arXiv:1706.10221v1 [math.NA] 30 Jun 2017

Weakly intrusive low-rank approximation method for nonlinear parameter-dependent equations

Loic Giraldi* and Anthony Nouy†‡

Abstract

This paper presents a weakly intrusive strategy for computing a low-rank approximation of the solution of a system of nonlinear parameter-dependent equations. The proposed strategy relies on a Newton-like iterative solver which only requires evaluations of the residual of the parameter-dependent equation and of a preconditioner (such as the differential of the residual) for instances of the parameters independently. The algorithm provides an approximation of the set of solutions associated with a possibly large number of instances of the parameters, with a computational complexity which can be orders of magnitude lower than when using the same Newton-like solver for all instances of the parameters. The reduction of complexity requires efficient strategies for obtaining low-rank approximations of the residual, of the preconditioner, and of the increment at each iteration of the algorithm. For the approximation of the residual and the preconditioner, weakly intrusive variants of the empirical interpolation method are introduced, which require evaluations of entries of the residual and the preconditioner. Then, an approximation of the increment is obtained by using a greedy algorithm for low-rank approximation, and a low-rank approximation of the iterate is finally obtained by using a truncated singular value decomposition. When the preconditioner is the differential of the residual, the proposed algorithm is interpreted as an inexact Newton solver for which a detailed convergence analysis is provided. Numerical examples illustrate the efficiency of the method.

Keywords: model order reduction, non-intrusive, low-rank approximation, inexact Newton solver, empirical interpolation method, singular value decomposition.

*Division of Computer, Electrical and Mathematical Sciences and Engineering, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia. E-mail: loic.giraldi@kaust.edu.sa.

†Department of Computer Science and Mathematics, Ecole Centrale de Nantes, Nantes, France. Email: anthony.nouy@ec-nantes.fr.

‡This research was supported by the French National Research Agency (grant ANR CHORUS MONU-0005).

1 Introduction

The purpose of this paper is to propose weakly intrusive variants of model order reduction methods for the efficient solution of a system of nonlinear equations

$$R(u(\xi);\xi) = 0 \tag{1}$$

whose solution $u(\xi) \in \mathbb{R}^N$ depends on parameters ξ taking values in a finite set Ξ. The parameter-dependent solution is assumed to admit an accurate approximation of the form

$$u(\xi) \approx \sum_{i=1}^{r} v_i \lambda_i(\xi),$$

where the set of parameter-independent vectors $v_1, \ldots, v_r$ constitutes a reduced basis in $\mathbb{R}^N$. When identifying u with a tensor in $\mathbb{R}^N \otimes \mathbb{R}^\Xi$, this can be interpreted as a rank-r approximation of u. Model order reduction methods are usually classified as intrusive if numerical codes for parameter-independent equations can not be used as pure black-boxes. In [12] and [13], the authors consider the solution of stochastic nonlinear equations with a stochastic Galerkin method, usually qualified as intrusive. The notion of intrusiveness was relaxed by allowing the access to pointwise evaluations of the residual of the equation, therefore resulting in a non (or say weakly) intrusive implementation of stochastic Galerkin methods. Here, we adopt a similar point of view.

We assume that we have a numerical code which for a given instance of ξ generates a sequence of approximations $(u^k(\xi))_{k\geq 1}$ converging to $u(\xi)$, and we further assume that we have access to more or less detailed information from this numerical code. More precisely, we consider Newton-type iterations

$$u^{k+1}(\xi) = u^k(\xi) + P(u^k(\xi);\xi)^{-1} R(u^k(\xi);\xi), \tag{2}$$

and we assume that we have access to evaluations of the residual $R(u^k(\xi);\xi) \in \mathbb{R}^N$ and the preconditioner $P(u^k(\xi);\xi) \in \mathbb{R}^{N\times N}$ (such as the differential of the residual), or some of their entries. A classical approach consists in using the iterative algorithm (2) for each instance of ξ independently. Here, we formally apply the iterative algorithm for all values of ξ simultaneously and introduce an additional truncation step in order to generate a sequence of low-rank iterates. The approach is similar to truncated iterative methods introduced for the solution for tensor-structured linear equations in [1, 2, 11, 14]. The resulting algorithm takes the form

$$u^{k+1}(\cdot) = \Pi_\varepsilon(u^k(\cdot) + \widetilde{P}(u^k(\cdot);\cdot)^{-1} \widetilde{R}(u^k(\cdot);\cdot)), \tag{3}$$

where $\widetilde{R}(u^k(\cdot);\cdot)$ and $\widetilde{P}(u^k(\cdot);\cdot)$ are low-rank approximations of $R(u^k(\cdot);\cdot)$ and $P(u^k(\cdot);\cdot)$, and where Π_ε is a truncation operator such that $\Pi_\varepsilon(v)$ provides a low-rank approximation of a function v with a controlled precision ε. The algorithm provides a low-rank approximation

of the solution of the parameter-dependent equation without using any snapshot of the solution. Assuming that the residual and the preconditioner admit accurate approximations with a low rank, a limited information on these quantities (i.e. a small number of evaluations of their entries) is sufficient to construct these approximations, which can yield a significant reduction of complexity when compared to a classical approach. Here, we rely on variants of the empirical interpolation method (EIM) [3] for the construction of these approximations.

In contrast to [15, 7, 16], the result of the method is not a reduced order model which is then evaluated in an online phase, but an approximation of the solution of a possibly large set of samples. However, if the samples are a set of quadrature points, interpolation points, or random samples, standard integration, interpolation or least-squares methods can then be used to obtain a representation of the parameter-dependent solution in a suitable approximation format.

The paper is organized as follows. In Section 2, we consider the approximation of the residual in the particular case where for a given $v(\xi)$ with low-rank representation, we have a partial knowledge on the low-rank representation of the residual $R(v(\xi);\xi)$. In this case, we introduce a variation of the approach proposed in [6] in order to compute a low-rank representation of this residual with a rigorous control of the error. The same approach can be used for obtaining an approximation of the preconditioner using a partial knowledge on its low-rank representation. In Section 3, we consider the approximation of the residual and the preconditioner without a priori knowledge on their representations as parameter-dependent algebraic quantities, and we propose an approximation method which requires simple evaluations of entries of these quantities. The approach relies on the EIM for vector- or matrix-valued parameter-dependent functions (see e.g. [16] for the matrix-valued case), and includes a statistical control of the error. Note that the proposed approach differs from the discrete EIM proposed in [7] in that it does not require the evaluations of samples of the solution to compute a reduced basis for its low-rank representation, and it includes a rigorous control of the error. In Section 4, we introduce a greedy rank-one algorithm (see e.g. [5, 9]) for computing an approximation of $\widetilde{P}(u^k(\xi);\xi)^{-1}\widetilde{R}(u^k(\xi);\xi)$ which exploits the low-rank structure of the operator $\widetilde{P}(u^k(\xi);\xi)$ and right-hand side $\widetilde{R}(u^k(\xi);\xi)$. In Section 5, we present the Newton-like truncated solver and we analyze its convergence in the particular case of a standard Newton truncated solver, which is interpreted as an inexact Newton algorithm [8]. In Section 6, numerical examples illustrate the efficiency of the method.

2 Approximation of residual and preconditioner with partially known low-rank structure

In this section, we consider the approximation of the residual $R(\xi) := R(u(\xi);\xi)$ and of the preconditioner $P(\xi) := P(u(\xi);\xi)$ for a given $u(\xi)$. It is assumed that when $u(\xi)$ admits a

given representation of the form $u(\xi) = \sum_{i=1}^{m} v_i \lambda_i(\xi)$, the residual and the preconditioner also admit representations of the form

$$R(\xi) = \sum_{i=1}^{s} g_i \gamma_i(\xi) \quad \text{and} \quad P(\xi) = \sum_{i=1}^{p} F_i \phi_i(\xi), \tag{4}$$

where the vectors $g_i \in \mathbb{R}^N$ and matrices $F_i \in \mathbb{R}^{N \times N}$ are not known but where the real-valued functions $\gamma_i(\xi)$ and $\phi_i(\xi)$ are known.

We follow [6] in order to construct an approximation of $R(\xi)$ and $P(\xi)$ based on the knowledge of $\gamma(\xi) := (\gamma_i(\xi))_{i=1}^s$ and $\phi(\xi) := (\phi_i(\xi))_{i=1}^p$ and a minimal number of evaluations of $R(\xi)$ and $P(\xi)$ at some suitable points in Ξ. Note that the knowledge of vectors $\{g_i\}_{i=1}^s$ and matrices $\{F_i\}_{i=1}^p$ is not required, hence this *weakly intrusive* denomination. Here, the novelty lies in a rigorous control of the error. The strategy is presented for the approximation of the residual. The application to the approximation of the preconditioner is straightforward.

Let us assume that an interpolation $\mathcal{I}_r[\gamma](\xi)$ of $\gamma(\xi)$ is available in the form

$$\mathcal{I}_r[\gamma](\xi) = \sum_{j=1}^{r} \gamma(\xi_j^\star) \alpha_j(\xi), \tag{5}$$

where the $\xi_j^\star$ are some interpolation points in Ξ and the $\alpha_j(\xi)$ are real-valued functions satisfying the interpolation property

$$\alpha_j(\xi_i^\star) = \delta_{i,j} \quad \text{for all} \quad 1 \leq i, j \leq r. \tag{6}$$

We then obtain an approximation $\mathcal{I}_r[R](\xi)$ of the residual $R(\xi)$ of the form

$$\mathcal{I}_r[R](\xi) = \sum_{i=1}^{s} g_i \mathcal{I}_r[\gamma]_i(\xi) = \sum_{i=1}^{s} g_i \sum_{j=1}^{r} \gamma_i(\xi_j^\star) \alpha_j(\xi) = \sum_{j=1}^{r} R(\xi_j^\star) \alpha_j(\xi),$$

which is an interpolation of $R(\xi)$ at points $\{\xi_j^\star\}_{j=1}^r$. Let $\|\cdot\|$ be a norm in $\mathbb{R}^N$ associated with an inner product $\langle \cdot, \cdot \rangle$. The interpolation error on the residual is

$$\|R(\xi) - \mathcal{I}_r[R](\xi)\| = \|\gamma(\xi) - \mathcal{I}_r[\gamma](\xi)\|_W, \tag{7}$$

where $W = (\langle g_i, g_j \rangle)_{1 \leq i,j \leq s} \in \mathbb{R}^{s \times s}$ is the Gram matrix of the set of vectors $\{g_i\}_{i=1}^s$, and $\|\cdot\|_W$ is the semi-norm in $\mathbb{R}^s$ induced by W, defined by $\|x\|_W^2 = x^T W x$. Therefore, in order to obtain a sharp control of the error of interpolation of $R(\xi)$, the error of interpolation of γ has to be controlled with respect to the semi-norm $\|\cdot\|_W$ and not the standard Euclidean norm in $\mathbb{R}^s$. We will then propose a mean to compute the Gram matrix W with less than r evaluations of the residual $R(\xi)$, and an empirical interpolation method for the construction of an interpolation $\mathcal{I}_r[\gamma]$ controlled with respect to the semi-norm $\|\cdot\|_W$.

2.1 Computation of the Gram matrix

Let $\Xi = \{\xi_k\}_{k=1}^Q$ and assume $s \leq Q$. The Gram matrix W of the set of vectors $\{g_i\}_{i=1}^s$ is equal to

$$W = G^T M G,$$

where $G \in \mathbb{R}^{N\times s}$ is the matrix whose columns are the vectors $\{g_i\}_{i=1}^s$, and where $M \in \mathbb{R}^{N\times N}$ is the symmetric positive definite matrix associated with the chosen residual norm $\|\cdot\|$ in $\mathbb{R}^N$. Therefore, it remains to compute the matrix G. Let $\Gamma \in \mathbb{R}^{s\times Q}$ and $\mathfrak{R} \in \mathbb{R}^{N\times Q}$ be the matrices whose columns are the evaluations of $\gamma(\xi)$ and $R(\xi)$ respectively, i.e.

$$\Gamma = [\gamma(\xi_1), \ldots, \gamma(\xi_Q)] \quad \text{and} \quad \mathfrak{R} = [R(\xi_1), \ldots, R(\xi_Q)],$$

such that

$$\mathfrak{R} = G\Gamma$$

holds. If the rank of Γ is not s, then we can find a factorization $\Gamma = L\widetilde{\Gamma}$ where the matrix $\widetilde{\Gamma} \in \mathbb{R}^{\tilde{s}\times Q}$ has full rank $\tilde{s} < s$ (e.g. using SVD or QR factorization) and write $\mathfrak{R} = \widetilde{G}\widetilde{\Gamma}$, with $\widetilde{G} = GL$. Therefore, without loss of generality, we now assume that Γ has a rank s.

Let $\xi'_1, \ldots, \xi'_s$ be the samples associated with s linearly independent columns of Γ and let $\Gamma' \in \mathbb{R}^{s\times s}$ (resp. $\mathfrak{R}' \in \mathbb{R}^{N\times s}$) be the submatrix of Γ (resp. $\mathfrak{R}$) associated with these samples,

$$\Gamma' = [\gamma(\xi'_1), \ldots, \gamma(\xi'_s)] \quad \text{and} \quad \mathfrak{R}' = [R(\xi'_1), \ldots, R(\xi'_s)].$$

Γ' is thus invertible, and

$$G = \mathfrak{R}' \left(\Gamma'\right)^{-1},$$

so that s evaluations of $R(\xi)$ are sufficient to compute G, and then the Gram matrix $W = G^T M G$.

2.2 Empirical interpolation method

Here, we present the EIM for the construction of an interpolation $\mathcal{I}_r[\gamma](\xi)$ of the vector-valued function $\gamma(\xi)$, with a control of the approximation error in the semi-norm $\|\cdot\|_W$. The interpolation $\mathcal{I}_r[\gamma](\xi)$ has the form (5), where the functions $\alpha_j(\xi)$ are defined such that for any $\xi \in \Xi$, $(\mathcal{I}_r[\gamma])_i(\xi) = \gamma_i(\xi)$ for a collection of indices $\{i_j\}_{j=1}^r$, i.e.

$$\sum_{j=1}^r \gamma_i(\xi_j^\star)\alpha_j(\xi) = \gamma_i(\xi), \quad \forall i \in \{i_1, \ldots, i_r\}. \tag{8}$$

For the selection of the interpolation points and indices, we use a greedy algorithm [3] which generates a sequence of pairs $\{(\xi_r^\star, i_r)\}_{r\geq 1}$ defined recursively by

$$\xi_{r+1}^\star \in \arg\max_{\xi\in\Xi} \|\gamma(\xi) - \mathcal{I}_r[\gamma](\xi)\|_W,$$

$$\text{and} \quad i_{r+1} \in \arg\max_{i\in\{1,\ldots,s\}} |\gamma_i(\xi_{r+1}^\star) - (\mathcal{I}_r[\gamma])_i(\xi_{r+1}^\star)|,$$

where for $r = 0$, we use the convention $\mathcal{I}_0[\gamma] = 0$. This construction ensures that the linear system of equations (8) is invertible for any $\xi \in \Xi$, and in particular, it ensures that the interpolation property (6) is satisfied. If the algorithm is stopped when r is such that

$$\max_{\xi \in \Xi} \|\gamma(\xi) - \mathcal{I}_r[\gamma](\xi)\|_W \leq \zeta, \tag{9}$$

it yields an interpolation of the residual such that $\|R(\xi) - \mathcal{I}_r[R](\xi)\| \leq \zeta$.

Remark 2.1. *We emphasize that the standard EIM applied to $R(\xi)$ should have required the evaluation of the residual $R(\xi)$ for all $\xi \in \Xi$ (Q evaluations), while the proposed approach requires the values of $\gamma(\xi)$ for all $\xi \in \Xi$ and only s evaluations of the residual, where s is the rank of γ.*

Remark 2.2. *The strategy presented in Sections 2.1 and 2.2 can be directly applied for the interpolation of the preconditioner, with an error control with respect to a matrix norm associated with an inner product, such as the Frobenius norm. Note that, controlling the error with respect to such a norm does not allow a sharp control of the error in subordinate matrix norms.*

3 Approximation of residuals and preconditioners with unknown low-rank structures

In this section, we consider the approximation of the residual and the preconditioner without a priori knowledge on their representations as parameter-dependent algebraic quantities. We assume that if $u(\xi) = \sum_{i=1}^{m} v_i \lambda_i(\xi)$, then the residual $R(\xi) := R(u(\xi); \xi)$ and the preconditioner $P(\xi) := P(u(\xi); \xi)$ are well-approximated in low-rank format, i.e.

$$R(\xi) \approx \sum_{i=1}^{r} g_i \gamma_i(\xi) \quad \text{and} \quad P(\xi) \approx \sum_{i=1}^{p} F_i \phi_i(\xi),$$

with moderate ranks r and p. However, we have no (even partial) information on these low-rank representations.

First, we present the strategy for interpolating the residual. Then a statistical error bound is derived for the a posteriori control of the approximation error. Finally, the method is extended to the interpolation of the preconditioner.

3.1 Interpolation of the residual

We here use a randomized version of the EIM, which is called adaptive cross approximation with partial pivoting in other contexts [4], for the construction of a sequence of

interpolations of $R(\xi)$ of the form

$$\mathcal{I}_r[R](\xi) = \sum_{j=1}^{r} R(\xi_j^\star)\alpha_j(\xi), \tag{10}$$

where the $\xi_j^\star$ are interpolation points in Ξ and the $\alpha_j(\xi)$ are real-valued functions satisfying the interpolation property $\alpha_j(\xi_j^\star) = \delta_{i,j}$, $1 \leq i, j \leq r$. These functions are defined such that for any $\xi \in \Xi$, $(\mathcal{I}_r[R])_i(\xi) = R_i(\xi)$ for a collection of indices $\{i_j\}_{j=1}^r$, i.e.

$$\sum_{j=1}^{r} R_i(\xi_j^\star)\alpha_j(\xi) = R_i(\xi), \quad \forall i \in \{i_1, \ldots, i_r\}. \tag{11}$$

The strategy differs from the one of Section 2.2 for the selection of interpolation points. Here, given $\{((i_j, \xi_j^\star)\}_{j=1}^r$ and the corresponding interpolation $\mathcal{I}_r[R]$, we select randomly at uniform the point $\xi_{r+1}^\star$ in $\Xi \backslash \{\xi_k^\star\}_{k=1}^r$. If $R(\xi_{r+1}^\star) - \mathcal{I}_r[R](\xi_{r+1}^\star) = 0$, then the point is rejected and a new candidate point $\xi_{r+1}^\star$ is randomly generated. If $R(\xi_{r+1}^\star) - \mathcal{I}_r[R](\xi_{r+1}^\star) \neq 0$, an associated index i_{r+1} is selected such that

$$i_{r+1} \in \arg\max_{i \in \{1,\ldots,N\}} |R_i(\xi_{r+1}^\star) - \mathcal{I}_r[R]_i(\xi_{r+1}^\star)|.$$

The selection of interpolation points does not satisfy an optimality condition but in contrast to standard EIM, it does not require the evaluation of $R(\xi)$ for all ξ. The condition $R(\xi_r^\star) - \mathcal{I}_{r-1}[R](\xi_r^\star) \neq 0$ ensures that the system of equations (11) admits a unique solution (see [4]).

3.2 Statistical error control

In order to certify the approximation, we provide a statistical bound for the error of interpolation of $R(\xi)$, based on evaluations of some entries of $R(\xi)$. Let $(I_k)_{k\in\mathbb{N}}$ (resp. $(\xi_k)_{k\in\mathbb{N}}$) be independent random variables with values in $\{1, \ldots, N\}$ (resp. Ξ) following the uniform law. Then the random variables $(X_k)_{k\geq 1}$ defined by

$$X_k = NQ(R_{I_k}(\xi_k) - (\mathcal{I}_r)[R]_{I_k}(\xi_k))^2$$

are independent are identically distributed. By the law of large numbers, the random variable $Y_M = \frac{1}{M}\sum_{k=1}^M X_k$ converges almost surely to $\mathbb{E}(X_k) = \sum_{\xi\in\Xi} \|R(\xi) - \mathcal{I}_r[R](\xi)\|_2^2 = \|R - \mathcal{I}_r[R]\|_F^2$ as $M \to \infty$, i.e. Y_M is a convergent and unbiased statistical estimation of the square of the interpolation error with respect to the Frobenius norm.

Let σ_M^2 be the statistical estimation of the variance of X_k, defined by

$$\sigma_M^2 = \frac{1}{M-1}\sum_{k=1}^{M}(X_k - Y_M)^2.$$

The random variable

$$\frac{Y_M - \|R - \mathcal{I}_r[R]\|_F^2}{\frac{\sigma_M}{\sqrt{M}}}$$

converges in law to a random variable T_M having the Student's t-distribution with $M-1$ degrees of freedom, as $M \to \infty$. Letting $t_{\alpha,M} \geq 0$ be such that $\mathbb{P}(T_M \leq t_{\alpha,M}) = \mathbb{P}(T_M \geq -t_{\alpha,M}) = 1-\alpha$, and

$$e_{M,\alpha}^2 = Y_M + t_{\alpha,M}\frac{\sigma_M}{\sqrt{M}}, \tag{12}$$

we then have

$$\mathbb{P}\left(\|R - \mathcal{I}_r[R]\|_F \leq e_{M,\alpha}\right) \underset{M\to\infty}{\longrightarrow} \mathbb{P}(T_M \geq -t_{\alpha,M}) = 1-\alpha,$$

which means that $e_{M,\alpha}$ is an asymptotic upper bound with confidence level $1-\alpha$ for the interpolation error.

3.3 Interpolation of the preconditioner

We define an interpolation $\mathcal{I}_r[P](\xi)$ of the operator $P(\xi)$ of the form

$$\mathcal{I}_r[P](\xi) = \sum_{k=1}^{r} P(\xi_k^\sharp)\beta_k(\xi), \tag{13}$$

where the $\xi_k^\sharp$ are interpolation points in Ξ and the $\beta_k(\xi)$ are real-valued functions satisfying the interpolation property $\beta_k(\xi_l^\sharp) = \delta_{k,l}$, $1 \leq k, l \leq r$. These functions are defined such that for all $\xi \in \Xi$, $(\mathcal{I}_r[P])_\alpha(\xi) = P_\alpha(\xi)$ for a subset of pairs of indices $\mathcal{A}_r = \{\alpha_k = (i_k, j_k)\}_{k=1}^r \subset \mathcal{A} := \{1, \ldots, N\}^2$, i.e.

$$\sum_{k=1}^{r} P_\alpha(\xi_k^\sharp)\beta_k(\xi) = P_\alpha(\xi), \quad \forall \alpha \in \mathcal{A}_r. \tag{14}$$

For the selection of the interpolation points and corresponding entries of matrices, we use again a greedy strategy. Given $\{\xi_k^\sharp\}_{k=1}^r$ and $\{\alpha_k\}_{k=1}^r$, we select $\xi_{r+1}^\sharp$ at random in $\Xi \setminus \{\xi_k^\sharp\}_{k=1}^r$ (until $P(\xi_k^\sharp) - \mathcal{I}_r[P](\xi_k^\sharp) \neq 0$) and we determine a corresponding pair of indices $\alpha_{r+1} = (i_{r+1}, j_{r+1})$ such that

$$\alpha_{r+1} \in \arg\max_{\alpha\in\mathcal{A}} |P_\alpha(\xi_{r+1}^\sharp) - \mathcal{I}_r[P]_\alpha(\xi_{r+1}^\sharp)|.$$

A statistical control of the error in the Frobenius norm can be obtained as in Section 3.2, with random variables X_k replaced by

$$X_k = N^2 Q(P_{A_k}(\xi_k) - \mathcal{I}_r[P]_{A_k}(\xi_k))^2, \tag{15}$$

where $(A_k)_{k\in\mathbb{N}}$ are independent random variables with values in $\mathcal{A}$ and with uniform law. For sparse parameter-dependent matrices $P(\xi)$ such that $P_\alpha(\xi) = 0$ for all $\alpha \in \mathcal{A}$ and all $\xi \in \Xi$, the random variables A_k are taken uniform on $\mathcal{A} \setminus \mathcal{A}_0$ and N^2 in Equation (15) is replaced by $\#\mathcal{A} \setminus \mathcal{A}_0$.

4 Computation of the iterates

Sections 2 and 3 provide two alternatives for computing low-rank approximations $\widetilde{R}(u^k(\xi);\xi) := R(\xi)$ and $\widetilde{P}(u^k(\xi);\xi) := P(\xi)$ of the residual $R(u^k(\xi);\xi)$ and preconditioner $P(u^k(\xi);\xi)$ at iteration k of the Newton-type algorithm,

$$R(\xi) = \sum_{i=1}^{r_R} R(\xi_i^\star)\alpha_i(\xi) \quad \text{and} \quad P(\xi) = \sum_{i=1}^{r_P} P(\xi_i^\sharp)\beta_i(\xi).$$

Here, we present an algorithm which exploits these low-rank representations for efficiently computing an approximation of the increment $\Delta u(\xi)$, solution of the following equation

$$P(\xi)\Delta u(\xi) = R(\xi). \tag{16}$$

The proposed algorithm is a greedy rank-one algorithm [5, 9] which provides a sequence of approximations $(\Delta u_r(\xi))_{r\geq 1}$ with increasing ranks, defined by

$$\Delta u_r(\xi) = \Delta u_{r-1}(\xi) + w_r\theta_r(\xi),$$

where $\Delta u_0 = 0$, and where the rank-one correction $w_r\theta_r(\xi)$ is the solution of the optimization problem

$$\min_{w\in\mathbb{R}^N,\theta\in\mathbb{R}^\Xi} \sum_{\xi\in\Xi} \|P(\xi)w\theta(\xi) - R_r(\xi)\|_M^2, \tag{17}$$

where

$$\begin{aligned} R_r(\xi) &= R(\xi) - P(\xi)\Delta u_{r-1} \\ &= \sum_{i=1}^{r_R} R(\xi_i^\star)\alpha_i(\xi) - \sum_{i=1}^{r_P}\sum_{j=1}^{r-1} P(\xi_i^\sharp)w_j\beta_i(\xi)\theta_j(\xi) := \sum_{i=1}^{s} g_i\gamma_i(\xi), \end{aligned}$$

and where the matrix M (possibly parameter-dependent) defines a residual norm. For the solution of (17), we use an alternating minimization algorithm which consists in successively

- minimizing over $w \in \mathbb{R}^N$, which yields the linear system of equations $Aw = b$, where

$$A = \sum_{\xi\in\Xi} P(\xi)^T M P(\xi)\theta(\xi)^2 = \sum_{i=1}^{r_P}\sum_{j=1}^{r_P} P(\xi_i^\sharp)^T M P(\xi_j^\sharp)\Big(\sum_{\xi\in\Xi}\beta_i(\xi)\beta_j(\xi)\theta(\xi)^2\Big),$$

  and

$$b = \sum_{\xi\in\Xi} P(\xi)^T M R_r(\xi)\theta(\xi) = \sum_{i=1}^{r_P}\sum_{j=1}^{s} P(\xi_i^\sharp)^T M g_j\Big(\sum_{\xi\in\Xi}\beta_i(\xi)\gamma_j(\xi)\theta(\xi)\Big),$$

- minimizing over $\theta \in \mathbb{R}^{\Xi}$, which yields

$$\theta(\xi) = \frac{w^T P(\xi)^T M R_r(\xi)}{w^T P(\xi)^T M P(\xi) w}, \quad \xi \in \Xi,$$

and iterating until convergence.

Remark 4.1. *In the case where $P(\xi)$ is symmetric positive definite for all $\xi \in \Xi$, then we can choose for M the parameter-dependent matrix $M = P(\xi)^{-1}$. The optimization problem* (17) *defining the rank-one correction $w_r\theta_r(\xi)$ is then equivalent to*

$$\min_{w \in \mathbb{R}^N, \theta \in \mathbb{R}^{\Xi}} \sum_{\xi \in \Xi} (w\theta(\xi))^T P(\xi) w\theta(\xi) - 2 \sum_{\xi \in \Xi} (w\theta(\xi))^T R_r(\xi). \tag{18}$$

In the alternating minimization algorithm, the minimization over w yields a system of equations $Aw = b$ with

$$A = \sum_{\xi \in \Xi} P(\xi)\theta(\xi)^2 = \sum_{i=1}^{r_P} P(\xi_i^\sharp) \sum_{\xi \in \Xi} \beta_i(\xi)\theta(\xi)^2,$$

and

$$b = \sum_{\xi \in \Xi} R_r(\xi)\theta(\xi) = \sum_{j=1}^{s} g_i \sum_{\xi \in \Xi} \gamma_j(\xi)\theta(\xi),$$

and the minimization over θ yields

$$\theta(\xi) = \frac{w^T R_r(\xi)}{w^T P(\xi) w}, \quad \xi \in \Xi.$$

5 Truncated iterative solver

The proposed algorithm constructs a sequence of approximations $(u^k)_{k \geq 0}$ as follows, starting with $u^0 = 0$. At iteration k, we compute low-rank approximations $\widetilde{R}(u^k(\xi); \xi)$ and $\widetilde{P}(u^k(\xi); \xi)$ of $R(u^k(\xi); \xi)$ and $P(u^k(\xi); \xi)$ with one of the approaches presented in Sections 2 and 3. Then, we compute a low-rank approximation $\Delta u^k(\xi)$ of $\widetilde{P}(u^k(\xi); \xi)^{-1} \widetilde{R}(u^k(\xi); \xi))$ with the greedy low-rank algorithm described in Section 4. Finally, we define the next iterate by

$$u^{k+1} = \Pi_\varepsilon(u^k + \Delta u^k), \tag{19}$$

where Π_ε is a truncation operator such that $\Pi_\varepsilon(v)$ provides a low-rank approximation of a function $v(\xi)$ with a controlled precision ε in L^2 norm, i.e.

$$\sum_{\xi \in \Xi} \|\Pi_\varepsilon(v)(\xi) - v(\xi)\|^2 \leq \varepsilon^2 \sum_{\xi \in \Xi} \|v(\xi)\|^2,$$

with a practical implementation relying on SVD. The truncation operator allows to avoid a blow-up in the representation ranks of the iterates.

Now, we analyze the proposed algorithm in the particular case of a Newton solver, where $P(u(\xi);\xi) = -R'(u(\xi);\xi)$, with $R'(u(\xi);\xi)$ the differential of $R(\cdot;\xi)$ at $u(\xi)$, and analyze the proposed algorithm as an inexact Newton method, following Dembo et al. [8]. This will provide us guidelines to avoid unnecessary efforts in the approximation of the different quantities (residual, preconditioner, increments and iterates). We first rewrite the truncated Newton algorithm in the space $(\mathbb{R}^N)^\Xi$ equipped with the norm $\|\cdot\|$ defined by $\|v\|^2 = \sum_{\xi\in\Xi} \|v(\xi)\|^2$, where $\|v(\xi)\|$ is the Euclidean norm of $v(\xi)$. The parameter-dependent nonlinear system of equations is written

$$\mathcal{R}(u) := (R(u(\xi);\xi))_{\xi\in\Xi} = 0,$$

where $\mathcal{R} : (\mathbb{R}^N)^\Xi \to (\mathbb{R}^N)^\Xi$. We denote by $\mathcal{R}'(u)$ the differential of the residual $\mathcal{R}(\cdot)$ at u, such that $\mathcal{R}'(u)v = (R'(u(\xi);\xi)v(\xi))_{\xi\in\Xi}$ for $v \in (\mathbb{R}^N)^\Xi$. $\mathcal{R}'(u)$ is an element of the space of linear operators from $(\mathbb{R}^N)^\Xi$ to $(\mathbb{R}^N)^\Xi$, which we equip with the operator norm $\|M\| = \max_{v\in(\mathbb{R}^N)^\Xi} \|Mv\|/\|v\|$.

Then the algorithm can be rewritten

$$\begin{aligned}\widetilde{\mathcal{R}}'(u^k)\Delta u^k &= -\widetilde{\mathcal{R}}(u^k) + \tilde{r}^k,\\ u^{k+1} &= u^k + \Delta u^k + e^k,\end{aligned}$$

where $\widetilde{\mathcal{R}}(u^k)$ and $\widetilde{\mathcal{R}}'(u^k)$ are approximations of $\mathcal{R}(u^k)$ and $\mathcal{R}'(u^k)$ respectively, Δu^k is the approximation of $\widetilde{\mathcal{R}}'(u^k)^{-1}\widetilde{\mathcal{R}}(u^k)$ computed with the greedy rank-one algorithm, $\tilde{r}^k$ the associated residual, and e^k represents the error related to the truncation step.

In the following, we assume that for all $\xi \in \Xi$,

(A1) there exists a unique solution $u(\xi)$ to $R(u(\xi);\xi) = 0$,

(A2) $R(\cdot;\xi)$ is continuously differentiable,

(A3) $R'(u(\xi);\xi)$ is invertible.

These assumptions respectively imply that there exists a unique solution to $\mathcal{R}(u) = 0$, $\mathcal{R}$ is continuously differentiable, and $\mathcal{R}'(u)$ is invertible.

Theorem 5.1. *Assume that*

- u^k *converges to the solution* u,
- $R(\cdot;\xi)$ *is Lipschitz continuous uniformly in* ξ, *i.e. there exists a constant* $C > 0$ *independent of* ξ *such that for all* $\xi \in \Xi$,

$$\|R(v;\xi) - R(w;\xi)\| \le C\,\|v - w\|\,, \quad \forall v, w \in \mathbb{R}^N,$$

- *For k sufficiently large, $\mathcal{R}'(u^k)$ is such that*

$$\alpha\|v\| \leq \left\|\mathcal{R}'(u^k)v\right\| \leq \beta\|v\|, \quad \forall v \in \mathbb{R}^N,$$

  for some constants α, β independent of k,

- *$\widetilde{\mathcal{R}}(u^k)$ (resp. $\widetilde{\mathcal{R}}'(u^k)$) is an approximation of $\mathcal{R}(u^k)$ (resp. $\mathcal{R}'(u^k)$) such that*

$$\|\widetilde{\mathcal{R}}(u^k) - \mathcal{R}(u^k)\| \leq \rho_k \quad \textit{and} \quad \|\widetilde{\mathcal{R}}'(u^k) - \mathcal{R}'(u^k)\| \leq \rho'_k.$$

If ρ_k, $\|e^k\|$ and $\|\widetilde{r}_k\|$ are $o(\|\mathcal{R}(u^k)\|)$ and ρ'_k is $o(1)$, then u^k converges to u superlinearly. Furthermore, if ρ_k, $\|e^k\|$ and $\|\widetilde{r}_k\|$ are $O(\|\mathcal{R}(u^k)\|^2)$ and ρ'_k is $O(\|\mathcal{R}(u^k)\|)$, then the sequence u^k converges with order at least 2.

Proof. Letting

$$s^k = u^{k+1} - u^k = \Delta u^k + e^k,$$

the algorithm can be rewritten as an inexact Newton solver

$$\mathcal{R}'(u^k)s^k = -\mathcal{R}(u^k) + r^k, \quad u^{k+1} = u^k + s^k,$$

where the residual $r^k = \mathcal{R}'(u^k)s^k + \mathcal{R}(u^k)$ has the following decomposition

$$r^k = \mathcal{R}'(u^k)e^k + (\mathcal{R}'(u^k) - \widetilde{\mathcal{R}}'(u^k))\Delta u^k + \mathcal{R}(u^k) - \widetilde{\mathcal{R}}(u^k) + \widetilde{r}^k.$$

Then

$$\|r^k\| \leq \|\mathcal{R}'(u^k)\|\|e^k\| + \rho'_k\|\Delta u^k\| + \rho_k + \|\widetilde{r}^k\|,$$

with

$$\|\Delta u^k\| \leq \|\mathcal{R}'(u^k)^{-1}\|(\|\widetilde{\mathcal{R}}(u^k)\| + \|\widetilde{r}^k\|) \leq \|\mathcal{R}'(u^k)^{-1}\|(\rho_k + \|\mathcal{R}(u^k)\| + \|\widetilde{r}^k\|).$$

Then, for k sufficiently large,

$$\|r^k\| \leq \beta\|e^k\| + \alpha^{-1}\rho'_k(\rho_k + \|\mathcal{R}(u^k)\| + \|\widetilde{r}^k\|) + \rho_k + (1 + \alpha^{-1})\|\widetilde{r}^k\|. \tag{20}$$

If ρ_k, $\|e^k\|$ and $\|\widetilde{r}_k\|$ are $o(\|\mathcal{R}(u^k)\|)$ and ρ'_k is $o(1)$, then $\|r^k\|$ is $o(\|\mathcal{R}(u^k)\|)$. If ρ_k, $\|e^k\|$ and $\|\widetilde{r}_k\|$ are $O(\|\mathcal{R}(u^k)\|^2)$ and ρ'_k is $O(\|\mathcal{R}(u^k)\|)$, then $\|r^k\|$ is $O(\|\mathcal{R}(u^k)\|^2)$. We then conclude by using [8, Th. 3.3]. □

Even though we provide guidelines to control the convergence rate of the Newton algorithm, the computation of α, β and ρ'_k is not a simple task. It requires the ability to compute the largest and lowest singular values of a linear operator on $(\mathbb{R}^N)^\Xi$, and to ensure that the singular values of $(\mathcal{R}'(u^k))_{k\in\mathbb{N}}$ are bounded by α and β.

6 Numerical example

6.1 Diffusion with nonlinear reaction equation

6.1.1 Problem setting

Let $\Omega = (0,1)^2$. We want to solve for all $\xi \in \Xi$ the nonlinear PDE

$$\begin{aligned} -\Delta u + \frac{\xi}{3} u^3 &= 1 \quad \text{on } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{21}$$

for $\Xi = (\xi_q)_{q=1}^{Q}$, a set of $Q = 5000$ i.i.d. samples is drawn such that $\xi = \exp(\zeta) - 1$, where the distribution of ζ is uniform between 0 and 10. The PDE is discretized with a finite element method where the dimension of the approximation space is $N = 9801$.

Assume that an approximation of the solution $u(\xi) = \sum_{i=1}^{m} v_i \lambda_i(\xi)$ is available. The strong form of the residual is

$$\begin{aligned} R^{\text{strong}}(u(\xi);\xi) &= 1 + \Delta u - \frac{\xi}{3} v u^3 \\ &= 1 + \sum_{i=1}^{m} \Delta v_i \lambda_i(\xi) - \frac{\xi}{3} \sum_{j=1}^{m}\sum_{k=1}^{m}\sum_{l=1}^{m} v_j v_k v_l \lambda_j(\xi)\lambda_k(\xi)\lambda_l(\xi) \\ &= \sum_{i=1}^{1+m+m^3} \gamma_i(\xi) G_i. \end{aligned}$$

We can clearly see here that the evaluation of γ only requires the knowledge of the collection $(\lambda_i)_{i=1}^{m}$ and the structure of the equation. It is therefore computable without having to sample the collection $(G_i)_{i=1}^{1+m+m^3}$. The considered preconditioner for this problem is the Jacobian of the residual hence, a Newton solver is used for solving this discretized equation. Given the low-rank structure of the solution, the preconditioner admits then an expansion of the form

$$P(\xi) = \sum_{i=1}^{1+m^2} P_i \phi_i(\xi),$$

where $\phi_1(\xi) = 1$ comes from the diffusion term, while $(\phi_i(\xi))_{i=2}^{1+m^2}$ are due to the cubic reaction term and are of the form $\phi_i(\xi) = \xi \lambda_j(\xi)\lambda_k(\xi)$.

6.1.2 Computation of the solution by exploiting the known low-rank structure

Given λ, the maps γ and ϕ are explicitly known. As a consequence, the example presented in this section fits the framework presented in Section 2 and we are therefore able to solve

this nonlinear problem in a weakly-intrusive manner, based on evaluations of the residual and the preconditioner.

We use here the guidelines provided by Theorem 5.1. The residual is approximated by the EIM such that

$$\left\|\mathcal{R}(u) - \widetilde{\mathcal{R}}(u)\right\| \leq \rho_{\mathcal{R}} \left\|\mathcal{R}(u)\right\|^2 . \tag{22}$$

Note that the error control of the EIM is done according to the supremum norm $\|\cdot\|_\infty$ defined by $\|v\|_\infty = \sup_{\xi\in\Xi}\|v(\xi)\|$ as stated in Equation (9). We therefore use the following inequality to bound the norm of the residual

$$\left\|\mathcal{R}(u)\right\|^2 \leq Q \left\|\mathcal{R}(u)\right\|_\infty^2 .$$

We therefore set the tolerance of the EIM to

$$\left\|\mathcal{R}(u)\right\|_\infty \leq \frac{\rho_R}{\sqrt{Q}} \left\|\mathcal{R}(u)\right\|^2 ,$$

such that Equation (22) is satisfied.

Regarding the interpolation of the preconditioner, we arbitrarily set

$$\left\|\mathcal{P}(u) - \widetilde{\mathcal{P}}(u)\right\|_F = \left(\sum_{\xi\in\Xi} \left\|P(u(\xi);\xi) - \widetilde{P}(u(\xi);\xi)\right\|_F\right)^{1/2} \leq \rho_P \left\|\mathcal{R}(u)\right\| ,$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Note that this condition implies that the spectral norm of the error on the interpolation of the Jacobian is $\mathcal{O}(\|\mathcal{R}(u)\|)$ as required in Theorem 5.1. In practice, ρ_R and ρ_P are set to 10^{-2}.

Concerning the tolerance parameters of the low-rank linear solver and the SVD truncation, they are arbitrary set to 10^{-12} given that these methods are computationally cheap compared to the approximation of the residual and the preconditioner. The low-rank solver is both controlled with respect to the norm of the relative residual and the stagnation of the approximation.

The error estimate ϵ is given by

$$\epsilon(u)^2 = \frac{\sum_{\xi\in\Xi} \left\|\widetilde{R}(u(\xi);\xi)\right\|^2}{\sum_{\xi\in\Xi} \left\|R(0;\xi)\right\|^2}, \tag{23}$$

and the computational performance of the algorithm is assessed using the cumulative number of calls to R and P.

In Table 1, the values of the error indicator $\epsilon(u)$, as well as the cumulative numbers of calls to R and P are given with respect to the number of iterations of the global Newton solver. The normalized cost is defined as the ratio between the effective number of calls to R or P and the number of calls required by a full-blown Monte-Carlo method. First,

the quadratic convergence of the Newton's method holds in this numerical experiment. As we can see, the estimated relative residual goes from 10^{-5} to 10^{-10} between iterations 4 and 5, in agreement with the convergence rate predicted by Theorem 5.1. Moreover, the table illustrates substantial computational gains. In particular, the proposed strategy requires the assembly of 448 residuals to solve the problem which corresponds to 1.18% of the assembly of the 25000 residuals requires for performing 5 iterations for each sample in a Monte-Carlo approach. The gain is even more important for preconditioners, as the technique requires the computation of only 4.40‰ of the number of preconditioner evaluations required by a Monte-Carlo method.

Table 1: Error indicator, cumulative number of calls to R and P and normalized cost of the assemblies compared to a Monte-Carlo method w.r.t. the number of iterations of the Newton's solver for the solution of Problem (21).

Iter.	$\epsilon(u)$	Residual		Preconditioner	
		#Calls	Cost	#Calls	Cost
1	2.40×10^{-1}	3	6.00×10^{-4}	1	2.00×10^{-4}
2	3.94×10^{-2}	41	4.10×10^{-3}	3	3.00×10^{-4}
3	2.27×10^{-3}	210	1.14×10^{-2}	18	1.20×10^{-3}
4	1.19×10^{-5}	375	1.88×10^{-2}	65	3.25×10^{-3}
5	4.07×10^{-10}	448	1.18×10^{-2}	110	4.40×10^{-3}

6.1.3 Approximation without prior knowledge on the structure of the equation

We consider the problem introduced in Section 6.1.1 where we ignore the prior knowledge on the low-rank structure of the residual and the preconditioner. Therefore, the strategy introduced in Section 3 is considered.

Regarding the tolerance values, the error is set to 10^{-12} for the low-rank linear solver and the SVD truncation. Regarding the randomized EIM, the error is assessed with $M = Q = 5000$ entries and the confidence level is set to $\alpha = 95\%$ for the approximation of the residual and the preconditioner. Let Z_M be defined by

$$Z_M^2 = \frac{NQ}{M} \sum_{k=1}^{M} R_{I_k}(\xi_k)^2,$$

where $(I_k)_k$ and $(\xi_k)_k$ are random variables defined in Section 3.2. Then Z_M is an estimator of $\|\mathcal{R}(u)\|$. Therefore, the convergence criterion on the residual is set such that the algorithm stops when one of the following conditions is satisfied:

$$e_{M,\alpha} \leq \rho_{\mathcal{R}} Z_M^2, \quad \text{or} \quad \max_{1 \leq k \leq M} \left| R_{I_k}(\xi_k) - \widetilde{R}_{I_k}(\xi_k) \right| \leq 10^{-15}, \tag{24}$$

where $(I_k, \xi_k)_k$ are the random entries sampled for the error estimation. The condition on the supremum norm of the error on the test set avoids excessive tolerances when realizations of Z_M^2 is small.

The approximation of the Jacobian is controlled such that

$$e_{M,\alpha}^P \leq \rho_{\mathcal{R}} \left\| \widetilde{\mathcal{R}}(u) \right\|,$$

where $e_{M,\alpha}^P$ is the upper bound on the error estimated with M entries of the Jacobian and a confidence level α, as derived in Section 3.2 in the case of the residual.

Regarding performance measures, we consider both the error estimation $\epsilon(u)$ introduced in Equation (23) and a specific complexity measure. The complexities of the solution are defined as the ratio of the cumulative number of evaluated entries of R and P and the cumulative number of entries that should have been evaluated in the case of a Monte-Carlo method. Note that the measure takes into account the entries used to assess the error and the sparsity pattern of the preconditioner induced by the finite element as mentioned in Section 3.3.

Table 2 shows the error estimation and the complexities with respect to the iteration of the Newton's solver. First, since the sample Ξ and the initial guess $u^0 = 0$ are identical to Section 6.1.2, and since the tolerances are stringent, we observe that the quantity $\epsilon(u)$ has the same convergence than in Table 1 and differs only for very small errors. The quadratic convergence of the Newton method is also satisfied.

One notable difference with the method used in Section 6.1.2 is the computational cost of the approach. While the cost was 1.18% (resp. 4.40%) compared to the Monte-Carlo method in term of residual (resp. preconditioner) evaluations, here the normalized cost is only 6.22‰ for the residual and 1.48‰ for the preconditioner in terms of entry evaluations.

Table 2: Error estimation and complexity for the solution of Problem (21) w.r.t. the iterations of the Newton's solver without exploiting the structure of the residual and the preconditioner.

Iter.	$\epsilon(u)$	Residual complexity	Preconditioner complexity
1	2.40×10^{-1}	8.08×10^{-4}	2.11×10^{-4}
2	3.94×10^{-2}	1.81×10^{-3}	3.17×10^{-4}
3	2.27×10^{-3}	2.15×10^{-3}	7.75×10^{-4}
4	1.20×10^{-5}	5.64×10^{-3}	8.99×10^{-4}
5	3.94×10^{-10}	6.22×10^{-3}	1.48×10^{-3}

6.2 Nonlinear diffusion equation

We are interested now in a nonlinear diffusion equation defined on $\Omega = (0,1)^2$ for all $\xi \in \Xi$ by

$$\begin{aligned} -\nabla \cdot (\exp(\xi u(\xi))\nabla u(\xi)) &= 1 \quad \text{on } \Omega, \\ u(\xi) &= 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{25}$$

The sample $\Xi = (\xi_q)_{q=1}^Q$ is such that $Q = 5000$ and $\xi = \exp(\zeta) - 1$ where ζ is drawn according to the uniform distribution between 0 and 3. The weak form of the residual is given by

$$\langle v, R(u(\xi);\xi)\rangle = \int_\Omega v \mathrm{d}x - \int_\Omega \exp(\xi u(\xi))\nabla v \cdot \nabla u \, \mathrm{d}x,$$

and the Jacobian is

$$\left\langle v, R'(u(\xi),\xi)w\right\rangle = -\int_\Omega \exp(\xi u)\nabla v \cdot \nabla w \, \mathrm{d}x - \int_\Omega \xi \exp(\xi u)(\nabla u \cdot \nabla v)w \, \mathrm{d}x.$$

For the preconditioner, we will only consider the symmetric part of the $R'(u(\xi);\xi)$ (i.e. the first of the two terms) in order to improve the efficiency of the low-rank solver and avoid to treat non-symmetric problems. The global solver is therefore a modified Newton's method. Due to the exponential term, a low-rank expression of the residual or the preconditioner is not directly available, we are therefore in the framework presented in Section 3.

The mesh used for the finite element approximation is the same than in Section 6.1.1. Regarding the tolerances, error estimation and complexity estimation, we use the quantities defined in Section 6.1.3 with the difference that $\rho_R = 0.1$, $\rho_P = 0.1$ and that the tolerance on the approximation of the residual is set such that

$$e_{M,\alpha} \leq \rho_R Z_M,$$

the difference being that the upper bound is linear with Z_M and not quadratic anymore. As a consequence, the relative error on the approximation of the residual is of the order of ρ_R.

Table 3 shows the efficiency of the method in terms of relative residual estimate $\epsilon(u)$ and normalized complexities. The estimated error is 3.29×10^{-9} after 8 iterations. This time the quadratic convergence does not hold because first the preconditioner is not exactly the derivative of the residual and then the convergence of the error on the interpolation of the residual is not quadratic anymore. We are nevertheless able to get a high accuracy in terms of relative residual (reaching 3.28×10^{-10}) with a low computational cost compared to a Monte-Carlo method. Indeed, the final complexity regarding the computation of the residual and the preconditioner is similar to the computation of the solution of about 10 samples of the deterministic problem, i.e. the highest normalized cost between the residual and the preconditioner is 2.03‰.

Table 3: Error estimation and complexity for the solution of Problem (25) w.r.t. the iterations of the solver.

Iter.	$\epsilon(u)$	R norm. cost	P norm. cost
1	1.41×10^{-1}	8.08×10^{-4}	2.11×10^{-4}
2	1.75×10^{-2}	7.57×10^{-4}	6.34×10^{-4}
3	1.79×10^{-3}	7.40×10^{-4}	7.75×10^{-4}
4	1.27×10^{-4}	8.07×10^{-4}	8.99×10^{-4}
5	1.14×10^{-5}	7.87×10^{-4}	1.10×10^{-3}
6	2.16×10^{-6}	8.74×10^{-4}	1.27×10^{-3}
7	2.77×10^{-7}	9.80×10^{-4}	1.42×10^{-3}
8	2.58×10^{-8}	1.25×10^{-3}	1.48×10^{-3}
9	2.44×10^{-9}	1.19×10^{-3}	1.79×10^{-3}
10	3.28×10^{-10}	1.68×10^{-3}	2.03×10^{-3}

7 Conclusion

A framework for solving parameter-dependent nonlinear equations in a weakly intrusive manner is proposed. The method requires first the fast approximation of the residual and the preconditioner in order to be efficient. We show here that they can be interpolated in a weakly intrusive manner thanks to an extensive use of the empirical interpolation method, in its vector or matrix variants. These interpolations enables the use of an efficient greedy rank-one solver, which is used to compute the increments of the solution at each iteration. Finally, the current solution is compressed at each iteration in order to reduce its representation and the entire strategy is illustrated on numerical examples. A convergence analysis is performed in the particular case of the Newton's solver, and the theory is validated experimentally. The efficiency of the methods is illustrated on numerical examples.

This work is proof of concept and opens the way to more complex applications, in particular in nonlinear mechanics. Indeed, the assembly of the residual and the preconditioner for such problems represents the main part of the computational costs and the strategy proposed in this paper could be suitable. The algorithm would be then comparable to the one proposed in [10] where a sparse integration methodology is used to reduced the assembly cost. The robustness of the method when a large number of parameters is used should also be assessed in future work.

References

[1] M. Bachmayr and R. Schneider. Iterative methods based on soft thresholding of hierarchical tensors. *Foundations of Computational Mathematics*, pages 1–47, 2016.

[2] J. Ballani and L. Grasedyck. A projection method to solve linear systems in tensor format. *Numerical Linear Algebra with Applications*, 20(1):27–43, 2013.

[3] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ?empirical interpolation? method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[4] M. Bebendorf, Y. Maday, and B. Stamm. Comparison of some reduced representation approximations. In Alfio Quarteroni and Gianluigi Rozza, editors, *Reduced Order Methods for Modeling and Computational Reduction*, volume 9 of *MS&A - Modeling, Simulation and Applications*, pages 67–100. Springer International Publishing, 2014.

[5] E. Cancès, V. Ehrlacher, and T. Lelièvre. Convergence of a greedy algorithm for high-dimensional convex nonlinear problems. *Mathematical Models and Methods in Applied Sciences*, 21(12):2433–2467, 2011.

[6] F. Casenave, A. Ern, and T. Lelièvre. A nonintrusive reduced basis method applied to aeroacoustic simulations. *Advances in Computational Mathematics*, pages 1–26, 2014.

[7] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

[8] R. S. Dembo, S. C. Eisenstat, and T. Steihaug. Inexact newton methods. *SIAM Journal on Numerical Analysis*, 19(2):400–408, 1982.

[9] A. Falcó and A. Nouy. A proper generalized decomposition for the solution of elliptic problems in abstract form by using a functional eckart?young approach. *Journal of Mathematical Analysis and Applications*, 376(2):469–480, April 2011.

[10] C. Farhat, P. Avery, T. Chapman, and J. Cortial. Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency. *International Journal for Numerical Methods in Engineering*, 98(9):625–662, June 2014.

[11] L. Giraldi, A. Nouy, and G. Legrain. Low-rank approximate inverse for preconditioning tensor-structured linear systems. *SIAM Journal on Scientific Computing*, 36(4):A1850–A1870, 2014.

[12] L. Giraldi, A. Litvinenko, D. Liu, H. G. Matthies, and Anthony Nouy. To be or not to be intrusive? the solution of parametric and stochastic equations—the ?plain vanilla? galerkin case. *SIAM Journal on Scientific Computing*, 36(6):A2720–A2744, 2014.

[13] L. Giraldi, D. Liu, H. G. Matthies, and A. Nouy. To be or not to be intrusive? the solution of parametric and stochastic equations—proper generalized decomposition. *SIAM Journal on Scientific Computing*, 37(1):A347–A368, 2015.

[14] D. Kressner and C. Tobler. Low-rank tensor krylov subspace methods for parametrized linear systems. *SIAM Journal on Matrix Analysis and Applications*, 32(4):1288–1316, October 2011.

[15] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM Journal on Numerical Analysis*, 40(2):492–515, 2002.

[16] F. Negri, A. Manzoni, and D. Amsallem. Efficient model reduction of parametrized systems by matrix discrete empirical interpolation. *Journal of Computational Physics*, 303:431–454, 2015.