of the method. Several algorithms are evaluated to speed-up the solution of the linearized global system.

- Given the DD color of the approach, we have first set our sights on the Additive Schwarz method as a preconditioner of the GMRes Krylov method [14]. An overlap of 1 was used and a modified Gramm-Schmidt algorithm with no restart was used for GMRes.

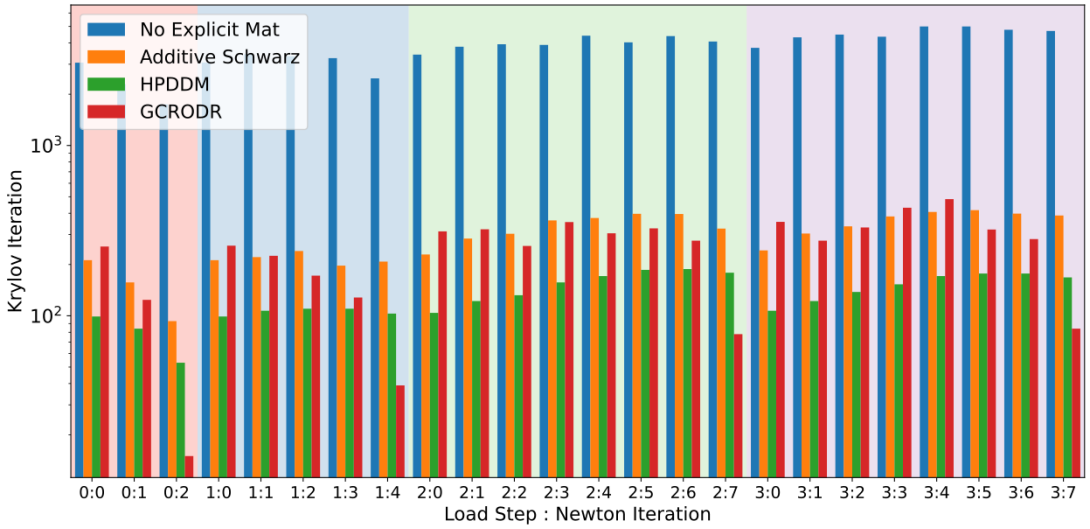

Figure 4: Convergence of various preconditioners

- Since successive solves are run with the same preconditioner, we have chosen to replace GMRes with GCRO-DR [15] in order to evaluate the benefit of Krylov subspace recycling. 10 harmonic Ritz vectors are used for deflation.
- In order to garanty the scalability of the approach, an algebraic spectral coarse problem is built using the strategy introduced in [16] available in the HPPDM library [17].

4.2 Physical problem

We consider a mechanically straightforward yet computationally demanding problem involving a beam of length $L = 300$ mm and a square cross-section of $10 \times 10\,\text{cm}^2$, rigidly fixed at both ends and subjected on its top face to a uniform pressure load of $F = 1.6\,\text{MPa}$. The mesh is composed of 20 nodes Serendipity elements, with 8 points integration scheme (aka underintegrated). The constitutive equation is a viscous elasto-plasticity with isotropic hardening, with parameters being $E = 195000$ MPa, $\nu = 0.3$, $\sigma^Y = 181$ MPa, $H = 1930$ MPa, $\sigma_0 = 6176$ MPa, $\varepsilon_0 = 3.3\text{e}13$ and $m = 6.76$. According to the hyperbolic sine model, the accumulated plastic strain-rate is given by $\dot{p} = \dot{\varepsilon}_0 \left[\sinh\left(\frac{<\Phi_p>}{\sigma_0}\right)\right]^m$ with Φ_p being the plastic threshold. The nonlinear problem involves approximately 5 million degrees of freedom and is distributed across 64 subdomains along the axis of the beam. This decomposition imposes considerable challenges on the scalability of the solution algorithm.

4.3 Numerical results

The total load is applied to the beam in four steps. Using a classical Newton–Krylov–RAS algorithm, divergence occurs during the fourth step, which necessitates subdividing it into four sub-steps to achieve convergence. In contrast, employing SRASPEN enables convergence within the original four load steps, demonstrating the superior efficiency of this approach.
Figure 4 shows the Krylov iteration counts required for convergence, plotted on a logarithmic scale along the y-axis. Different background colors distinguish the various load steps, and within each step, the iteration count is reported for every Newton iteration. The blue bars represent the classical SRASPEN approach without explicit Jacobian construction. The advantage of the proposed method is evident with the simple addition of an Additive Schwarz preconditioner. Furthermore, GCRO-DR provides a substantial reduction in iterations, particularly during the early stages of the simulation when plasticity remains localized and the linear systems evolve gradually. Finally, incorporating a second-level correction via HPDDM proves highly effective, achieving an additional 30% reduction in iteration count compared to Additive Schwarz.

5 Conclusion

The proposed method enhances the SRASPEN framework by introducing an explicit Jacobian construction, which allows the integration of efficient linear preconditioners and recycling techniques. This approach mitigates the scalability limitations of SRASPEN when dealing with a large number of subdomains. Numerical results confirm that even a simple additive Schwarz preconditioner provides noticeable improvements, while Krylov recycling and coarse level further accelerate the convergence. Future work will focus on exploring its application to more complex nonlinear problems.

References

[1] Martin J. Gander. On the origins of linear and non-linear preconditioning. In *Domain Decomposition Methods in Science and Engineering XXIII*, pages 153–161. Springer International Publishing, 2017.

[2] Peter R. Brune, Matthew G. Knepley, Barry F. Smith, and Xuemin Tu. Composing scalable nonlinear algebraic solvers. *SIAM Review*, 57(4):535–565, 2015.

[3] Xiao-chuan Cai and David Keyes. Nonlinearly preconditioned inexact Newton algorithms. *SIAM Journal on Scientific Computing*, 24, 08 2001.

[4] V. Dolean, M. J. Gander, W. Kheriji, F. Kwok, and R. Masson. Nonlinear preconditioning: How to use a nonlinear schwarz method to precondition newton's method. *SIAM Journal on Scientific Computing*, 38(6):A3357–A3380, 2016.

[5] Faycal Chaouqui, Martin J. Gander, Pratik M. Kumbhar, and Tommaso Vanzan. Linear and nonlinear substructured restricted additive Schwarz iterations and preconditioning. *Numerical Algorithms*, 91(1):81–107, 2022.

[6] Ettaouchi El Mehdi, Luc Giraud, Carola Kruse, and Nicolas Tardieu. Convergence analysis of overlapping domain decomposition preconditioners for nonlinear problems. *SIAM Journal on Scientific Computing*, 47(6):A3464–A3493, 2025.

[7] Alexander Heinlein, Axel Klawonn, Jascha Knepper, and Oliver Rheinbach. Multiscale coarse spaces for overlapping Schwarz methods based on the acms space in 2d. *ETNA - Electronic Transactions on Numerical Analysis*, 48:156–182, 01 2018.

[8] Feng-Nan Hwang and Xiao-Chuan Cai. A class of parallel two-level nonlinear Schwarz preconditioned inexact Newton algorithms. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1603–1611, 2007. Domain Decomposition Methods: recent advances and new challenges in engineering.

[9] Achi Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, 31(138):333–390, 1977.

[10] P.R. Amestoy, I. S. Duff, J. Koster, and J.-Y. L'Excellent. A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications*, 23(1):15–41, 2001.

[11] Electricité de France. Finite element *code_aster*, analysis of structures and thermomechanics for studies and research,. Open source on www.code-aster.org.

[12] Lisandro D. Dalcin, Rodrigo R. Paz, Pablo A. Kler, and Alejandro Cosimo. Parallel distributed computing using python. *Advances in Water Resources*, 34(9):1124–1139, 2011. New Computational Methods and Software Tools.

[13] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media. 2006.

[14] Youcef Saad and Martin H. Schultz. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, July 1986.

[15] Michael L. Parks, Eric de Sturler, Greg Mackey, Duane D. Johnson, and Spandan Maiti. Recycling Krylov subspaces for sequences of linear systems. *SIAM Journal on Scientific Computing*, 28(5):1651–1674, 2006.

[16] Hussam Al Daas, Pierre Jolivet, and Tyrone Rees. Efficient algebraic two-level Schwarz preconditioner for sparse matrices. *SIAM Journal on Scientific Computing*, 45(3):A1199–A1213, 2023.

[17] Pierre Jolivet, Jose E. Roman, and Stefano Zampini. KSPHPDDM and PCHPDDM: Extending PETSc with advanced krylov methods and robust multilevel overlapping Schwarz preconditioners. *Computers & Mathematics with Applications*, 84:277–295, 2021.