‘On-the-fly’ snapshots selection for Proper Orthogonal Decomposition with application to nonlinear dynamics

P. Phalippou, S. Bouabdallah, Piotr Breitkopf, P. Villon, M. Zarroug

▶ **To cite this version:**

P. Phalippou, S. Bouabdallah, Piotr Breitkopf, P. Villon, M. Zarroug. ‘On-the-fly’ snapshots selection for Proper Orthogonal Decomposition with application to nonlinear dynamics. Computer Methods in Applied Mechanics and Engineering, 2020, 367, pp.113120. 10.1016/j.cma.2020.113120 . hal-02952463

HAL Id: hal-02952463
https://hal.science/hal-02952463

Submitted on 3 Jun 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Version of Record: https://www.sciencedirect.com/science/article/pii/S0045782520303054
Manuscript_be0027c6fcfdda95eab45631bb9979cd

'On-the-fly' snapshots selection for Proper Orthogonal Decomposition with application to nonlinear dynamics

P. PHALIPPOU[1,2,*], S. BOUABDALLAH[1], P. BREITKOPF[2], P. VILLON[2], M. ZARROUG[3]
[1]ALTAIR ENGINEERING France, 5 rue de la Renaissance, 92160 Antony, France
[2]Laboratoire Roberval, FRE 2012 UTC-CNRS Université de Technologie de Compiègne France
[3]PSA Group, Route de Gisy, 78140 Vélizy-Villacoublay, France

Abstract

Over the last few decades, Reduced Order Modeling (ROM) has slowly but surely inched towards widespread acceptance in computational mechanics, as well as other simulation-based fields. Projection-based Reduced Order Modeling (PROM) relies on the construction of an appropriate Reduced Basis (RB), which is typically a low-rank representation of a set of "observations" made using full-field simulations, usually obtained through truncated Singular Value Decomposition (SVD). However, SVD encounters limitations when dealing with a large number of high-dimensional observations, requiring the development of alternatives such as the incremental SVD. The key advantages of this approach are reduced computational complexity and memory requirement compared to a regular "single pass" spectral decomposition. These are achieved by only using relevant observations to enrich the low-rank representation as and when available, to avoid having to store them. In addition, the RB may be truncated 'on-the-fly' so as to reduce the size of the matrices involved as much as possible and, by doing so, avoid the quadratic scale-up in computational effort with the number of observations. In this paper, we present a new error estimator for the incremental SVD, which is shown to be an upper bound for the approximation error, and propose an algorithm to perform the incremental SVD truncation and observation selection 'on-the-fly', instead of using a prohibitively large number of frequently "hard to set" parameters. The performance of the approach is discussed on the reduced-order Finite Element (FE) model simulation of impact on a Taylor beam.

Key Words : Singular Value Decomposition, Principal Component Analysis, low-rank representation, snapshot selection, Model Order Reduction, Proper Orthogonal Decomposition, vehicle crash simulation

Introduction

Model Order Reduction (MOR) is nowadays extensively used to tackle prohibitive computational times in numerical mechanics. Methods such as the Proper Orthogonal Decomposition (POD) and the Dynamic Mode Decomposition (DMD), exploit information contained in large data sets to understand and reduce the physics of a given problem [11, 26, 27], often invoking spectral decomposition methods.

Projection-based Reduced Order Modeling (PROM) methods approximate the unknown field variable as a linear combination of a set of domain-spanning Reduced Basis (RB) functions, that replace a large number of local, element-based shape functions. The POD, often used in computing the RB, traces its origins to statistical data analysis [22] and has found extensive application in turbulent flow modeling [6, 19, 20, 35] and various other fields [15, 24, 25, 30, 31, 32, 33]. The RB is built during the *offline* "training" phase through Singular Value Decomposition (SVD) of *previously collected* solution vector observations, typically called snapshots. PROM applications to highly nonlinear problems, such as complex structural dynamics involving nonlinear material laws, geometrical nonlinearity, contact, and crack formation, remains a challenge. The projection of internal variables may be approximated with the Discrete Empirical Interpolation Methods (DEIM) [9, 12, 36] or Hyper-Reduction (HR) [14, 18, 34].

Since the ultimate goal is a reduction in overall computation time, we need to limit the "cost" of the *offline* phase. SVD ends up either expensive or infeasible when applied to a large number

© 2020 published by Elsevier. This manuscript is made available under the CC BY NC user license
https://creativecommons.org/licenses/by-nc/4.0/

of high-dimensional snapshots. To tackle this issue, alternatives such as the "randomized SVD" [3, 4] and the "incremental SVD" [7] has recently emerged within the POD framework [28]. In this method, originally developed for streaming data analysis, SVD is performed by subsequent updates using a mathematical identity that we formally present in section 1. This avoids expensive data manipulation since snapshots are used for the updating as and when available. Moreover, incremental approaches enable *online* update of the Reduced Order Model (ROM), which is a weak point of methods relying on spectral decompositions against other popular methods such as the Proper Generalized Decomposition (PGD) [10, 16, 23, 37].

The highlights of the incremental SVD presented in [28] were: adaptive snapshot selection, *on-the-fly* snapshot selection and *on-the-fly* truncation. Adaptive snapshot selection identifies the simulation times at which observations must be made for the first order differential equation. *On-the-fly* snapshot selection pre-evaluates the (potential) contribution of a snapshot before actually updating the RB, while *on-the-fly* truncation limits the size of the RB during subsequent enrichments. These two features enable a reduction in computational effort by avoiding the treatment of redundant observations, thus allowing the algorithm to work on smaller matrices, which is discussed in detail in section 1.2.

Needless to say, the addition of these two features introduces new difficulties: firstly, two problem-dependent tolerances need to be set, and secondly, a loss of information in the SVD would inhibit the computation of the singular value truncation error necessary for the POD. An error bound for the incremental SVD has been developed in [13], where the authors proved that the incremental SVD yields the exact SVD of an approximated data set. Furthermore, their proposed error norm (calculated using a new weighted inner product) was the upper bound on the SVD error with respect to the original data set. This error bound is incrementally computed by keeping track of singular values missed due to 'on-the-fly' calculation of the RB.

In the current paper, the authors use the same incremented variables to build a new error estimator that will serve as an upper bound on the singular value truncation error. The present work was first presented by the author at the 6^{th} European Conference on Computational Mechanics (ECCM6) in Glasgow, UK [29]. The originality of our approach consists in using this incremented error estimator to monitor 'on-the-fly' truncation and snapshot selection, as opposed to the problem-dependent tolerances in the traditional approach. The result is a relatively straightforward incremental SVD with nearly optimal 'on-the-fly' truncation and snapshot selection. This version of incremental SVD is then implemented in the research branch of industry-level explicit dynamics simulation code Altaire Radioss [2] and its performance is tested against that obtained with standard SVD. Interestingly, the proposed approach, beyond limiting the number of parameters and displaying competitive performances in the *offline* training phase, performs well in the *online* phase. The latter is discussed in terms of the relative weight of repetitive snapshots in the produced RB.

The paper is organized in the following manner. In Section 1, we present a comprehensive review of the state-of-the-art incremental SVD. Section 2 presents the error estimator developed in this work. The algorithm is discussed in Section 3, and we then move on to validation of the approach in Section 4 where a Taylor beam impact FE model is used to test the proposed method against the traditional incremental SVD as well as single-pass SVD. To evaluate the efficacy of our approach, we compare computational time, RB performances during the *online* reduced-order run of the legacy explicit dynamics industrial code Altair Radioss [2] as well as the effective precision of the collected data approximation.

Notations

Throughout this paper, curly brackets designate vectors and square brackets designate matrices. Following notations are sorted out in alphabetical order.

$\{0_N\}$ - column vector of size N with all coefficients equal to zero

$\{1_N\}$ - column vector of size N with all coefficients equal to one

$\{\alpha\}$ - reduced unknown displacement, $\alpha \in \mathbb{R}^k$

b - plasticity hardening coefficient

$(c_i)_i$ - training configurations, $i \in [\![1, n_s]\!]$

E - Young's modulus

$\underline{\underline{\varepsilon}}$ - strain tensor

ε_p - equivalent plastic strain

ϵ_{est} - error estimator in the proposed incremental SVD algorithm

ϵ_{in} - part of discarded snapshots represented by $[\Phi^k]$

ϵ_{int} - work of internal forces approximation error

ϵ_{orth} - tolerance for reorthonormalization in the incremental SVD

ϵ_{out} - part of discarded snapshots non-represented by $[\Phi^k]$

ϵ_{rb} - user-specified threshold on the basis function selection criterion

ϵ_{sv}^k - singular value truncation error for a reduced basis of size k

ϵ_{svd} - tolerance for snapshot selection in the state-of-the-art incremental SVD

$\{f_{ext}\}$ - space discretized external forces, $\{f_{ext}\} \in \mathbb{R}^N$

$\{f_{int}\}$ - space discretized internal forces, $\{f_{int}\} \in \mathbb{R}^N$

$\|[\bullet]\|_F$ - Frobenius norm of a matrix, $\|[\bullet]\|_F = (\sum_{i,j} [\bullet]_{i,j}^2)^{\frac{1}{2}}$

k - Reduced basis size

k_{max} - Maximum size for the reduced basis in the state-of-the-art incremental SVD

$[\mathbb{M}]$ - symmetric positive-definite mass matrix, $[\mathbb{M}] \in \mathbb{R}^{N \times N}$

$[\tilde{\mathbb{M}}]$ - reduced mass matrix, $[\tilde{\mathbb{M}}] = [\Phi^{(B)}]^T [\mathbb{M}] [\Phi^{(B)}] \in \mathbb{R}^{k \times k}$

m - number of singular vectors in the training data POD decomposition, $m = \min(N, n_s)$

N - number of degrees of freedom in the full-order FE space discretization

n - plasticity hardening exponent

n_p - number of parameters of the FE model

n_s - number of snapshots in the training data set

ν - Poisson's ratio

$(p_i)_i$ - parameters of the FE model, $i \in [\![1, n_p]\!]$

$[\Phi]$ - left singular vectors of $[S]$, $[\Phi] = [\phi_1, \ldots, \phi_{n_s}] \in \mathbb{R}^{N \times m}$

$[\Phi^k]$ - reduced basis of size k

$[\Psi]$ - right singular vectors of $[S]$, $[\Psi] = [\psi_1, \ldots, \psi_m] \in \mathbb{R}^{n_s \times m}$

ρ - density

$[S]$ - training data set, $[S] = [\{u(t_1)\}, \ldots, \{u(t_{n_s})\}] \in \mathbb{R}^{N \times n_s}$

$\{s\}$ - singular values of $[S]$ in decreasing order, $\{s\} = (s_1, \ldots, s_m)^T \in \mathbb{R}^m$

$\underline{\underline{\sigma}}$ - stress tensor

σ_{eq} - equivalent stress

σ_Y - plastic yield stress

T - final simulation time

$\{u\}$ - space discretized unknown displacement field, $\{u\} \in \mathbb{R}^N$

$\{\tilde{u}\}$ - displacement field approximation in the reduced model, $\tilde{u} = [\Phi^k]\alpha \in \mathbb{R}^N$

V_0 - initial velocity

W_{int} - work of internal forces

1 Incremental SVD

The Singular Value Decomposition (SVD) is a generalization of the eigenvalue decomposition for non-square matrices. Given a matrix $[S] \in \mathbb{R}^{N \times n_s}$ the SVD is given by

$$[S] = [\Phi][\Sigma][\Psi]^T \tag{1}$$

In the 'regular' SVD $[\Phi]$ and $[\Psi]$ are respectively $N \times N$ and $n_s \times n_s$ square matrices and $[\Sigma]$ is a matrix of dimension $N \times n_s$ with non-negative values on the diagonal. In the 'thin' SVD, on the other hand, $[\Sigma] = [\text{diag}(\{s\})]$ is square with size $m = \min(N, n_s)$. From the ROM point of view, both 'regular' and 'thin' decomposition are equivalent, the reason being that the dimension of the space spanned by the RB may neither exceed the number of degrees of freedom in the Full Order Model (FOM) nor the number of snapshots. Hence a 'regular' SVD may result in unnecessary computations. In this paper, SVD refers to the "thin" SVD, $[\Phi] \in \mathbb{R}^{N \times m}$ and $[\Psi] \in \mathbb{R}^{n_s \times m}$ are orthonormal matrices and $\{s\} = (s_1, s_2, \ldots, s_m)^T \in \mathbb{R}^m$ is the vector of singular values in descending order.

In the POD framework, $[S]$ is a matrix of observations represented by column vectors.

$$[S] = [\{u_1\}, ..., \{u_{n_s}\}] \in \mathbb{R}^{N \times n_s}, \tag{2}$$

and the RB $[\Phi^k]$ of size k is obtained by taking the first k columns of $[\Phi]$ interpreted as the main features of $[S]$ in decreasing order of importance. This RB, by construction, spans the vector space of dimension k that best approximates the data, in the sense that it minimizes the projection error given the desired precision $\epsilon_{rb} \in [0, 1]$,

$$k = \underset{l \in \mathbb{N}}{\text{argmin}} \{\epsilon_{sv}^l \leq \epsilon_{rb}\} \tag{3}$$

$$\epsilon_{sv}^l = \frac{|| [S] - [\Phi^l][\Phi^l]^T[S] ||_F}{|| [S] ||_F} = \sqrt{\frac{\sum_{i=l+1}^{m} s_i^2}{\sum_{j=1}^{m} s_j^2}} \tag{4}$$

with $|| [\bullet] ||_F = \sqrt{\sum_{i,j} [\bullet]_{ij}^2}$ being the Frobenius norm of a matrix.

When POD is applied to FE models, the snapshot matrix may become prohibitively large. To avoid storing $[S]$, the incremental SVD approach updates the RB 'on-the-fly', as and when a new observation is available, which also avoids the need to store snapshots. In addition, truncation could be performed in-between the updates, thus keeping the size of the RB reasonably small while also making room for a reduction in the number of floating point operations needed.

The following subsection 1.1 presents the mathematical identity that allows the incremental computation of the SVD, while 'on-the-fly' snapshot selection and truncation are presented in subsection 1.2.

1.1 Incremental enrichment

As mentioned in the previous section, the incremental SVD [8] computes the decomposition (1) of $[S]$ *without* storing the entire matrix, by updating the RB as and when a new observation is available.

The snapshot matrix at the i^{th} iteration is given by

$$[S^{(i)}] = [\{u_1\}, ..., \{u_i\}] \in \mathbb{R}^{N \times i}.$$

The same matrix at the following iteration with one more observation appended to it may be written:

$$[S^{(i+1)}] = [S^{(i)}, \{u_{i+1}\}] \in \mathbb{R}^{N \times (i+1)}. \tag{5}$$

The decomposition of $[S^{(i)}]$ is given by:

$$[S^{(i)}] = [\Phi^{(i)}][\text{diag}(\{s^{(i)}\})][\Psi^{(i)}]^T \tag{6}$$

with

- $[\Phi^{(i)}] \in \mathbb{R}^{N\times k}$ an orthonormal matrix (i.e. $[\Phi^{(i)}]^T[\Phi^{(i)}] = [Id]_{k\times k}$)
- $[\Psi^{(i)}] \in \mathbb{R}^{i\times k}$ an orthonormal matrix (i.e. $[\Psi^{(i)}]^T[\Psi^{(i)}] = [Id]_{k\times k}$)
- $\{s^{(i)}\} = (s_1^{(i)}, \ldots, s_k^{(i)})^T \in \mathbb{R}^i$ with $s_j^{(i)} \neq 0$, $\forall j \in [\![1, k]\!]$

At the $(i+1)^{th}$ iteration the SVD of $[S^{(i)}]$ is known as well as N and k. The projection of a new observation $\{u_{i+1}\} \in \mathbb{R}^N$ on the current basis $[\Phi^{(i)}]$ is given by $\{\tilde{u}\} = [\Phi^{(i)}][\Phi^{(i)}]^T\{u_{i+1}\}$. The incremental SVD updates the decomposition of $[S^{(i)}]$ according to $[S^{(i+1)}] = [\Phi^{(i+1)}][\mathrm{diag}(\{s^{(i+1)}\})][\Psi^{(i+1)}]^T$. Here the following identity [7] is used:

$$\begin{aligned}\left[[S^{(i)}], \{u_{i+1}\}\right] &= \left[[\Phi^{(i)}][\mathrm{diag}(\{s^{(i)}\})][\Psi^{(i)}]^T, \{u_{i+1}\}\right] \\ &= \left[[\Phi^{(i)}], \{\xi\}\right] \underbrace{\begin{bmatrix}[\mathrm{diag}(\{s^{(i)}\})] & [\Phi^{(i)^T}]\{u_{i+1}\} \\ \{0_k\}^T & \alpha\end{bmatrix}}_{:=[Q]} \begin{bmatrix}[\Psi^{(i)}] & \{0_i\} \\ \{0_k\}^T & 1\end{bmatrix}^T, \end{aligned} \tag{7}$$

where

- $\alpha = \|\{u_{i+1}\} - \{\tilde{u}_{i+1}\}\|_2$
- $\{\xi\} = \frac{\{u_{i+1}\}-\{\tilde{u}_{i+1}\}}{\alpha}$.

Note, that $\left[[\Phi^{(i)}], \{\xi\}\right]$ and $\begin{bmatrix}[\Psi^{(i)}] & \{0_i\} \\ \{0_k\}^T & 1\end{bmatrix}$ are orthonormal matrices. The main advantage of the incremental SVD lies in the economical diagonalization of the matrix $[Q]$.

$$[Q] = [\Phi'][\mathrm{diag}(\{s'\})][\Psi']^T \tag{8}$$

yielding

$$[S^{(i+1)}] = [\Phi^{(i+1)}][\mathrm{diag}(\{s^{(i+1)}\})][\Psi^{(i+1)}]^T \tag{9}$$

where

- $[\Phi^{(i+1)}] = \left[[\Phi^{(i)}], \{\xi\}\right][\Phi']$
- $\{s^{(i+1)}\} = \{s'\}$
- $[\Psi^{(i+1)}] = \begin{bmatrix}[\Psi^{(i)}] & \{0_i\} \\ \{0_k\}^T & 1\end{bmatrix}[\Psi']$

$[\Phi^{(i+1)}] \in \mathbb{R}^{N\times k}$ and $[\Psi^{(i+1)}] \in \mathbb{R}^{(i+1)\times k}$ are orthonormal matrices by construction, being products of orthonormal matrices each. Thus, the newly formed decomposition of $[S^{(i+1)}]$ is in fact its SVD.

Savings in computational resources (compared to the traditional SVD) occur during the diagonalization of [Q], where simplifications are possible due to its particular form. Note that $[Q] \in \mathbb{R}^{k\times k}$ is a small "half-arrowhead matrix", meaning it is nearly diagonal except for its last column, which renders the SVD of $[Q]$, which is achieved by bi-diagonalization and the Golub-Kahan algorithm [17], computationally inexpensive.

1.2 'On-the-fly' snapshot selection and truncation

The primary motivation for using the incremental SVD is to avoid the manipulation of a large amount of possibly redundant data while building a low-rank approximation. The two parameters ϵ_{svd} and k_{max} of this method first detailed in [28] allow for these savings:

- At the beginning of the $i+1^{th}$ iteration, the new observation $\{u_{i+1}\}$ may be rejected, if already well represented by the RB $[\Phi^{(i)}]$, so as to avoid unnecessary computations on redundant data. In the state-of-the-art algorithm [8], this is controlled by the parameter ϵ_{svd} and the new observation is skipped if $\alpha^2 \leq \epsilon_{svd}$.

- At the end of an iteration, $[S^{(i+1)}]$ decomposition (9) may be truncated to keep it as small as possible, accelerating subsequent iterations. Routinely [28], the basis is truncated 'on-the-fly' when its size exceeds a pre-determined value given by the parameter k_{max}.

Tuning ϵ_{SVD} and k_{max} enables significant computational savings, compared with a single-pass SVD. When the single-pass SVD is used in the POD framework to build an RB, the right singular space $[\Psi]$ is generally not used. Such an SVD of a matrix $[S] \in \mathbb{R}^{N \times n_s}$ will need $O(Nm^2)$ floating point operations and $O(2mN + m)$ memory. In the incremental SVD, if the RB's size does not exceed k, the method would only require $O(mNk)$ operations and $O(Nk + k)$ memory. In applications, the size k of the RB is very small in comparison with the FOM's size N and the number of observations n_s. The fundamental hypothesis of MOR is that the dimension of the underlying manifold, where the discretized solution of a computational mechanics problem "lives" and evolves, is small compared to the number of degrees of freedom. Therefore, incremental SVD is very promising for MOR. Finally, 'on-the-fly' snapshot selection would prevent computation on redundant data and thus limit the number of snapshots n_s processed, reducing the complexity as shown.

However, these two features also induce loss of information about the singular values rendering the exact final approximation error (4) inaccessible. An estimator for this error is developed and tested in the following section.

2 Error estimator

In the original incremental SVD, 'on-the-fly' snapshot rejection and basis truncation prevent the computation of all singular values, meaning that it is impossible to compute the truncation error (4). The modified version of the incremental SVD, proposed in this paper, tracks the lost information (due to the missed singular values) using two variables, ϵ_{in} and ϵ_{out}.

Since the basis is truncated 'on-the-fly', the singular value associated with the truncated mode is lost, therefore, ϵ_{out} is incremented by the square of the skipped singular value:

$$\epsilon_{out} = \epsilon_{out} + s_{i+1}^2. \tag{10}$$

If a snapshot $\{u_{i+1}\}$ has been rejected 'on-the-fly', we can no longer calculate changes in the singular values, ϵ_{in} is incremented by the norm of projection of $\{u_{i+1}\}$ on $[\Phi^{(i)}]$:

$$\epsilon_{in} = \epsilon_{in} + \|\{\tilde{u}_{i+1}\}\|_2^2, \tag{11}$$

and ϵ_{out} is then incremented by the error of projection of the skipped snapshot:

$$\epsilon_{out} = \epsilon_{out} + \alpha^2. \tag{12}$$

These two variables are used to compensate for the unavailable singular values, yielding the following expression for the error estimator :

$$\epsilon_{est} = \sqrt{\frac{\epsilon_{out}}{\|\{s\}\|_2^2 + \epsilon_{in} + \epsilon_{out}}}. \tag{13}$$

An essential condition on the error estimator is that it must be larger than the actual error. In the event that the new observation is used for enriching the RB, the error estimator does not induce any error since the square of the exact singular value associated with truncated mode is incremented in ϵ_{out}. On the other hand, if the new observation is not selected, then the error estimator satisfies the following condition:

$$\mathcal{P}^{(i)} = \left\{ \frac{\|[S^{(i)}] - [\Phi^{(i)}][\Phi^{(i)}]^T[S^{(i)}]\|_F^2}{\|[S^{(i)}]\|_F^2} \leq \frac{\epsilon_{out}^{(i)}}{\|\{s^{(i)}\}\|_2^2 + \epsilon_{in}^{(i)} + \epsilon_{out}^{(i)}} = (\epsilon_{est}^{(i)})^2 \right\}. \tag{14}$$

This ensures that the estimator is an upper bound on the actual error. A straightforward recursive proof of the above property is given below:

initialization: For the first iteration of the algorithm, $i = 1$, the error is null, $\epsilon_{in} = \epsilon_{out} = 0$ and no information has been lost (yet) due to either rejection or truncation.

induction: If we assume that $\mathcal{P}^{(i)}$ is true and the new observation $\{u_{i+1}\}$ has been rejected, then the RB is left unchanged ($[\Phi^{(i+1)}] = [\Phi^{(i)}]$ and $\{s\}^{(i+1)} = \{s\}^{(i)}$), $\epsilon_{out}^{(i+1)} = \epsilon_{out}^{(i)} + \alpha^2$ and $\epsilon_{in}^{(i+1)} = \epsilon_{in}^{(i)} + \|[\Phi][\Phi^{(i)}]^T\{u_i\}\|_2^2$. In which case, using identity (5):

$$\frac{\|[S^{(i+1)}] - [\Phi^{(i+1)}][\Phi^{(i+1)}]^T[S^{(i+1)}]\|_F^2}{\|[S^{(i+1)}]\|_F^2} = \frac{\|([Id] - [\Phi^{(i+1)}][\Phi^{(i+1)}]^T)[S^{(i)}, \{u_{i+1}\}]\|_F^2}{\|[S^{(i)}, \{u_{i+1}\}]\|_F^2},$$

using the additivity of the Frobenius norm, the above expression is rewritten

$$= \frac{\|([Id] - [\Phi^{(i+1)}][\Phi^{(i+1)}]^T)[S^{(i)}]\|_F^2 + \alpha^2}{\|[S^{(i)}]\|_F^2 + \|\{u_{i+1}\}\|_2^2},$$
$$= \frac{\|([Id] - [\Phi^{(i)}][\Phi^{(i)}]^T)[S^{(i)}]\|_F^2 + \alpha^2}{\|[S^{(i)}]\|_F^2 + \|\{u_{i+1}\}\|_2^2},$$

and finally, $\mathcal{P}^{(i)}$ is used

$$\leq \frac{\epsilon_{out}^{(i)} + \alpha^2}{\|\{s^{(i)}\}\|_2^2 + \epsilon_{in}^{(i)} + \epsilon_{out}^{(i)} + \|\{u_{i+1}\}\|_2^2} = \frac{\epsilon_{out}^{(i+1)}}{\|\{s^{(i+1)}\}\|_2^2 + \epsilon_{in}^{(i+1)} + \epsilon_{out}^{(i+1)}} = \epsilon_{est}^{(i+1)},$$

which proves $\mathcal{P}^{(i+1)}$.

3 Proposed algorithm

Algorithm 1 incorporates the proposed incremented error estimator and both the corresponding 'on-the-fly' truncation and snapshot selection. ϵ_{rb} is the threshold on the singular value truncation error and is the only input argument that the user needs to set in this version of the algorithm. All other arguments are initialized from steps 2 to 6 and incremented during subsequent enrichments.

The main feature of the algorithm is the use of the proposed error estimator for the 'on-the-fly' snapshot selection at steps 8 and 9 as well as 'on-the-fly' truncation at steps 17 to 23. The philosophy behind the algorithm is speeding up computations by rejecting the maximal number of snapshots and truncating the RB as soon as possible, in each enrichment, while *still* controlling the overall singular value error of truncation and avoiding the necessity to set the problem-dependent parameters ϵ_{svd} and k_{max}.

One potential issue is the loss of orthonormality of the RB due to subsequent enrichments (line 13). While this does not change the range of the RB, it does, however, prevent a good projection of the new observation as $\|\{u_i\}\|_2^2 = \|[\Phi][\Phi]^T\{u_i\}\|_2^2 + \|\{u_i\} - [\Phi][\Phi]^T\{u_i\}\|_2^2$ is not necessarily true if $[\Phi]$ is not orthonormal. This is why re-orthonormalization is performed (lines 23 to 26) when the scalar product of the first and last columns of the current RB is larger than a threshold (ϵ_{orth}), typically set to 10^{-14}. This is performed using the highly parallelizable Modified Gram Schmidt (MGS) algorithm, but may also be implemented by QR factorization.

Another important remark concerns "centering" the data. In some POD applications [38], the snapshot matrix is centered before the spectral approximation is computed, motivated, in part, by the statistical point of view interpreting $[S]^T[S]$ as the covariance matrix of the data. In our applications, data points are not centered as simply applying a "lift" to the POD approximation (19) will not improve the precision over adding a column to $[\Phi]$ That said, [7, 28] use 'a posteriori' centering for the incremental SVD. To this end, the right subspace $[\Psi]$ of the decomposition must be computed and stored, the size of which scales up with the number of observations and cannot be truncated 'on-the-fly'. In our experience, 'a posteriori' centering is not compatible with 'on-the-fly' truncation. Truncated/selected basis vectors as well as rejected observations during the enrichment steps would not be the same, had they been performed using snapshots that were centered from the beginning. Moreover, 'on-the-fly' truncation would deteriorate the snapshot matrix reconstruction and render the 'a posteriori' centering inaccurate. The centered POD is thus more straightforward with the regular than the incremental SVD.

4 Numerical tests

In this section, we validate the proposed incremental SVD approach using a Taylor beam impact model. We first give a short review of the POD method in section 4.2, followed by testing the

Algorithm 1: Proposed incremental SVD

Input: $[\Phi]$, $\{s\}$, $\{u_i\}$, ϵ_{in}, ϵ_{out}, ϵ_{rb}
Output: $[\Phi]$, $\{s\}$, ϵ_{in}, ϵ_{out}

if $(size([\Phi], 2) == 0)$ **then**
  if $(\|\{u_i\}\|_2 > 0)$ **then**
    $\{s\} = \|\{u_i\}\|_2$
    $[\Phi] = \frac{\{u_i\}}{\|\{u_i\}\|_2}$
    $\epsilon_{in} = \epsilon_{out} = 0$
  end
else
  $error = \frac{\epsilon_{out} + \|\{u_i\} - [\Phi][\Phi]^T\{u_i\}\|_2^2}{\epsilon_{in} + \epsilon_{out} + \|\{s\}\|_2^2 + \|\{u_i\}\|_2^2} - \epsilon_{rb}^2$
  if $(error > 0)$ **then**
    $\alpha = \|\{u_i\} - [\Phi][\Phi]^T\{u_i\}\|_2$
    $[Q] = \begin{bmatrix} [\text{diag}(\{s\})] & [\Phi]^T\{u_i\} \\ 0 & \alpha \end{bmatrix}$
    $[[\Phi'], \{s'\}, [\Psi']] = \text{SVD}([Q])$
    % RB is expanded
    $[\Phi] = [[\Phi], \frac{\{u_i\} - [\Phi][\Phi]^T\{u_i\}}{\|\{u_i\} - [\Phi][\Phi]^T\{u_i\}\|_2}][\Phi']$
    $\{s\} = \{s'\}$
    % RB is truncated with new error estimator
    $error = \frac{\epsilon_{out} + s(end)}{\epsilon_{in} + \epsilon_{out} + \|\{s\}\|_2^2} - \epsilon_{rb}^2$
    while $(error < 0)$ **do**
      $\epsilon_{out} = \epsilon_{out} + s(end)$ **equation** (10)
      $\{s\} = \{s\}(1 : (end - 1))$
      $[\Phi] = [\Phi]_{:,1:(end-1)}$
      $error = error + \frac{s(end)}{\epsilon_{in} + \epsilon_{out} + \|\{s\}\|_2^2}$
    end
    if $([\Phi]_{:,1}^T [\Phi]_{:,end} > \epsilon_{orth})$ **then**
      % RB has lost numerical orthogonality and is re-orthonormalized
      $[\Phi] =$ **MGS**$([\Phi])$
    end
  else
    $\epsilon_{out} = \epsilon_{out} + \|\{u_i\} - [\Phi][\Phi]^T\{u_i\}\|_2^2$ **equation** (12)
    $\epsilon_{in} = \epsilon_{in} + \|[\Phi][\Phi]^T\{u_i\}\|_2^2$ **equation** (11)
  end
end

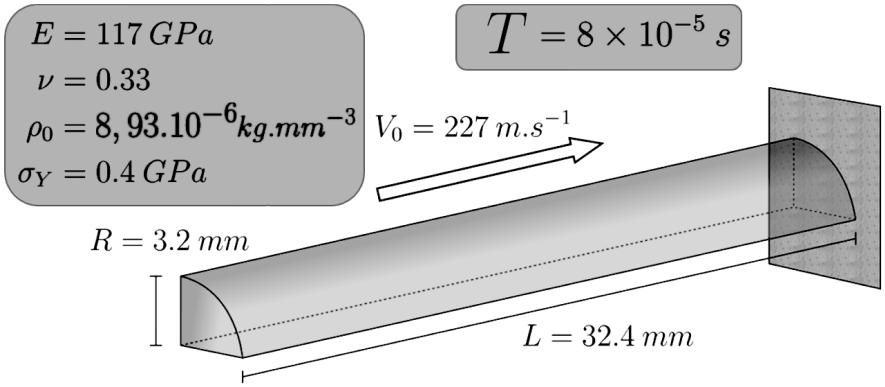
approach for computation time and estimated training data approximation error in section 4.3. The proposed incremental SVD is benchmarked against the state-of-the-art incremental SVD first in section 4.4 in terms of computation time and *offline* training data approximation error and then in section 4.5 in terms of work of internal forces approximation in the *online* reduction phase. Finally, section 4.6 describes the scalability of the proposed incremental SVD with respect to the dimensions of the snapshots matrix and targeted error of approximation.

NOTE: The Taylor beam impact test was performed with the proposed incremental SVD implementation using the state-of-the-art crash simulation code Altair Radioss [2].

4.1 Taylor beam impact

The tests have been performed on a snapshot matrix $[S]$ obtained from a Taylor beam impact simulation, using the model shown in Figure 1. The beam is made of an elasto-plastic steel

Figure 1: Taylor beam test case



of density $\rho = 8.93 * 10^{-9}\ T.mm^{-3}$, Young's modulus $E = 117000\ MPa$ and Poisson's ratio $\nu = 0.33$. The hardening rule is described by the Johnson-Cook law (/MAT/PLAS_JOHNS in Altair Radioss), neglecting temperature and strain rate effects

$$\sigma_{eq} = \sigma_Y + b * \varepsilon_p^n, \tag{16}$$

with σ_{eq} the equivalent stress and ε_p the equivalent plastic strain. Other parameters are plastic yield stress $\sigma_Y = 400\ MPa$, plasticity hardening coefficient $b = 100$ and plasticity hardening exponent $n = 1$. 8-node solid elements with one integration point are used to discretize the model into $N = 8514$ dofs. The central difference method is used for time integration along with the lumped mass approach, yielding a diagonal mass matrix.

$n_s = 236$ incremental displacements snapshots have been taken every 50 time steps over the total simulation time $T = 8 * 10^{-5}$ seconds.We begin with a brief review of the Galerkin PROM in the next subsection, followed by evaluating the RB's performance during the *online* reduction phase later in this section.

4.2 POD review

Full-order model

In nonlinear structural dynamics, the semi-discretized finite element formulation takes on the following form:

$$[\mathbb{M}]\{\ddot{u}(t)\} + \{f_{int}\}(\{u(t)\}, t) = \{f_{ext}(t)\}, \tag{17}$$

where $\{u(t)\} \in \mathbb{R}^N$ is the vector of nodal displacements (for all nodes and along three directions) unknown at time t. NOTE: In general, $\{u(t)\}$ may also contain shell rotations, however, that is not the primary focus of this paper. N denotes the number of degrees of freedom (dofs), the size of the FOM. $[\mathbb{M}] \in \mathbb{R}^{N \times N}$ is a symmetric, real, positive definite mass matrix.

The FOM may depend on n_p parameters $(p_i)_{1\leq i\leq n_p}$ which are generally shell thicknesses and material properties in vehicle crash simulations. The first step consists of extracting observations, or snapshots, at various simulation times and for different parameter value sets. For convenience, the observations time t and observation parameter values $(p_1, p_2, \ldots, p_{n_p})^T$ are placed together in a vector $\{c\} = (t, p_1, p_2, \ldots, p_{n_p})^T$, that we refer to as the training configuration.

Reduced basis

The FOM solutions of training configurations $(\{c_i\})_{1\leq i\leq n_s}$ are collected in the snapshot matrix $[S] \in \mathbb{R}^{N\times n_s}$

$$[S] = [\{u(\{c_1\})\}, \{u(\{c_2\})\}, \ldots, \{u(\{c_{n_s}\})\}]. \tag{18}$$

In the second step, we compute a basis for the vector space of minimal dimension k that is capable of approximating the data in $[S]$ to a user-defined precision ϵ_{rb}. This new vector space of dimension k is spanned by the columns of $[\Phi] \in \mathbb{R}^{N\times k}$

$$[\Phi] = [\{\phi_1\}, \{\phi_2\}, \ldots, \{\phi_k\}] \in \mathbb{R}^{N\times k}.$$

The POD approximation is then written as:

$$\{u(t)\} \approx \{\tilde{u}(t)\} = \sum_{i=1}^{k} \alpha_i(t)\{\phi_i\} = [\Phi]\{\alpha(t)\} \tag{19}$$

$\{\alpha(t)\} = (\alpha_1(t), \alpha_2(t), \ldots, \alpha_k(t))^T \in \mathbb{R}^k$ is the vector of ROM unknowns.

$[\Phi]$ is obtained by computing a low-rank representation of the snapshot matrix $[S]$. The RB size k is given by the desired precision ϵ_{rb} of the low-rank representation of $[S]$.

Reduced model

The ROM is constructed by injecting the approximation (19) into (17) and projecting the resulting equations on $[\Phi]^T$.

$$[\tilde{\mathbb{M}}]\{\ddot{\alpha}(t)\} + [\Phi]^T\{f_{int}([\Phi]\{\alpha(t)\}, t)\} = [\Phi]^T\{f_{ext}(t)\} \tag{20}$$

We introduce the following notation: $[\tilde{\mathbb{M}}] = [\Phi]^T[\mathbb{M}][\Phi] \in \mathbb{R}^{k\times k}$. The size of the reduced model (20) is k, which is much smaller than the size N of the original full order model (17). However, the use of POD for the reduction of explicit structural dynamics models induces computational overhead during the *online* phase due to the Galerkin projection used to build the PROM. Solution information at each node is mandatory in order to compute internal variables in each element. The reduced unknown $\{\alpha\}$ does not provide information explicitly at each node and the full-scale approximation needs to be computed. In order to map reduced unknowns $\{\alpha\}$ to the full-scale approximation $\{\tilde{u}\}$, defined in equation (19), supplementary matrix vector multiplications are performed at each time step, which negatively affects the *online* speedup. Thus, we require additional reduction methods to obtain a computational speedup during the *online* reduction phase in explicit nonlinear POD applications. These methods attempt to approximate the projected nonlinear internal variables operator $[\Phi]^T\{f_{int}([\Phi]\bullet, t)\}$. The most popular methods to achieve this approximation are hyper-reduction ([14, 18, 34]) and the DEIM ([9, 12, 36]).In this work, we opted for the ECSW hyper-reduction method ([14]).

Another important feature of the ROM (20) is that the stability condition for optimal time step is also projected. In the event that the nodal time step is used, the projection of the CFL condition on $[\Phi]$ leads to a larger time step, some theoretical developments on this subject may be found in [5].

4.3 Proposed incremental SVD tests

In this section, we present some preliminary tests to validate the proposed algorithm, wherein we compare the error estimator (13) with the 'actual' error of approximation (4) (which requires

storage of the snapshot matrix). Computation times using different values of user-defined error ϵ_{rb} are also compared with those for the single-pass SVD from the Fortran Intel MKL package. Next, we present a comparison of the performance of these RBs in the *online* reduction phase of the Taylor beam impact.

For these tests, we employ the matrix $[S] \in \mathbb{R}^{8514\times 236}$ of incremental displacements snapshots obtained using the full-order Taylor beam model runs. The snapshot selection and 'on-the-fly' truncation are now driven by keeping the error estimator (13) below the single target error ϵ_{rb}, in place of ϵ_{svd} and k_{max}, for the state-of-the-art incremental SVD. We test the method for $\epsilon_{rb} \in [10^{-8}, 1]$.

Figure 2 shows the RB's training data reconstruction error (4) and associated error estimator (13) values as a function of ϵ_{rb} (estimated errors in gray, real errors in black). Computation times of those RBs are compared with that of the single-pass SVD on Figure 3. The single-pass SVD used here is the DGESVD from the Intel MKL library provided with Intel FORTRAN compilers (version 12.1.3.300) and is used to compute only the first $m = \min(n_s, N) = 236$ left singular vectors and associated singular values which correspond to the flags JOBU = 'S' and JOBVT = 'N' (c.f. DGESVD Documentation [1]).

Figure 2: Incremental SVD error estimator ϵ_{est} in function of the target error ϵ_{rb} for the Taylor beam impact. Compared to the measured error ϵ_{sv}^k and the target error.

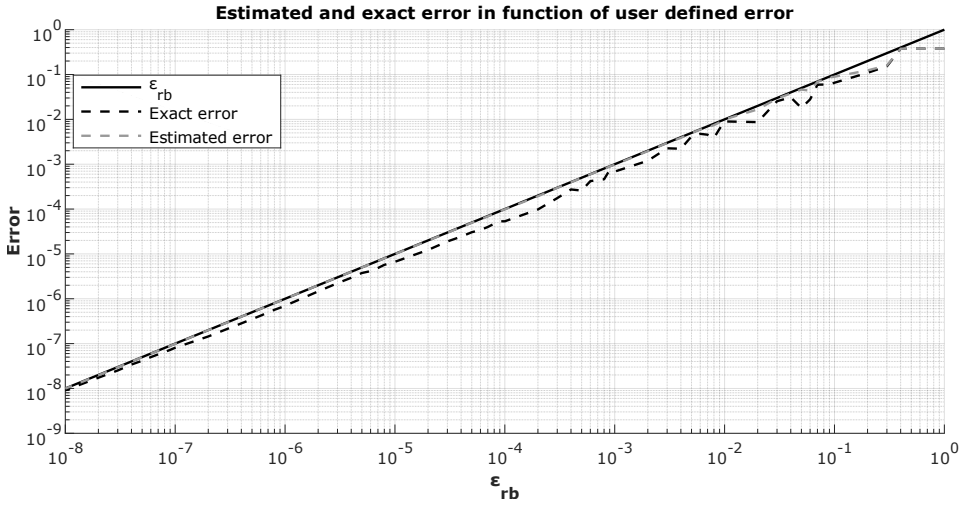


Figure 3: Incremental SVD computation time in function of the target for Taylor beam impact.

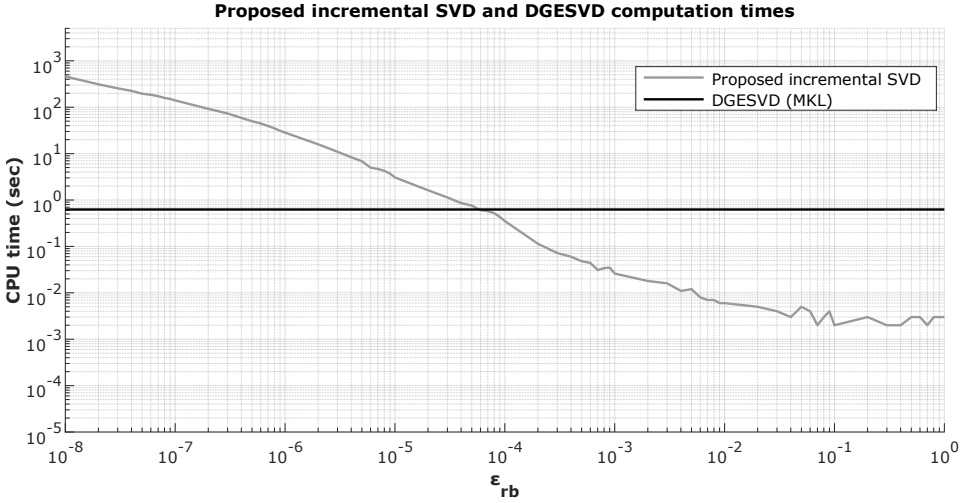


Figure 2 clearly shows that the error estimator ϵ_{est} is always lower than the target ϵ_{rb} and

higher than the actual approximation error ϵ_{sv}^k. The smaller is the desired ϵ_{rb}, the more precise is the error estimator (13) and the 'actual' RB approximation error (4).

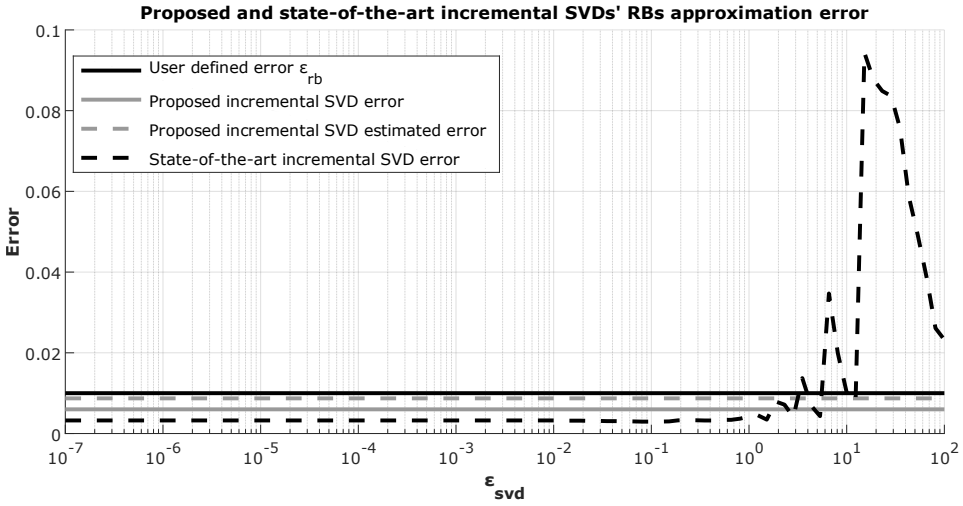
The computation times reported in Figure 3 show that our algorithm is faster than the single-pass SVD up to a precision of $\epsilon_{rb} = 10^{-4}$. However, the incremental SVD is not designed for a single-pass SVD since the computation times tends to skyrocket with very low ϵ_{rb}. It is also important to note that given a targeted precision ϵ_{rb}, the incremental SVD may build a larger RB than that obtained using the single-pass SVD, since the error is only *estimated* and, in reality, may well be lower than the target error.

4.4 Comparison with state-of-the-art incremental SVD

Comparing the state-of-the-art with the proposed incremental SVD is not a straightforward task as the methods do not involve the same parameters. In particular, as explained in section 1.2, the state-of-the-art incremental SVD [28] has various tolerances that have to be set to appropriate values.

We use the same snapshot matrix $[S] \in \mathbb{R}^{8514\times 236}$ within this section. The algorithm is tested in the following manner: First, the RB (of size $k = 3$) is computed using the proposed algorithm with a user specified error $\epsilon_{rb} = 10^{-2}$. In the second phase, state-of-the-art incremental SVD runs are performed for $k_{max} = 3$ and hundred different values of ϵ_{svd} *logarithmically* spaced between 10^{-7} and 10^{2}. The re-orthonormalization parameter is set to $\epsilon_{orth} = 10^{-14}$ in both algorithms. The resulting computation times are given in Figure 5 and approximation errors in Figure 4.

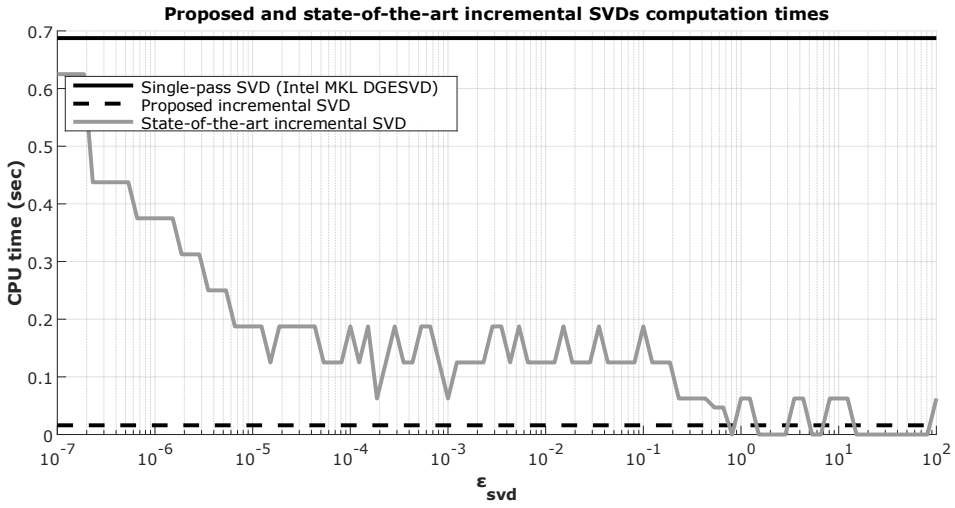
Figure 4: Proposed and state-of-the-art incremental SVDs approximation errors for reduced basis of size $k = 3$ on snapshots from the Taylor beam impact simulations



In this example, both the proposed and state-of-the-art incremental SVD outperform the single-pass SVD from the Fortran Intel MKL package. This is because the data of 236 vectors may be approximated with a precision of $\epsilon_{rb} = 10^{-2}$ with only $k = 3$ basis vectors. This "very low-dimensional" approximation works in favor of the two incremental methods.

Using properly-chosen parameters within the state-of-the-art SVD ($k_{max} = 3$ and $\epsilon_{svd} = 2$), the incremental methods' performances are both comparable. However, it is not clear how to set these two parameters in practice, since they depend on the data dimensionality and magnitude as well as the target precision. k_{max} has been set to the minimal value allowing an error of approximation smaller than 10^{-2}. The results in Figure 4 show that if ϵ_{svd} is too large, then the desired error cannot be attained. On the other hand, if its value is too small, we see from Figure 5 that computation time would be significantly higher than that of the proposed version. Moreover, the state-of-the-art SVD approximation errors given in Figure 4 have been computed *a posteriori*, by storing *all* the data in memory, which is not desirable in practice.

Figure 5: Proposed and state-of-the-art incremental SVDs computation times for reduced basis of size $k = 3$ on snapshots from the Taylor beam impact simulations



In this example, the proposed incremental SVD shows better performance in terms of precision and computation time without having to set any other parameter than the targeted singular value approximation error (13).

4.5 Online comparison

We now compare the performance of the RBs during the *online* reduction phase. For this, we consider 5 RBs of size $k = 6$, the first being obtained with the truncated single-pass SVD, the next three using the state-of-the-art incremental SVD with parameters $\epsilon_{svd} \in \{10^{-14}, 10^{-6}, 10^{-4}\}$ and $k_{max} = 6$ and the last RB corresponding to our proposed incremental SVD with $\epsilon_{rb} = 2 * 10^{-3}$.

Online performances are evaluated in terms of work of internal forces reconstruction. Internal work at a given time is defined by

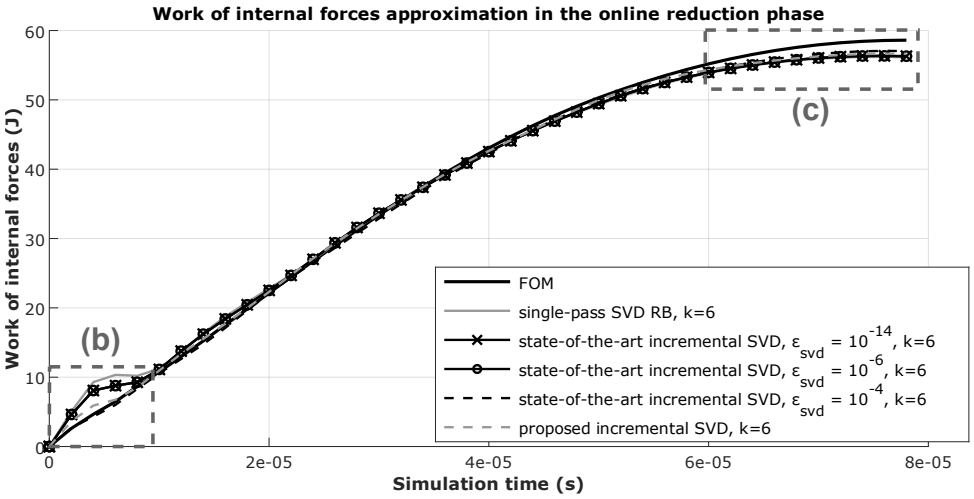
$$W_{int}(t) = \int_{\Omega} \underline{\underline{\varepsilon}} : \underline{\underline{\sigma}} dV \tag{21}$$

with $\underline{\underline{\varepsilon}}$ the strain tensor and $\underline{\underline{\sigma}}$ the stress tensor. Denoting W_{int} and $\tilde{W}_{int}$ the work of internal forces in the FOM and in the *online* reduction phase, respectively, the error ϵ_{int} considered here is

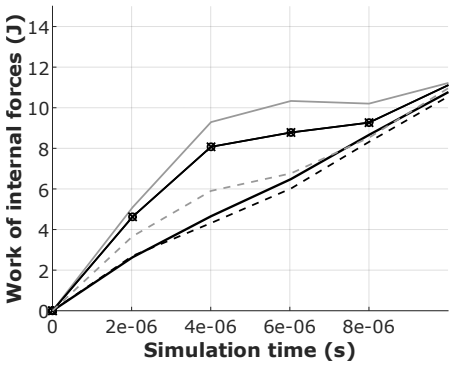
$$\epsilon_{int} = \frac{\int_0^T \mid W_{int}(t) - \tilde{W}_{int}(t) \mid dt}{\int_0^T W_{int}(t) dt}. \tag{22}$$

Figure 6 plots the variation of the work of internal forces during the reduction phase for all 5 RBs as well as for the FOM.

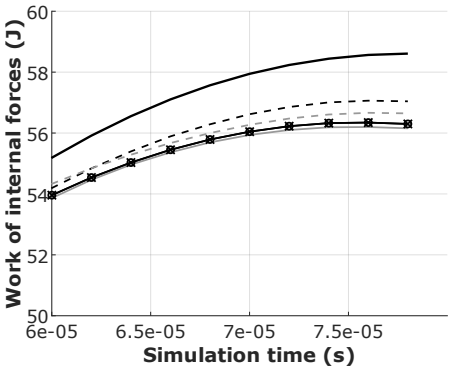
Even though the single-pass SVD is optimal as far as the singular value truncation error during the *offline* training phase is concerned, it is clearly not optimal for work of internal forces reconstruction during the *online* POD reduction phase. When snapshots are uniformly sampled from the simulation, SVD tends to preferentially emphasize those corresponding to later simulation times since they are generally larger in magnitude and redundant. On the other hand, snapshots taken at earlier simulation times tend to be smaller in magnitude and unique, therefore, they are given less importance in the SVD. As a consequence, POD approximation error is often larger during the early stages of the simulation, as seen on this example. This behavior is seen to a lesser degree in the incremental SVD as 'on-the-fly' selection avoids the stacking of redundant, larger magnitude observations by only selecting snapshots that introduce new information. In Figure 6, RBs obtained with the state-of-the-art incremental SVD yield a better approximation of the work of internal forces at the beginning of the reduced simulation. The RBs obtained with $\epsilon_{svd} \leq 10^{-6}$ are identical. Increasing ϵ_{svd} from 10^{-6} to 10^{-4} results in an even better approximation. However, for $\epsilon_{svd} > 10^{-4}$, it is not possible to build RBs of size $k = 6$ as the number of snapshots selected



(a) Work of internal forces in the entire simulation



(b) Zoom on first simulation times



(c) Zoom on last simulation times

Figure 6: Comparison of the work of internal forces reconstruction in the *online* reduction phase on the Taylor beam test case. RBs of size $k = 6$ computed with the proposed incremental SVD, the state-of-the-art SVD and the single-pass SVD are used.

'on-the-fly' is insufficient. So $\epsilon_{svd} = 10^{-4}$ is the optimal value for this particular example. But this value is problem-dependent, and increasing the number of degrees of freedom N, the maximal size of the RB k_{max}, the number of snapshots n_s or switching unit system will change the optimal value for ϵ_{svd}. Although the state-of-the-art incremental SVD with optimal value of ϵ_{svd} yields the best RB in this example, the actual value itself is unknown in practice. The proposed incremental SVD involves a nearly optimal 'on-the-fly' selection criterion that only depends on the target approximation error, which is easier to use and *not* problem-dependent. The RB obtained with the proposed method approximates the work of internal forces almost as well as the state-of-the-art incremental SVD with $\epsilon_{svd} = 10^{-4}$.

We mentioned that, the singular value approximation error being only approximated in the proposed method, the RB's size may be larger than necessary. However, this is not an issue as results within this section show that the singular value approximation error does not guarantee the error incurred within the *online* reduction phase. As a matter of fact, RB computed with the proposed method performs better in the *online* reduction phase that the one computed using the single-pass SVD even though it produces a larger training data singular value approximation error. Finally, the test during the *online* POD phase shows a satisfactory performance. RB computed with the proposed method performs well in comparison with RBs of same sized obtained with state-of-the-art SVD and single-pass SVD using the same training data set.

4.6 Proposed incremental SVD scalability

In this subsection, we test the computation time for algorithm 1 against the single-pass SVD DGESVD from the Intel MKL FORTRAN library using snapshots matrices obtained from two different Taylor beam models. In Figure 7, the computation times are plotted for $[S] \in \mathbb{R}^{4752 \times n_s}$ and for $n_s \in [\![100, 1000]\!]$. A larger model of the Taylor beam impact has been used for Figure 8 in which $[S] \in \mathbb{R}^{18810 \times n_s}$ and $n_s \in [\![500, 8000]\!]$. The snapshots in this figure have been uniformly sampled from the simulation time interval for all values of n_s. No binary file manipulation times are taken into consideration here.

Figure 7: Taylor beam impact, Scalability of incremental vs. single-pass SVD, $N = 4752$

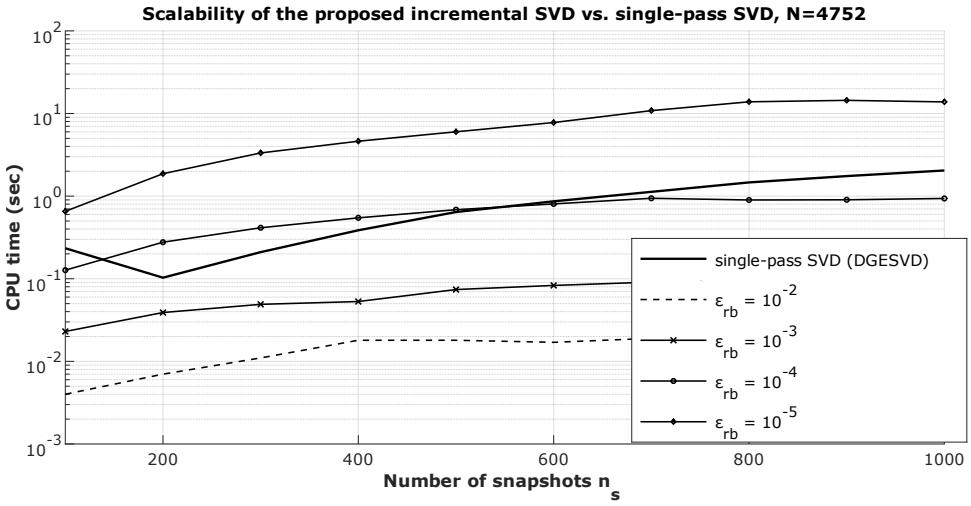
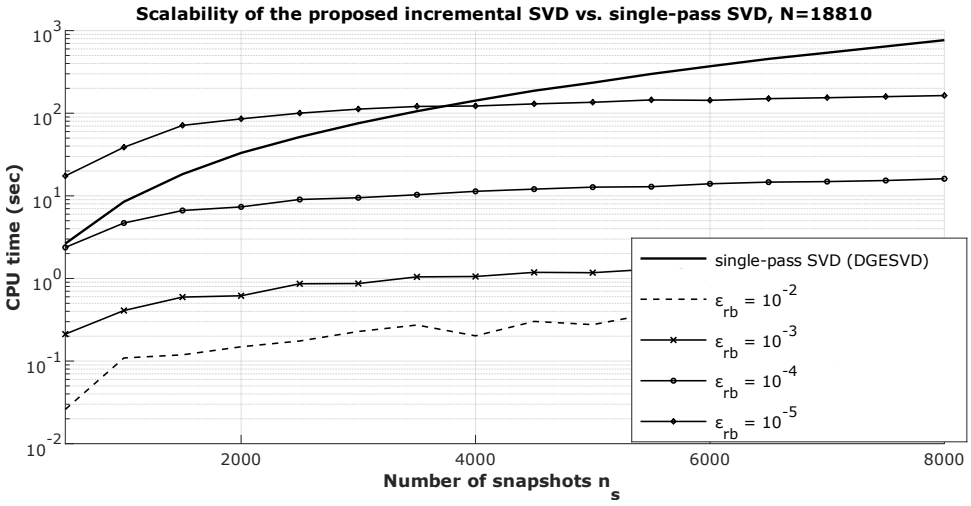


Figure 8: Taylor beam impact, Scalability of incremental vs. single-pass SVD, $N = 18810$



The complexity of the single-pass SVD (DGESVD) scales up according to $m = \min(N, n_s)$ and thus its computation time plateaus when there are more snapshots than degrees of freedom ($n_s > N$). However, the computation time of the DGESVD grows quadratically with the number n_s of snapshots until $n_s = N$. On the other hand, the incremental SVD's complexity scales up according to the dimensionality of the approximating vectorial space, hence, the computational time stabilizes very fast. In fact, for a given target precision ϵ_{rb}, the incremental SVD computation time grows quadratically with the number of snapshots until $n_s = k$, where k is given by equation (3). This means that for an insufficient number of snapshots n_s and a high target precision ϵ_{svd}, the proposed incremental SVD presents a poorer performance than the DGESVD, until we reach the minimum number of snapshots given by equation (3) after which the proposed approach shows superior

performance. In figure 8, the incremental SVD is seen to perform better than the single-pass SVD for precision up to 10^{-4}, independently of the number of snapshots n_s. For higher precision 10^{-5}, the curves are seen to intersect at $n_s = 3800$, with the incremental SVD outperforming the single-pass SVD afterwards. Note, that in the latter case, the incremental SVD's CPU time remain relatively steady after $n_s \approx 2000$, as opposed to the single-pass version. This behavior may be explained by the limited number of relevant basis vectors governing the physical system, confirming the fundamental hypothesis of reduced order modeling, as well as providing excellent scalability to our method.

5 Conclusion

This paper proposes an approach reducing memory usage and floating point operations, compared to the regular Singular Value Decomposition (SVD), by proceeding in an incremental fashion. By treating snapshots one after the other, it permits 'on-the-fly' selection for enriching and truncating the RB. By doing so, the method avoids computations on possibly redundant data, as well as unnecessary computations on data that will only be truncated later. The memory usage is minimal as only a truncated representation of the data and a single new observation need to be stored at any given time. As a result, this method is particularly efficient in approximating extensive data sets that have low underlying dimensionality, as is usually the case in computational structural dynamics.

The method is only driven by the desired singular value truncation error, avoiding parameters that depend on the nature and dimension of the data. Moreover, the use of this error estimator allows for improved snapshot selection and the 'on-the-fly' RB truncation, ultimately enhancing performance.

The only remaining parameter in our algorithm is the re-orthonormalization parameter, which has been set here to $\epsilon_{orth} = 10^{-14}$. The aim of the re-orthonormalization step between subsequent enrichments is to avoid incurring a significant error while computing $[\Phi][\Phi]^T\{u_i\}$. This means that the optimal value for ϵ_{orth} depends on the actual dimensions of the RB. Further work on optimizing this parameter should conceivably enable additional computational savings.

In the present work, results are given for a sequential algorithm. It goes without saying that a parallel implementation of the incremental SVD is possible following the lines proposed by [21].

Equipped with error estimator presented in this work, the proposed incremental SVD turns out to be more efficient and less problem-dependent than the traditional version. Using this error estimator in place of the usual tolerances makes the proposed method easy to use, and nearly optimal for training RBs for MOR. Within the POD framework, the proposed algorithm has been shown to outperform the original state-of-the-art incremental SVD as well as the single-pass SVD both in the *offline* training phase as well as the *online* reduction phase, for the presented Taylor beam impact test case.

Acknowledgments

This work is part of a thesis co-funded by the French National Association of Technical Research (ANRT), the car manufacturer PSA Group and the software developer Altair Engineering France.

References

[1] *Reference Manual for Intel® Math Kernel Library (Intel® MKL)*, 2019.

[2] ALTAIR Engineering, Troy. *Altair RADIOSS 2017 Reference Guide*, 2017.

[3] C Bach, D Ceglia, L Song, and F Duddeck. Randomized low-rank approximation methods for projection-based model order reduction of large nonlinear dynamical problems. *International Journal for Numerical Methods in Engineering*, 118(4):209–241, 2019.

[4] C Bach, F Duddeck, and L Song. Fixed-precision randomized low-rank approximation methods for nonlinear model order reduction of large systems. *International Journal for Numerical Methods in Engineering*.

[5] C Bach, L Song, T Erhart, and F Duddeck. Stability conditions for the explicit integration of projection based nonlinear reduced-order and hyper reduced structural mechanics finite element models. *arXiv preprint arXiv:1806.11404*, 2018.

[6] G. Berkooz, P. Holmes, and John L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25:539–575, 1993.

[7] M. Brand. Incremental singular value decomposition of uncertain data with missing values. *Springer LEC Notes Compute SC*, 2350, 2002.

[8] M. Brand. Fast low-rank modifications of the thin singular value decomposition. *Linear Algebra Appl*, 415:20–30, 2006.

[9] S. Chaturantabut. *Nonlinear Model reduction via Discrete Empirical Interpolation*. PhD thesis, Rice University, 2011.

[10] F. Chinesta, R. Keunings, and A. Leygue. *The Proper Generalized Decompostition for Advanced Numerical Simulations*. SPRINGER, 2014.

[11] Francisco Chinesta, Elias Cueto, Miroslav Grmela, Beatriz Moya, and Michal Pavelka. Learning physics from data: a thermodynamic interpretation. *arXiv preprint arXiv:1909.01074*, 2019.

[12] R.J. Dedden. Model order reduction using the discrete empirical interpolation method. Master's thesis, Delft University of Technology, 2012.

[13] Hiba Fareed and John R Singler. Error analysis of an incremental pod algorithm for pde simulation data. *arXiv preprint arXiv:1803.06313*, 2018.

[14] 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. *Int. J. Numer. Meth. Engng*, 2014.

[15] F Fritzen and D Ryckelynck. *Machine Learning, Low-Rank Approximations and Reduced Order Modeling in Computational Mechanics*. Mathematical and Computational Applications, 2019.

[16] Anthony Giacoma, David Dureisseix, and Anthony Gravouil. An efficient quasi-optimal space-time pgd application to frictional contact mechanics. *Advanced Modeling and Simulation in Engineering Sciences*, 3(1):12, 2016.

[17] G. H. Golub and C. F. Van Loan. *Matrix Computations Third Edition*. The Johns Hopkins University Press, 1996.

[18] J.A. Hernandez, M.A. Caicedo, and A. Ferrer. Dimensional hyper-reduction of nonlinear finite element models via empirical cubature. *Comput Meth Appl M*, 2016.

[19] Saddam Hijazi, Giovanni Stabile, Andrea Mola, and Gianluigi Rozza. Data-driven pod-galerkin reduced order model for turbulent flows. *arXiv preprint arXiv:1907.09909*, 2019.

[20] Philip Holmes, John L Lumley, Gahl Berkooz, and Clarence W Rowley. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge university press, 2012.

[21] M.A. Iwen and B.W. Ong. A distributed and incremental svd algorithm for agglomerative data analysis on large networks. *SIAM Journal on Matrix Analysis and Applications*, 37(4):1699–1718, 2016.

[22] DD. Kosambi. Statistics in function space. *Journal of Indian Mathematical Society*, 7:76–88, 1943.

[23] Pierre Ladevèze, Ch Paillet, and David Néron. Extended-pgd model reduction for nonlinear solid mechanics problems involving many parameters. In *Advances in Computational Plasticity*, pages 201–220. Springer, 2018.

[24] Ye Lu, N Blal, and A Gravouil. Space–time pod based computational vademecums for parametric studies: application to thermo-mechanical problems. *Advanced Modeling and Simulation in Engineering Sciences*, 5(1):3, 2018.

[25] Anna Madra, Piotr Breitkopf, Balaji Raghavan, and François Trochu. Diffuse manifold learning of the geometry of woven reinforcements in composites. *Comptes Rendus Mécanique*, 346(7):532–538, 2018.

[26] L Meng, P Breitkopf, Balaji Raghavan, G Mauvoisin, O Bartier, and X Hernot. On the study of mystical materials identified by indentation on power law and voce hardening solids. *International Journal of Material Forming*, pages 1–16, 2018.

[27] Liang Meng, Piotr Breitkopf, Guénhaël Le Quilliec, Balaji Raghavan, and Pierre Villon. Nonlinear shape-manifold learning approach: concepts, tools and applications. *Archives of Computational Methods in Engineering*, 25(1):1–21, 2018.

[28] G. M. Oxberry, T. Kostova-Vassilevska, B. Arrighi, and K. Chand. Limited-memory adaptive snapshot selection for proper orthogonal decomposition. Technical Report LLNL-TR-669265, Lawrence Livermore National Laboratory, 2015.

[29] P. Phalippou, S. Bouabdallah, P. Breitkopf, and P. Villon. 'on the fly' snapshot selection for hyper-reduced proper orthogonal decomposition with application to nonlinear dynamic. In 6^{th} *European Conference on Computational Mechanics (ECCM6)*, Glasgow, UK, June 2018.

[30] Balaji Raghavan and Piotr Breitkopf. Asynchronous evolutionary shape optimization based on high-quality surrogates: application to an air-conditioning duct. *Engineering with Computers*, 29(4):467–476, Oct 2013.

[31] Balaji Raghavan, Piotr Breitkopf, Yves Tourbier, and Pierre Villon. Towards a space reduction approach for efficient structural shape optimization. *Structural and Multidisciplinary Optimization*, 48(5):987–1000, 2013.

[32] Balaji Raghavan, Mohamed Hamdaoui, Manyu Xiao, Piotr Breitkopf, and Pierre Villon. A bi-level meta-modeling approach for structural optimization using modified pod bases and diffuse approximation. *Computers & Structures*, 127:19–28, 2013.

[33] Balaji Raghavan, Manyu Xiao, Piotr Breitkopf, and Pierre Villon. Implicit constraint handling for shape optimisation with pod-morphing. *European Journal of Computational Mechanics*, 21(3-6):325–336, 2012.

[34] D. Ryckelynck, F. Vincent, and S. Cantournet. Multidimensional a priori hyper-reduction of mechanical models involving internal variables. *Comput Meth Appl M*, 2012.

[35] Lawrence Sirovich. Turbulence and the dynamics of coherent structures. i. coherent structures. *Quarterly of applied mathematics*, 45(3):561–571, 1987.

[36] P. Tiso, R. Dedden, and D. Rixen. A modified discrete empirical interpolation method for reducing non-linear structural finite element models. In *Proceedings of the ASME Design Engineering Technical Conference*, 2013.

[37] Matthieu Vitse, David Néron, and P-A Boucard. Dealing with a nonlinear material behavior and its variability through pgd models: Application to reinforced concrete structures. *Finite Elements in Analysis and Design*, 153:22–37, 2019.

[38] Manyu Xiao, Piotr Breitkopf, Rajan Filomeno Coelho, Pierre Villon, and Weihong Zhang. Proper orthogonal decomposition with high number of linear constraints for aerodynamical shape optimization. *Applied mathematics and computation*, 247:1096–1112, 2014.