



HAL
open science

A Non-Intrusive Space-Time Interpolation from Compact Stiefel Manifolds of Parametrized Rigid-Viscoplastic FEM Problems

Orestis Friderikos, Marc Olive, Emmanuel Baranger, Dimitrios Sagris,
Constantine David

► To cite this version:

Orestis Friderikos, Marc Olive, Emmanuel Baranger, Dimitrios Sagris, Constantine David. A Non-Intrusive Space-Time Interpolation from Compact Stiefel Manifolds of Parametrized Rigid-Viscoplastic FEM Problems. Computational Mechanics, 2021, 10.1007/s00466-021-02050-0 . hal-03118209v3

HAL Id: hal-03118209

<https://hal.science/hal-03118209v3>

Submitted on 24 Oct 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.

1 **A Non-Intrusive Space-Time Interpolation from**
2 **Compact Stiefel Manifolds of Parametrized**
3 **Rigid-Viscoplastic FEM Problems**

4 **Orestis Friderikos¹, Marc Olive¹, Emmanuel Baranger¹, Dimitris Sagris²,**
5 **Constantine David²**

6 ¹ Université Paris-Saclay, ENS Paris-Saclay, CNRS, LMT - Laboratoire de Mécanique et Tech-
7 nologie, 91190, Gif-sur-Yvette, France

8 ² Mechanical Engineering Department, Laboratory of Manufacturing Technology & Machine
9 Tools, International Hellenic University, GR-62124 Serres Campus, Greece.

Received: date / Revised version: date

10 **Contents**

11 **1 Introduction** 4
12 **2 Space-Time POD, Grassmann and compact Stiefel manifolds** 10
13 **2.1 Riemannian geometry on Grassmann manifolds** 13
14 **2.2 Target Algorithm on compact Stiefel manifolds** 19
15 **3 Space-Time Interpolation on compact Stiefel manifolds** 23
16 **3.1 Oriented SVD on generic matrices** 25

Send offprint requests to: Marc Olive
Correspondence to: marc.olive@math.cnrs.fr

17	3.2 Space-Time interpolation algorithm	29
18	4 Rigid-Viscoplastic FEM Formulation	33
19	4.1 Governing Field Equations	34
20	4.2 Variational form	35
21	4.3 Discretization and iteration	36
22	4.4 Heat Transfer Analysis	38
23	4.5 Computational Procedure for Thermo-Mechanical Analysis	39
24	5 Numerical Investigations	40
25	5.1 Mechanical field	42
26	5.2 Temperature field	45
27	5.3 Computational complexity	46
28	6 Conclusions	47

29 **Abstract** This work aims to interpolate parametrized Reduced Order Model (ROM)
30 basis constructed via the Proper Orthogonal Decomposition (POD) to derive a robust
31 ROM of the system's dynamics for an unseen target parameter value. A novel non-intrusive
32 Space-Time (ST) POD basis interpolation scheme is proposed, for which we define ROM
33 spatial and temporal basis *curves on compact Stiefel manifolds*. An interpolation is finally
34 defined on a *mixed part* encoded in a square matrix directly deduced using the space part,
35 the singular values and the temporal part, to obtain an interpolated snapshot matrix,
36 keeping track of accurate space and temporal eigenvectors. Moreover, in order to establish
37 a well-defined curve on the compact Stiefel manifold, we introduce a new procedure, the so-
38 called oriented SVD. Such an oriented SVD produces unique right and left eigenvectors for
39 generic matrices, for which all singular values are distinct. It is important to notice that the
40 ST POD basis interpolation does not require the construction and the subsequent solution

41 of a reduced-order FEM model as classically is done. Hence it is avoiding the bottleneck
42 of standard POD interpolation which is associated with the evaluation of the nonlinear
43 terms of the Galerkin projection on the governing equations. As a proof of concept, the
44 proposed method is demonstrated with the adaptation of rigid-thermoviscoplastic finite
45 element ROMs applied to a typical nonlinear open forging metal forming process. Strong
46 correlations of the ST POD models with respect to their associated high-fidelity FEM
47 counterpart simulations are reported, highlighting its potential use for near real-time
48 parametric simulations using off-line computed ROM POD databases.

Notations

$\text{Mat}_{n,p}(\mathbb{R})$	Set of $n \times p$ matrices in \mathbb{R}	
\mathbf{I}_p	Identity matrix in $\text{Mat}_{p,p}(\mathbb{R})$	
$[\mathbf{y}_1, \dots, \mathbf{y}_p]$	Matrix in $\text{Mat}_{n,p}(\mathbb{R})$	Matrix with column vectors $\mathbf{y}_i \in \mathbb{R}^n$
$O(p)$	Orthogonal group on \mathbb{R}^p	$\{Q \in \text{Mat}_{p,p}(\mathbb{R}), Q^T Q = \mathbf{I}_p\}$
$\mathcal{G}(p, n)$	Grassmann manifold	Set of p linear subspaces in \mathbb{R}^n
$\pi^{-1}(\mathbf{m})$	Fiber at $\mathbf{m} \in \mathcal{G}(p, n)$	If $\mathbf{y}_1, \dots, \mathbf{y}_p$ is an orthonormal basis of \mathbf{m} $\pi^{-1}(\mathbf{m}) = \{\mathbf{Y}Q, Q \in O(p), \mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_p]\}$
$\mathcal{T}_{\mathbf{m}} := \mathcal{T}_{\mathbf{m}}\mathcal{G}(p, n)$	Tangent space of $\mathcal{G}(p, n)$ at \mathbf{m}	For $\mathbf{Y} \in \pi^{-1}(\mathbf{m})$, one model of $\mathcal{T}_{\mathbf{m}}$ is $\{\mathbf{Z} \in \text{Mat}_{n,p}(\mathbb{R}), \mathbf{Z}^T \mathbf{Y} = \mathbf{0}\}$
$St(p, n)$	Stiefel manifold	Set of ordered p -tuples independent vectors in \mathbb{R}^n
$St_c(p, n)$	Compact Stiefel manifold	Set of ordered p -tuples of orthonormal vectors in \mathbb{R}^n $St_c(p, n) = \{\mathbf{Y} \in \text{Mat}_{n,p}(\mathbb{R}), \mathbf{Y}^T \mathbf{Y} = \mathbf{I}_p\}$
$\text{Hor}_{\mathbf{Y}}$	Horizontal space at \mathbf{Y}	$\text{Hor}_{\mathbf{Y}} := \{\mathbf{Z} \in \text{Mat}_{n,p}(\mathbb{R}), \mathbf{Z}^T \mathbf{Y} = \mathbf{0}\}$
$v \in \mathcal{T}_{\mathbf{m}}$	Velocity vector on the tangent plane $\mathcal{T}_{\mathbf{m}}$	Represented by a horizontal lift $\mathbf{Z} \in \text{Hor}_{\mathbf{Y}}$, with $\mathbf{Y} \in \pi^{-1}(\mathbf{m})$
$\mathbf{S}^{(i)}$	Snapshot matrix	$\mathbf{S}^{(i)} \in \text{Mat}_{n,m}(\mathbb{R})$ corresponding to parameter value λ_i

49 **1 Introduction**

50 Computational metal forming has been widely used in academic laboratories and the man-
51 ufacturing industry over the last decades, becoming nowadays a mature, well established

52 technology. Nevertheless, new challenging fields are emerging, among others, uncertainty
53 quantification, optimization of processes and parameter identification in design analysis [1,
54 2]. One of the key challenging topics mentioned in [1] is the introduction of Model Or-
55 der Reduction (MOR) methods to combat the high computational cost, which is also of
56 paramount interest in the above-mentioned fields. Moreover, due to the multiple sources of
57 strong non-linearities inherent in manufacturing problems, design optimization and multi-
58 parametric studies of large scale models turns out to be prohibitively expensive. Indeed,
59 simulation of complex configurations can be intractable since the computational times can
60 highly increase.

61 To this end, meta-model techniques are often used to tackle the computational burden.
62 These rely on a manifold learning stage during which we need to capture the original space
63 where the solution of the model problem lies. This data collection consists of solving
64 the full-scale model for an ensemble of training data over the parametric range and is
65 commonly referred to as the offline stage. Even though meta-models can speed up the
66 simulation time, nevertheless their construction with standard computations based on
67 full-order models is expensive.

68 Closely related to the concept of metamodeling, Reduced Order Models (ROMs) have
69 been chosen to reduce the problem's dimensionality while at the same time maintaining
70 solution accuracy. ROMs can decrease the computational complexity of large-scale sys-
71 tems, solving parametrized problems and offering the potential for near real-time analysis.
72 The methods for building ROMs can be classified into two general families: *a priori* and
73 *a posteriori* ones. The well known *a priori* MOR includes methods such as the Proper
74 Generalized Decomposition (PGD) [3], and the *a priori* reduction method (APR) [3,4].

75 The main characteristic of all these methods is that they do not require any precomputed
76 ROMs. In the second class of methods, the reduced basis is built, *a posteriori*, from the
77 state variable snapshots in the parametric space. One popular method is the POD [5,6,
78 7], also known as Karhunen-Loève Decomposition (KLD) [8,9], Singular Value Decom-
79 position (SVD) [10] or Principal Component Analysis (PCA) [11,12,13,14].

80 For nonlinear systems, even though a Galerkin projection reduces the number of un-
81 knowns, however, the computational burden for obtaining the solution could still be high
82 due to the computational costs involved in the evaluation of nonlinear terms. Hence, the
83 nonlinear Galerkin projection principally leads to a ROM, but its evaluation could be more
84 expensive than the corresponding one of the original problem. To this effect, to make the
85 resulting ROMs computationally efficient, a sparse sampling method is used, also called
86 hyper reduction, to mention among others, the missing point estimation (MPE) [15], the
87 empirical interpolation method (EIM) [16], the discrete empirical interpolation method
88 (DEIM) [17], the Gappy POD method [18], and the Gauss-Newton with approximated
89 tensors (GNAT) method [19]. Thus, all these methods imply the solution of a new ROM
90 FEM problem.

91 In the case of a parametric analysis using POD basis interpolation on Grassmann man-
92 ifolds [20,21], the method starts with a training stage during which the problem is solved
93 for several training points. Then, using the FEM solutions, the full-order field ‘snapshots’
94 are compressed using the POD to generate a ROM that is expected to reproduce the
95 most characteristic dynamics of its high-fidelity counterpart solution. However, the rele-
96 vant information is contained in the vector spaces generated by the (left or right) singular
97 vectors of the snapshot matrices. Now, for a new parameter value, interpolation methods

98 have to be defined from such relevant *subspaces* spanned by the POD basis vectors [20].
99 Other approaches obviously could be considered, such as interpolations computed on the
100 space of matrices of a fixed rank, whereby the mechanical origin of the problem imposes
101 to consider the vector subspaces, and not the matrices themselves [21]. Nevertheless, such
102 methods as the one of interpolation between two positive semidefinite matrices of fixed
103 rank [22], can be used in the special case of interpolation between two precomputed ROM
104 basis and a single parameter which is not of general interest in mechanical applications.

105 To interpolate between different vector spaces of the same dimension (encoded into
106 the mode p of the POD), a Grassmann manifold [23] must be used, which is the set of
107 p -dimensional subspaces of \mathbb{R}^n . Such a manifold is in fact a Riemannian manifold [24],
108 so we can construct *geodesics* between two points, and use such geodesics to define a
109 logarithm map to *linearize*, and conversely using the exponential map to return back to
110 the Grassmann manifold. While an interpolation cannot be done directly on Grassmann
111 manifolds, linearization allows computing such an interpolation, at least locally once a
112 reference point has been selected [20,21]. To any new parameter value, thus we get a new
113 subspace obtained from interpolation between all subspaces related to the spatial eigen-
114 vectors of the snapshot matrices. Another approach using inverse distance weighting was
115 initiated in [21,25], but it also relies on several choices (as one of the weights). Further-
116 more, an extension of Neville-Aitken's algorithm to Grassmann manifolds which computes
117 the Lagrange interpolation polynomial in a recursive way from the interpolation of two
118 points was recently presented [26].

119 In the standard POD interpolation mentioned above [20], the spatial ROM basis cor-
120 responding to the target point is used to generate a ROM FEM, which is expected to

121 have a lower computational cost compared to the high-fidelity problem. The key idea in
 122 the *Space-Time* (ST) POD basis interpolation proposed by [27,28], is that the reduced
 123 spatial and temporal basis are considered separately, both defining points on two dif-
 124 ferent Grassmann manifolds. However, such points are strongly related: a spatial vector
 125 directly corresponds to a temporal vector, and *vice versa*. From this, firstly we need to
 126 consider the p -tuples of spatial (and temporal) vectors, instead of the p -dimensional sub-
 127 space, which defines points on an associated *compact Stiefel manifold*, strongly connected
 128 to Grassmann manifolds. Contrary to what is suggested in [28], we propose a different
 129 interpolation scheme, as we do not perform interpolation of the singular values, followed
 130 by spatial and temporal calibration. Instead, we exploit the dependence between the spa-
 131 tial and temporal parts. Indeed, using an interpolation algorithm defined on a Grassmann
 132 manifold, we derive *curves on a compact Stiefel manifold*, which are no longer interpolat-
 133 ing, but which nevertheless allow us to obtain new singular vectors for the spatial part,
 134 and separately for the temporal part. Such space and temporal singular vectors finally are
 135 taken to define a *mixed part* on which a classical interpolation can be computed. In the
 136 end, we get in this way a ROM matrix corresponding to a new parameter value. Note that
 137 in order to obtain a well-defined curve on compact Stiefel manifolds, we have to introduce
 138 a new procedure, the so-called *oriented SVD*. Such an oriented SVD produces unique right
 139 and left eigenvectors for snapshot matrices, supposed to be *generic matrices*, for which all
 140 non-zero singular values are distinct.

141 The off-line stage in the ST approach consists of solving FEM problems which are
 142 corresponding to the training points of the given parameter. The on-line stage concerns
 143 the use of a curve defined on a compact Stiefel manifold to determine the spatial and

144 temporal ROM basis for the target point, in order to construct the related ROM snapshot
145 matrix. In fact, the ST interpolation offers the advantage of reconstructing a snapshot
146 matrix without relaunching ROM FEM computations. To this end, it results in near-real-
147 time solutions due to direct matrix multiplications in the on-line stage.

148 We could also mention some other ST approaches [29,30,31,32], where neither Grass-
149 mann nor compact Stiefel manifolds are considered. For instance, an approximation of the
150 spatial and temporal basis functions by linear interpolation of their modes is proposed
151 in [29] to study the flow past a cylinder at low Reynolds numbers. A non-intrusive ROM
152 approach for nonlinear parametrized time-dependent PDEs based on a two-level POD
153 method by using Radial Basis Functions interpolation is presented in [30,33].

154 The method proposed in this work is applied to a coupled thermomechanical rigid
155 visco-plastic (RVP) FEM analysis based on an incremental implicit approach [34,35,36,
156 37]. Note that the RVP formulation specifically is tailored for metal forming simulations,
157 where the plastic flow is unconstrained and usually of finite magnitude, involving large
158 strain-rates and high temperatures. In the present study, all simulations are performed
159 by using an in-house Matlab code which consists of two independent FEM solvers. A
160 mechanical solver for the viscoplastic deformation analysis [38] and a thermal solver for
161 the heat transfer analysis. A staggered procedure is used to solve the system of coupled
162 equations.

163 The paper is organized as follows: in [section 2](#), the Proper Orthogonal Decomposition
164 is presented, followed by an introduction to some basic notions about the geometry of
165 the Grassmann and Stiefel manifolds to make the article reasonable self-contained. POD
166 basis interpolation on Grassmannian manifolds is introduced considering the underlying

167 formulation of the logarithm and the exponential map. The core of this paper is illustrated
 168 in [section 3](#), where the computational framework for the ROM adaptation based on a
 169 novel non-intrusive Space-Time POD basis interpolation on compact Stiefel manifolds is
 170 developed. The following [section 4](#) covers the rigid visco-plastic formulation, the general
 171 framework of the thermal field equations, and the thermomechanical coupling. In [section 5](#),
 172 the interpolation performance applied to a metal forming process is shown, as well as
 173 further computational aspects are discussed. Finally, [section 6](#) highlights the main results
 174 and some important outcomes.

175 **2 Space–Time POD, Grassmann and compact Stiefel manifolds**

176 Let us recall here the important link between Proper Orthogonal Decomposition and
 177 Grassmann manifold [\[20,39,40,21,25\]](#).

178 Assume $\mathbf{S} \in \text{Mat}_{n,m}(\mathbb{R})$ to be any real matrix of size $n \times m$ (with $n \geq m$), taken here
 179 to be a snapshot matrix with $n = 3N_S$ obtained from the spatial discretization N_s , and
 180 $m = N_t$ obtained from the time one. Any spatial POD of mode p leads to a p -dimensional
 181 vector space $\mathcal{V}_p \subset \mathbb{R}^m$ such that the Frobenius norm

$$182 \quad \|\mathbf{S} - \mathbf{I}_p \mathbf{S}\|_F^2$$

183 is minimal, where matrix \mathbf{I}_p corresponds to the orthogonal projection on \mathcal{V}_p (see [\[21\]](#) for
 184 more details). Such a matrix \mathbf{I}_p is directly obtained from a Singular Value Decomposition
 185 (SVD) of \mathbf{S} . Indeed, let us write a SVD

$$186 \quad \mathbf{S} = \boldsymbol{\Phi} \boldsymbol{\Sigma} \boldsymbol{\Psi}^T$$

187 with $\Phi = [\phi_1, \dots, \phi_r]$ and $\Psi = [\psi_1, \dots, \psi_r]$, where the columns $\phi_k \in \mathbb{R}^n$ and $\psi_k \in \mathbb{R}^m$
 188 form a set of orthonormal vectors, and $\Sigma \in \text{Mat}_{r,r}(\mathbb{R})$ is a diagonal matrix, where r
 189 denotes the rank of \mathbf{S} . Then, we can define $\Phi_p := [\phi_1, \dots, \phi_p] \in \text{Mat}_{n,p}(\mathbb{R})$ and we obtain
 190 $\Pi_p = \Phi_p \Phi_p^T$.

191 In this classical approach, the relevant object is not the reduced matrix $\mathbf{S}_p := \Pi_p \mathbf{S}$,
 192 supposed to be of maximal rank, but the p -dimensional vector space \mathcal{V}_p spanned by vectors
 193 ϕ_1, \dots, ϕ_p , and thus the image of the matrix Φ_p . From this, interpolation has to be con-
 194 sidered on the set of all p -dimensional vector spaces, that is on the so-called *Grassmann*
 195 *manifold* $\mathcal{G}(p, n)$:

$$196 \quad \mathcal{G}(p, n) := \{\mathcal{V}_p \subset \mathbb{R}^n, \quad \dim(\mathcal{V}_p) = p\}.$$

197 Note here that the *point* $\mathbf{m} := \mathcal{V}_p \in \mathcal{G}(p, n)$ defines a vector space spanned by the
 198 set ϕ_1, \dots, ϕ_p represented by matrix Φ_p , however this matrix representation is not unique
 199 (see Example 1).

200 Take now a set $\{\lambda_1, \dots, \lambda_N\}$ of parameter values leading to snapshot matrices $\mathbf{S}^{(1)}, \dots, \mathbf{S}^{(N)}$
 201 with SVD

$$202 \quad \mathbf{S}^{(k)} = \Phi^{(k)} \Sigma^{(k)} \Psi^{(k)}, \quad \Phi^{(k)} = [\phi_1^{(k)}, \dots, \phi_r^{(k)}], \quad \Psi^{(k)} = [\psi_1^{(k)}, \dots, \psi_r^{(k)}],$$

203 where $\phi_i^{(k)}$ are orthonormal vectors in \mathbb{R}^n and $\psi_j^{(k)}$ are orthonormal vectors in \mathbb{R}^m .

204 The classical approach [20,21] then considers the spatial POD of the snapshot ma-
 205 trices $\mathbf{S}_p^{(1)}, \dots, \mathbf{S}_p^{(N)}$ of mode p , so that we obtain points \mathbf{m}_i ($i = 1, \dots, N$) on $\mathcal{G}(p, n)$,
 206 respectively represented by the matrices

$$207 \quad \Phi_p^{(k)} := [\phi_1^{(k)}, \dots, \phi_p^{(k)}] \in \text{Mat}_{n,p}(\mathbb{R}), \quad \left(\Phi_p^{(k)}\right)^T \Phi_p^{(k)} = \mathbf{I}_p.$$

208 To any new parameter value $\tilde{\lambda}$, it is possible to make an interpolation considering
 209 the spatial part based on the points $\mathbf{m}_i \in \mathcal{G}(p, n)$, using a local chart given by normal
 210 coordinates [20, 21, 25], in order to obtain a point $\tilde{\mathbf{m}} \in \mathcal{G}(p, n)$ represented by a matrix $\tilde{\Phi}$.
 211 From such a point $\tilde{\mathbf{m}} \in \mathcal{G}(p, n)$, we deduce a p -dimensional vector space on which some
 212 POD-Galerkin approach [21] can lead to a new ROM model.

213 On the contrary, we propose another approach as we consider a *Space-Time interpola-*
 214 *tion*, using both the spatial vector spaces represented by matrices $\Phi_p^{(k)}$ and the temporal
 215 vector spaces represented by matrices

$$216 \quad \Psi_p^{(k)} := [\psi_1^{(k)}, \dots, \psi_p^{(k)}] \in \text{Mat}_{m,p}(\mathbb{R}), \quad \left(\Psi_p^{(k)}\right)^T \Psi_p^{(k)} = \mathbf{I}_p.$$

217 An important observation now is that matrices $\Phi_p^{(k)}$ (resp. $\Psi_p^{(k)}$) directly define an *ordered*
 218 *p -tuple of orthonormal vectors* in \mathbb{R}^n (resp. \mathbb{R}^m), that is a point on the *compact Stiefel*
 219 *manifold*

$$220 \quad \mathcal{St}_c(p, n) := \{\text{Ordered orthonormal } p\text{-tuple of vectors in } \mathbb{R}^n\}.$$

221 To obtain a Space-Time POD interpolation (instead of a spatial POD interpolation fol-
 222 lowed by Galerkin approach), we finally adopted the following strategy, when dealing with
 223 a parameter value $\tilde{\lambda}$:

- 224 1. Define a *curve* on the compact Stiefel manifold corresponding to the spatial part

$$225 \quad \lambda \mapsto \Phi(\lambda) \in \mathcal{St}_c(p, n)$$

226 obtained using the already known interpolation algorithm on Grassmann manifold.

- 227 2. In the same way, define a *curve* on the compact Stiefel manifold corresponding to the
 228 temporal part

$$229 \quad \lambda \mapsto \Psi(\lambda) \in \mathcal{St}_c(p, m).$$

230 3. Construct an *interpolated curve* $\lambda \mapsto \mathbf{S}(\lambda)$ passing through the POD of mode p snap-
 231 shot matrices $\mathbf{S}_p^{(k)}$, in order to obtain an interpolation of a ROM matrix $\tilde{\mathbf{S}} := \mathbf{S}(\tilde{\lambda})$.

232 In the next subsections, we give all important details to obtain such an interpolated curve
 233 $\lambda \mapsto \mathbf{S}(\lambda)$. First, in [subsection 2.1](#) we explain how to compute on Grassmann manifolds
 234 using their Riemannian structure to obtain explicit formulae for the *geodesics* defining
 235 *normal coordinates*. From this explicit formulae, we can deduce in [subsection 2.2](#) a *target*
 236 *algorithm* in order to define the curves

$$237 \quad \lambda \mapsto \Phi(\lambda) \in \mathcal{St}_c(p, n), \quad \lambda \mapsto \Psi(\lambda) \in \mathcal{St}_c(p, m)$$

238 on compact Stiefel manifolds. The question on how to define an interpolated curve for
 239 matrices $\mathbf{S}_p^{(k)}$ will then be addressed in [section 3](#).

240 2.1 Riemannian geometry on Grassmann manifolds

241 We will summarize now some essential results about Grassmann manifolds. Such manifolds
 242 are in fact *complete Riemannian manifolds* [\[24\]](#), meaning for instance that we can define
 243 the *length* of a curve. Moreover, we can always construct a curve of the shortest length
 244 between two points, which is called a *geodesic*, and it will be the starting point to define
 245 *normal coordinates* via the exponential and logarithm map ([Definition 1](#) and [2](#)). As we
 246 cannot do direct computations on Riemann manifolds, normal coordinates enable us to
 247 obtain formulae of curves, such as the Lagrangian polynomials. Note finally that a rigorous
 248 mathematical background of all of this is given in [\[41\]](#).

249 After we give a definition of the Grassmann manifold and how to *represent* its points
 250 with matrices, we propose to define the *tangent plane* using matrix representative, to have

251 formulae for a scalar product, given by (3). From this, we deduce a classical expression
 252 for geodesics (Theorem 1).

253 Let $p \leq n$ be two non-zero integers and $\mathcal{G}(p, n)$ the Grassmann manifold of p -dimensional
 254 subspaces in \mathbb{R}^n . In fact, Grassmann manifolds are special cases of *quotient manifolds*,
 255 meaning that a point on such a manifold can have many *representatives*. Let us consider
 256 indeed a p -dimensional linear subspace \mathcal{V} of \mathbb{R}^n . Such a subspace can be defined using any
 257 ordered set of p independent vectors $\mathbf{v}_1, \dots, \mathbf{v}_p$ in \mathbb{R}^n , encoded into a full rank matrix

$$258 \quad \mathbf{M} := [\mathbf{v}_1, \dots, \mathbf{v}_p] \in \text{Mat}_{n,p}(\mathbb{R}).$$

259 Any other basis $\mathbf{v}'_1, \dots, \mathbf{v}'_p$ of \mathcal{V} will then lead to another full rank matrix

$$260 \quad \mathbf{M}' := [\mathbf{v}'_1, \dots, \mathbf{v}'_p] \in \text{Mat}_{n,p}(\mathbb{R}),$$

261 and we necessary have

$$262 \quad \mathbf{M}' = \mathbf{M}\mathbf{P}$$

263 where $\mathbf{P} \in \text{GL}(p)$ is some invertible matrix in $\text{Mat}_{p,p}(\mathbb{R})$. From all this, we deduce that
 264 the point $\mathbf{m} := \mathcal{V} \in \mathcal{G}(p, n)$ is represented by the infinite set of matrices

$$265 \quad \{\mathbf{M}\mathbf{P}, \quad \mathbf{P} \in \text{GL}(p)\}.$$

266 Now, the ordered set of p independent vectors in \mathbb{R}^n and thus the set of full rank
 267 matrices in $\text{Mat}_{n,p}(\mathbb{R})$ define the Stiefel manifold (see Figure 1)

$$268 \quad \mathcal{S}t(p, n) := \{\mathbf{M} = [\mathbf{v}_1, \dots, \mathbf{v}_p] \in \text{Mat}_{n,p}(\mathbb{R}), \quad \text{rg}(\mathbf{M}) = p\}$$

269 so that we obtain a natural map from such Stiefel manifold and the Grassmann manifold
 270 $\mathcal{G}(p, n)$ (see Figure 2):

$$271 \quad \mathbf{M} = [\mathbf{v}_1, \dots, \mathbf{v}_p] \in \mathcal{S}t(p, n) \mapsto \mathbf{m} = \{\mathbf{M}\mathbf{P}, \quad \mathbf{P} \in \text{GL}(p)\}.$$

272 In our situation, nevertheless, we will only focus on *orthonormal bases* of p -dimensional
 273 subspaces. Doing so, we thus consider matrices defined by orthonormal vectors, leading
 274 to the so-called *compact Stiefel manifold*

$$275 \quad St_c(p, n) := \{ \mathbf{Y} \in \text{Mat}_{n,p}(\mathbb{R}), \quad \mathbf{Y}^T \mathbf{Y} = \mathbf{I}_p \} \quad (1)$$

276 and any point $\mathbf{m} \in \mathcal{G}(p, n)$ will then be represented by the infinite set

$$277 \quad \{ \mathbf{Y} \mathbf{Q}, \quad \mathbf{Q} \in \text{O}(p) \}$$

278 where $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_p]$ is defined using an orthonormal basis $\mathbf{y}_1, \dots, \mathbf{y}_p$ of \mathbf{m} . This defines
 279 a *surjective* map

$$280 \quad \pi : \mathbf{Y} \in St_c(p, n) \mapsto \mathbf{m} = \pi(\mathbf{Y}) = \{ \mathbf{Y} \mathbf{Q}, \quad \mathbf{Q} \in \text{O}(p) \} \in \mathcal{G}(p, n)$$

281 and the set of all matrices representing the same point $\mathbf{m} \in \mathcal{G}(p, n)$ is called the *fiber* of
 282 π at \mathbf{m} (see Figure 3 for an illustration of a fiber):

$$283 \quad \pi^{-1}(\mathbf{m}) = \{ \mathbf{Y} \mathbf{Q}, \quad \mathbf{Q} \in \text{O}(p) \}.$$

284 *Remark 1* An important point here is that, from now on, any computation on $\mathcal{G}(p, n)$
 285 will be done using a *choice in the fibers*. Nevertheless, for any point $\mathbf{m} \in \mathcal{G}(p, n)$, *there*
 286 *is no canonical* way to choose an element $\mathbf{Y} \in \pi^{-1}(\mathbf{m})$, so any computation has to be
 287 independent of that choice.

288 We need now to define the *geodesics* of Grassmann manifold, which can be done once we
 289 have defined the *tangent plane* at each point $\mathbf{m} \in \mathcal{G}(p, n)$ and a Riemannian metric. Take
 290 any point $\mathbf{m} \in \mathcal{G}(p, n)$ represented by a matrix $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_p]$ of orthonormal vectors,

291 the tangent plane $\mathcal{T}_{\mathbf{m}} := \mathcal{T}_{\mathbf{m}}\mathcal{G}(p, n)$ is then represented by the $p(n-p)$ dimensional vector
 292 space

$$293 \quad \text{Hor}_{\mathbf{Y}} := \{ \mathbf{Z} \in \text{Mat}_{n,p}(\mathbb{R}), \quad \mathbf{Z}^T \mathbf{Y} = 0 \}, \quad (2)$$

294 called the *horizontal space*, where $\mathbf{Y} \in \pi^{-1}(\mathbf{m})$. From all this, a vector $v \in \mathcal{T}_{\mathbf{m}}$ will be
 295 called a *velocity vector*, which can be *represented* by a matrix $\mathbf{Z} \in \text{Mat}_{n,p}(\mathbb{R})$ such that
 296 $\mathbf{Z}^T \mathbf{Y} = 0$, and \mathbf{Z} is called a *horizontal lift* of v .

297 *Example 1* Take here $p = 2$ and $n = 5$, so that $\mathcal{G}(2, 5)$ is the set of planes in a five
 298 dimensional space. The matrices

$$299 \quad \mathbf{Y} = \begin{bmatrix} \frac{1}{2} & 0 \\ -\frac{1}{2} & \frac{\sqrt{2}}{2} \\ 0 & 0 \\ \frac{1}{2} & \frac{\sqrt{2}}{2} \\ \frac{1}{2} & 0 \end{bmatrix}, \quad \mathbf{Y}' = \begin{bmatrix} \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} \\ \frac{2-\sqrt{2}}{4} & \frac{2+\sqrt{2}}{4} \\ 0 & 0 \\ \frac{2+\sqrt{2}}{4} & \frac{2-\sqrt{2}}{4} \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} \end{bmatrix}$$

300 are in the compact Stiefel manifold $St_c(2, 5)$, representing the same plane $\mathbf{m} \in \mathcal{G}(2, 5)$.
 301 The horizontal space $\text{Hor}_{\mathbf{Y}}$ defined by (2) is a 6-dimensional vector space of matrices \mathbf{Z} ,
 302 for instance given by

$$303 \quad \mathbf{Z} = \begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \\ u_3 & v_3 \\ -u_2 & -v_2 \\ -u_1 + u_2 - u_4 & -v_1 + v_2 - v_4 \end{bmatrix}, \quad u_i, v_i \in \mathbb{R}.$$

304 Taking now velocity vectors $v_1, v_2 \in \mathcal{T}_{\mathbf{m}}$ with respective horizontal lifts $\mathbf{Z}_1, \mathbf{Z}_2 \in \text{Hor}_{\mathbf{Y}}$
 305 we define the point-wise scalar product [42, 40]:

$$306 \quad \langle v_1, v_2 \rangle_{\mathbf{m}} := \text{tr} (\mathbf{Z}_1^T \mathbf{Z}_2). \quad (3)$$

307 Such a Riemannian metric leads to explicit geodesics given by [40, 39]:

308 **Theorem 1** Let $\mathbf{m} \in \mathcal{G}(p, n)$ represented by $\mathbf{Y} \in \text{St}_c(p, n)$. For any $v \in \mathcal{T}_{\mathbf{m}}$ with horizon-
 309 tal lift given by \mathbf{Z} in $\text{Hor}_{\mathbf{Y}}$, let $\mathbf{Z} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ be a thin SVD of \mathbf{Z} . Then

$$310 \quad \alpha_v : t \in \mathbb{R} \mapsto \alpha_v(t) := \pi \left[(\mathbf{Y}\mathbf{V} \cos(t\mathbf{\Sigma}) + \mathbf{U} \sin(t\mathbf{\Sigma})) \mathbf{V}^T \right] \in \mathcal{G}(p, n) \quad (4)$$

311 is the unique maximal geodesic such that $\alpha_v(0) = \mathbf{m}$ and initial velocity

$$312 \quad \dot{\alpha}_v(0) := \left. \frac{\partial \alpha_v(t)}{\partial t} \right|_{t=0} = v.$$

313 *Remark 2* Up to our knowledge, there is no proof that

$$314 \quad \mathbf{Y}(t) := (\mathbf{Y}\mathbf{V} \cos(t\mathbf{\Sigma}) + \mathbf{U} \sin(t\mathbf{\Sigma})) \mathbf{V}^T \in \text{St}_c(p, n). \quad (5)$$

315 In fact, this follows by direct computation. Indeed, $\mathbf{Z} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ being a thin SVD, we
 316 have $\mathbf{V} \in O(p)$ and

$$317 \quad \mathbf{Z}^T \mathbf{Y} = \mathbf{V}\mathbf{\Sigma}\mathbf{U}^T \mathbf{Y} = 0 \implies \mathbf{\Sigma}\mathbf{U}^T \mathbf{Y} = 0$$

318 so that

$$319 \quad \sin(t\mathbf{\Sigma})\mathbf{U}^T \mathbf{Y} = 0 \text{ and } \mathbf{Y}^T \mathbf{U} \sin(t\mathbf{\Sigma}) = 0.$$

Finally, we have:

$$\mathbf{Y}^T(t)\mathbf{Y}(t) = \mathbf{V} \left(\cos^2(t\mathbf{\Sigma}) + \underbrace{\sin(t\mathbf{\Sigma})\mathbf{U}^T \mathbf{Y} \mathbf{V} \cos(t\mathbf{\Sigma})}_{=0} + \right. \\ \left. \cos(t\mathbf{\Sigma}) \underbrace{\mathbf{V}^T \mathbf{Y}^T \mathbf{U} \sin(t\mathbf{\Sigma})}_{=0} + \sin^2(t\mathbf{\Sigma}) \right) \mathbf{V}^T$$

320 which concludes the proof.

321 *Remark 3* In many cases, formulas of the geodesic do not use the right multiplication by
 322 \mathbf{V}^T , as for instance in [40, 21]. Of course, as \mathbf{V} being in $O(p)$ both matrices

$$323 \quad (\mathbf{Y}\mathbf{V} \cos(t\boldsymbol{\Sigma}) + \mathbf{U} \sin(t\boldsymbol{\Sigma})) \mathbf{V}^T \text{ and } \mathbf{Y}\mathbf{V} \cos(t\boldsymbol{\Sigma}) + \mathbf{U} \sin(t\boldsymbol{\Sigma})$$

324 define the same point on $\mathcal{G}(p, n)$. Now, the choice of such right multiplication in (5) is
 325 related to the choice of the horizontal lift $\mathbf{Z} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$. Indeed, taking back the path given
 326 by (5), we have

$$327 \quad \dot{\mathbf{Y}}(t) = (-\mathbf{Y}\mathbf{V}\boldsymbol{\Sigma} \sin(t\boldsymbol{\Sigma}) + \mathbf{U}\boldsymbol{\Sigma} \cos(t\boldsymbol{\Sigma})) \mathbf{V}^T \implies \dot{\mathbf{Y}}(0) = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T = \mathbf{Z}$$

328 which corresponds to the choice of the horizontal lift for velocity vector $v \in \mathcal{T}_{\mathbf{m}}$.

329 A consequence of Theorem 1 is an explicit formula for the exponential map [40, 21] (see
 330 Figure 4):

Definition 1 Let $\mathbf{m} \in \mathcal{G}(p, n)$ be represented by $\mathbf{Y} \in St_c(p, n)$. For any velocity vector
 $v \in \mathcal{T}_{\mathbf{m}}$ with horizontal lift $\mathbf{Z} \in \text{Hor}_{\mathbf{Y}}$, take $\mathbf{Z} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ to be a thin SVD of \mathbf{Z} . Then we
 define the exponential map

$$\text{Exp}_{\mathbf{m}} : \mathcal{T}_{\mathbf{m}} \longrightarrow \mathcal{G}(p, n),$$

$$v \mapsto \text{Exp}_{\mathbf{m}}(v) := \pi [(\mathbf{Y}\mathbf{V} \cos(\boldsymbol{\Sigma}) + \mathbf{U} \sin(\boldsymbol{\Sigma})) \mathbf{V}^T] = \alpha_v(1).$$

331 Now, it is possible to define directly some inverse map of the exponential map, called
 332 the logarithm map [40], but only *locally*. For any \mathbf{m} and \mathbf{Y} in its fiber, let us first define
 333 the open space

$$334 \quad \mathbf{U}_{\mathbf{m}} := \{\mathbf{m}_1 \in \mathcal{G}(p, n), \quad \mathbf{Y}^T \mathbf{Y}_1 \text{ is invertible, } \mathbf{Y}_1 \in \pi^{-1}(\mathbf{m}_1)\}. \quad (6)$$

335 Then we have:

336 **Definition 2 (Logarithm map on Grassmannian manifold)** Let $\mathbf{m} \in \mathcal{G}(p, n)$ be
 337 represented by a matrix $\mathbf{Y} \in St_c(p, n)$. For any point \mathbf{m}_1 in the open space $U_{\mathbf{m}}$ represented
 338 by a matrix $\mathbf{Y}_1 \in St_c(p, n)$, define a thin SVD

$$339 \quad \mathbf{Y}_1 (\mathbf{Y}^T \mathbf{Y}_1)^{-1} - \mathbf{Y} = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^T.$$

340 Then the logarithm $\text{Log}_{\mathbf{m}}(\mathbf{m}_1) \in \mathcal{T}_{\mathbf{m}}$ is the velocity vector in $\mathcal{T}_{\mathbf{m}}$ with horizontal lift

$$341 \quad \mathbf{Z} = \mathbf{U} \arctan(\boldsymbol{\Sigma}) \mathbf{V}^T \in \text{Hor}_{\mathbf{Y}}.$$

342 *Remark 4* The logarithm map is only defined on some *open set* $U_{\mathbf{m}}$. This means that for
 343 any point $\mathbf{m}_1 \notin U_{\mathbf{m}}$, the associated matrix $\mathbf{Y}^T \mathbf{Y}_1$ is not invertible, so that the computa-
 344 tion of

$$345 \quad \mathbf{Y}_1 (\mathbf{Y}^T \mathbf{Y}_1)^{-1} - \mathbf{Y}$$

346 can not be done. Note finally that such an open set is strongly related to the *cut-locus* of
 347 a Grassmann manifold [43].

348 2.2 Target Algorithm on compact Stiefel manifolds

349 All the mathematical background summarized in [subsection 2.1](#) can be used to obtain an
 350 interpolation curve between points $\mathbf{m}_1, \dots, \mathbf{m}_N$ on Grassmann manifold $\mathcal{G}(p, n)$ [20, 21],
 351 where each point \mathbf{m}_i corresponds to a parameter value λ_i . Indeed, once a *reference point*
 352 $\mathbf{m}_{i_0} \in \{\mathbf{m}_1, \dots, \mathbf{m}_N\}$ is chosen (see [Figure 4](#)):

- 353 • We use the logarithm map $\text{Log}_{\mathbf{m}_{i_0}}$ to *linearize*, i.e., meaning we define velocity vectors
 354 $v_i := \text{Log}_{\mathbf{m}_{i_0}}(\mathbf{m}_i)$ on the *vector space* $\mathcal{T}_{\mathbf{m}_{i_0}}$.

- 355 • We obtain an interpolation curve $\lambda \mapsto v(\lambda)$ between vectors v_i , using for instance
 356 Lagrangian polynomial, and thus

$$357 \quad v(\lambda_i) = v_i, \quad \forall i = 1, \dots, N.$$

- 358 • Taking the exponential map $\text{Exp}_{\mathbf{m}_{i_0}}$, we obtain back an interpolation curve

$$359 \quad \lambda \mapsto \mathbf{m}(\lambda) := \text{Exp}_{\mathbf{m}_{i_0}}(v(\lambda))$$

360 between the points $\mathbf{m}_1, \dots, \mathbf{m}_N$ on $\mathcal{G}(p, n)$, so that

$$361 \quad \mathbf{m}(\lambda_i) = \mathbf{m}_i, \quad \forall i = 1, \dots, N.$$

362 We propose here to define curves on the compact Stiefel manifold $\mathcal{St}(p, n)$ instead of
 363 the ones defined on the Grassmann manifold $\mathcal{G}(p, n)$. The starting point is a set of matrices
 364 $\mathbf{Y}_1, \dots, \mathbf{Y}_N$ in the compact Stiefel manifold $\mathcal{St}(p, n)$, corresponding to parameter values
 365 $\lambda_1, \dots, \lambda_N$. Once a reference parameter value λ_{i_0} has been chosen, we obtain a curve

$$366 \quad \lambda \mapsto \mathbf{Y}(\lambda)$$

367 where in general,

$$368 \quad \mathbf{Y}(\lambda_i) \neq \mathbf{Y}_i.$$

369 As a consequence, such a curve will not be an interpolation curve between the matrices
 370 $\mathbf{Y}_1, \dots, \mathbf{Y}_N$ (see Remark 5). Before doing so, and to obtain well-defined curves, we need
 371 to make a specific definition:

372 **Definition 3 (Genericity)** *A matrix is said to be generic if all its non-zero singular*
 373 *values are distinct. The set of generic matrices in $\text{Mat}_{n,p}(\mathbb{R})$ is denoted $\text{Mat}_{n,p}^0(\mathbb{R})$.*

374 For any generic matrix $\mathbf{M} \in \text{Mat}_{n,p}^0(\mathbb{R})$, we know that its thin SVD $\mathbf{M} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ is
 375 well defined. Indeed, taking $\sigma_1 > \dots > \sigma_p$ to be its ordered singular values, we can write

$$376 \quad \mathbf{M} = \sum_{i=1}^p \sigma_i \mathbf{u}^i \mathbf{v}_i^T \quad (7)$$

377 where \mathbf{u}_i (resp. \mathbf{v}_i) is a left singular vector associated to σ_i (resp. a right singular vector).

378 All singular values being distinct, the only other possibility is to consider singular vectors

379 $\epsilon_i \mathbf{u}_i$ and $\epsilon_i \mathbf{v}_i$, with $\epsilon_i = \pm 1$, so that the decomposition (7) remains the same. We thus

380 deduce that the target Algorithm below is well defined:

381 **Algorithm 1 (Target algorithm)**

382 • **Inputs:**

383 – Matrices $\mathbf{Y}_1, \dots, \mathbf{Y}_N$ in $\mathcal{St}_c(p, n)$, corresponding to parameter values $\lambda_1 < \dots <$

384 λ_N .

385 – A reference parameter value λ_{i_0} with $i_0 \in \{1, \dots, N\}$.

386 – A parameter value λ .

387 • **Output:** A matrix $\mathbf{Y}(\lambda) \in \mathcal{St}_c(p, n)$.

388 1. Define $\mathbf{Z}_{i_0} := \mathbf{0}$ and for each $k \in \{1, \dots, N\}$ with $k \neq i_0$ compute a thin SVD of the

389 generic matrix

$$390 \quad \mathbf{Y}_k (\mathbf{Y}_{i_0}^T \mathbf{Y}_k)^{-1} - \mathbf{Y}_{i_0} = \mathbf{U}_k \boldsymbol{\Sigma}_k \mathbf{V}_k^T$$

391 and define

$$392 \quad \mathbf{Z}_k := \mathbf{U}_k \arctan(\boldsymbol{\Sigma}_k) \mathbf{V}_k^T, \quad \text{with assumption } \mathbf{Z}_k \in \text{Mat}_{n,p}^0(\mathbb{R}).$$

393 2. Define an $n \times p$ matrix and compute a thin SVD

$$394 \quad \mathbf{Z}(\lambda) := \sum_{i=1}^N \prod_{i \neq j} \frac{\lambda - \lambda_j}{\lambda_i - \lambda_j} \mathbf{Z}_i = \mathbf{U}(\lambda) \boldsymbol{\Sigma}(\lambda) \mathbf{V}(\lambda)^T,$$

395 with assumption $\mathbf{Z}(\lambda) \in \text{Mat}_{n,p}^0(\mathbb{R})$.

396 3. Define the $n \times p$ matrix in $\mathcal{St}_c(p, n)$ (see Remark 3):

$$397 \quad \mathbf{Y}(\lambda) := [\mathbf{Y}_{i_0} \mathbf{V}(\lambda) \cos(\boldsymbol{\Sigma}(\lambda)) + \mathbf{U}(\lambda) \sin(\boldsymbol{\Sigma}(\lambda))] \mathbf{V}(\lambda)^T. \quad (8)$$

398 Note: \cos and \sin act only on diagonal entries.

399 In this algorithm, as already noticed and following the assumptions of genericity, the
400 matrices \mathbf{Z}_k , $\mathbf{Z}(\lambda)$ and $\mathbf{Y}(\lambda)$ do not depend on the choice of matrices in the associated
401 thin SVD.

402 *Remark 5* Using this target Algorithm to parameter value $\lambda := \lambda_k$ leads to some matrix
403 $\mathbf{Y}(\lambda_k)$ generally different from \mathbf{Y}_k (except for $k = i_0$). Thus, such an algorithm computed
404 on compact Stiefel manifold do not produce an *interpolation* on the points $\mathbf{Y}_1, \dots, \mathbf{Y}_N$
405 (see Figure 3). Indeed, to represent an interpolation curve between these points means
406 that if we consider the parameter value $\lambda = \lambda_k$ (with $k \in \{1, \dots, N\}$) as input in the
407 algorithm, one should expect to return as output $\mathbf{Y}(\lambda_k)$ (given by (8)) the initial matrix
408 \mathbf{Y}_k , which is not the case in general.

409 Nevertheless, matrices $\mathbf{Y}(\lambda_k)$ and \mathbf{Y}_k define the same point on the Grassmann manifold
410 $\mathcal{G}(p, n)$, meaning that they both define an *orthonormal basis of the same subspace* \mathbf{m}_k
411 (see Remark 4). As a consequence, a projection matrix onto the subspace \mathbf{m}_k is given by
412 $\mathbf{Y}(\lambda_k)^T \mathbf{Y}(\lambda_k)$ or equivalently by $\mathbf{Y}_k^T \mathbf{Y}_k$.

413 *Example 2* Take for instance the compact Stiefel manifold $\mathcal{St}_c(2, 5)$, and the three matrices

$$414 \quad \mathbf{Y}_1 := \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{Y}_2 := \begin{bmatrix} \frac{\sqrt{3}}{3} & \frac{\sqrt{3}}{3} \\ 0 & \frac{\sqrt{3}}{3} \\ \frac{\sqrt{3}}{3} & 0 \\ -\frac{\sqrt{3}}{3} & \frac{\sqrt{3}}{3} \\ 0 & 0 \end{bmatrix}, \quad \mathbf{Y}_3 := \begin{bmatrix} \frac{\sqrt{3}}{3} & -\frac{\sqrt{6}}{6} \\ 0 & \frac{\sqrt{6}}{4} \\ \frac{\sqrt{3}}{3} & \frac{\sqrt{6}}{12} \\ 0 & \frac{\sqrt{6}}{4} \\ \frac{\sqrt{3}}{3} & \frac{\sqrt{6}}{12} \end{bmatrix}$$

415 which correspond respectively to $\lambda_1 = 15$, $\lambda_2 = 22$ and $\lambda_3 = 27$. Choosing the reference

416 parameter value to be λ_1 and following the target Algorithm 1 we obtain

$$417 \quad \mathbf{Y}_2(\mathbf{Y}_1^T \mathbf{Y}_2)^{-1} - \mathbf{Y}_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & -1 \\ -1 & 2 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{Y}_3(\mathbf{Y}_1^T \mathbf{Y}_3)^{-1} - \mathbf{Y}_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$$

418 Taking $\lambda = \lambda_2$ and $\lambda = \lambda_3$ as inputs in the algorithm, we finally obtain the matrices (with

419 computation done using 5 digits):

$$420 \quad \mathbf{Y}(\lambda_2) = \begin{bmatrix} 0.77460 & 0.25820 \\ 0.25820 & 0.51640 \\ 0.51640 & -0.25820 \\ -0.25820 & 0.77460 \\ 0 & 0 \end{bmatrix} \neq \mathbf{Y}_2, \quad \mathbf{Y}(\lambda_3) = \begin{bmatrix} 0.67860 & -0.19876 \\ -0.19876 & 0.57922 \\ 0.47984 & 0.38046 \\ -0.19876 & 0.57922 \\ 0.47984 & 0.38046 \end{bmatrix} \neq \mathbf{Y}_3.$$

421 3 Space-Time Interpolation on compact Stiefel manifolds

422 As already noticed, POD is extracting the optimal space structures and the associated

423 temporal modes. An important property is that the spatial and temporal orthogonal modes

424 are *coupled*: each space component is associated with a temporal component partner
 425 and there is a one-to-one correspondence between both spaces. Taking advance of this
 426 decomposition into orthogonal modes, it is natural to try a *Space-Time* interpolation on
 427 compact Stiefel manifolds based on the target Algorithm 1, instead of an interpolation of
 428 the space part alone, followed by a Galerkin approach as is classically done [20,21].

429 As a starting point, take a set of *snapshot matrices* $\mathbf{S}^{(1)}, \dots, \mathbf{S}^{(N)}$, where each matrix
 430 $\mathbf{S}^{(k)} \in \text{Mat}_{n,m}(\mathbb{R})$ corresponds to a given parameter value $\lambda_k \in \mathbb{R}$, with $\lambda_1 < \dots < \lambda_N$
 431 and $n = 3N_s$ corresponding to the spatial part, while $m = N_t$ corresponds to the temporal
 432 part. For a given mode $p \leq N_t$, our goal is to

- 433 1. Extract in a unique way a POD of mode p of each matrix $\mathbf{S}^{(k)}$, so that we have a well
 434 defined map

$$435 \quad \mathbf{S}^{(k)} \in \text{Mat}_{n,m}(\mathbb{R}) \mapsto \mathbf{S}_p^{(k)} \in \text{Mat}_{n,m}(\mathbb{R}).$$

- 436 2. Obtain for each $\mathbf{S}_p^{(k)} \in \text{Mat}_{n,m}(\mathbb{R})$ a unique matrix $\Phi_p^{(k)} \in \mathcal{St}_c(p, n)$ for the spatial
 437 part and another unique matrix $\Psi_p^{(k)} \in \mathcal{St}_c(p, m)$ for the temporal part.
- 438 3. Use the target Algorithm 1 on matrices $\Phi_p^{(k)}$ first, and then on matrices $\Psi_p^{(k)}$, in order
 439 to obtain two curves

$$440 \quad \lambda \mapsto \Phi(\lambda), \quad \lambda \mapsto \Psi(\lambda) \tag{9}$$

441 *which are not interpolated curves*, as in general $\Phi(\lambda_k) \neq \Phi_p^{(k)}$ and $\Psi(\lambda_k) \neq \Psi_p^{(k)}$ (see
 442 Remark 5).

- 443 4. Define an interpolation curve $\lambda \mapsto \mathbf{S}(\lambda)$ between matrices $\mathbf{S}_p^{(1)}, \dots, \mathbf{S}_p^{(N)}$, using curves
 444 obtained by (9).

445 We now detail two key points: the first concerns a new type of SVD, called oriented SVD,
 446 which allows defining the matrices $\Phi_p^{(k)}$ and $\Psi_p^{(k)}$ in a unique way. Finally, we will explain

447 in [subsection 3.2](#) how to construct the curve $\lambda \mapsto \mathbf{S}(\lambda)$, which requires the introduction
 448 of a mixed part.

449 3.1 Oriented SVD on generic matrices

450 As already noticed in [section 2](#), any computation of a POD of mode p of a matrix $\mathbf{S} \in$
 451 $\text{Mat}_{n,m}(\mathbb{R})$ can be obtained from a SVD. Suppose now that \mathbf{S} is of rank $r \geq p$. Any SVD
 452 of \mathbf{S} with singular values $\sigma_1 > \dots > \sigma_r$ leads to *spatial orthonormal vectors* ϕ_1, \dots, ϕ_r
 453 in \mathbb{R}^n (the left singular vectors) and *temporal orthonormal vectors* ψ_1, \dots, ψ_r in \mathbb{R}^m (the
 454 right singular vectors). A POD of mode p then writes

$$\mathbf{S}_p = \mathbf{\Phi}_p \mathbf{\Sigma}_p \mathbf{\Psi}_p^T, \quad \mathbf{\Phi}_p := [\phi_1, \dots, \phi_p], \quad \mathbf{\Sigma}_p := \text{diag}(\sigma_1, \dots, \sigma_p), \quad \mathbf{\Psi}_p := [\psi_1, \dots, \psi_p].$$

455 (10)

456 Now, because of sign indeterminacy of the spatial vectors ϕ_i and temporal vectors ψ_i , the
 457 matrices $\mathbf{\Phi}_p, \mathbf{\Psi}_p$ are not uniquely defined.

458 To overcome this difficulty, we need to introduce a *new SVD* so that, under the as-
 459 sumption of genericity (see [Definition 3](#)), the matrices $\mathbf{\Phi}_p$ and $\mathbf{\Psi}_p$ given by [\(10\)](#) can be
 460 well-defined.

461 The main idea of the new SVD introduced here is to make an intrinsic choice on the
 462 orientation for each space and temporal vector. Indeed, for each spatial vector ϕ , only two
 463 choices can occur: ϕ or $-\phi$ (thus inducing a choice on the associated temporal vector). A
 464 choice of orientation is then made as follows. Taking the column vectors $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_m]$
 465 and \mathbf{s} to be the first column vector such that the scalar product $\langle \mathbf{s}, \phi \rangle$ is non zero, we
 466 impose the sign taking $\langle \mathbf{s}, \phi \rangle > 0$.

467 Let us now give all details to compute the oriented SVD before obtaining algorithm 2.
 468 A first Lemma, obtained by direct computation, allows us to use a column vector of the
 469 initial snapshot matrix \mathbf{S} to choose orientation:

470 **Lemma 1** *Let us consider $\mathbf{s}_1, \dots, \mathbf{s}_m \in \mathbb{R}^n$ to be the column vectors of $\mathbf{S} \in \text{Mat}_{n,m}(\mathbb{R})$
 471 and take $\phi \in \mathbb{R}^n$ to be a unit spatial vector of \mathbf{S} , associated with a non-zero singular value
 472 σ . Then, there exists $i \in \{1, \dots, m\}$ such that $\langle \mathbf{s}_i, \phi \rangle = \mathbf{s}_i^T \phi \neq 0$.*

473 From this, for any unit spatial vector $\phi \in \mathbb{R}^n$ of \mathbf{S} , let us define $\mathbf{s}(\phi)$ to be the first column
 474 vector \mathbf{s}_i in $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_m]$ such that $\langle \phi, \mathbf{s}_i \rangle \neq 0$:

$$475 \quad \mathbf{s}(\phi) := \mathbf{s}_i, \quad i := \min \{j, \quad \langle \mathbf{s}_j, \phi \rangle \neq 0\}. \quad (11)$$

476 Any spatial eigenvector can therefore have a specific orientation:

477 **Definition 4 (Oriented eigenvectors)** *Let $\mathbf{S} \in \text{Mat}_{n,m}(\mathbb{R})$ and $\phi \in \mathbb{R}^n$ a unit spatial
 478 vector associated to a non-zero singular value σ . Then ϕ is said to be oriented if $\langle \mathbf{s}(\phi), \phi \rangle >$
 479 0 .*

480 From all this, let us now deduce the new SVD:

481 **Lemma 2 (Oriented SVD)** *Let $\mathbf{S} \in \text{Mat}_{n,m}^0(\mathbb{R})$ ($m \leq n$) of rank r such that all its non-
 482 zero singular values are distinct. Then, there exists one and only one couple of matrices*

$$483 \quad \mathbf{\Phi} = [\phi_1, \dots, \phi_r] \in \text{Mat}_{n,r}(\mathbb{R}), \quad \mathbf{\Psi} = [\psi_1, \dots, \psi_r] \in \text{Mat}_{m,r}(\mathbb{R}) \quad (12)$$

485 such that

$$486 \quad \langle \phi_i, \phi_j \rangle = \langle \psi_i, \psi_j \rangle = \delta_{ij}, \quad \mathbf{S} = \mathbf{\Phi} \mathbf{\Sigma} \mathbf{\Psi}^T, \quad \mathbf{\Sigma} := \text{Diag}(\sigma_1, \dots, \sigma_r) \in \text{Mat}_{r,r}(\mathbb{R}) \quad (13)$$

487 and ϕ_i are oriented spatial unit eigenvectors:

$$488 \quad \langle \mathbf{s}(\phi_i), \phi_i \rangle > 0 \quad (14)$$

489 with $\mathbf{s}(\phi_i)$ defined by (11). Such a decomposition is called an oriented SVD.

490 *Proof* First, any couple (ϕ, ψ) of spatial-temporal unit eigenvector for \mathbf{S} is defined modulo
491 ± 1 , and ψ is obtained in a unique way from ϕ .

492 Let us suppose now we do not have uniqueness, so that there exist two unit spatial
493 vectors ϕ and ϕ' associated to σ such that

$$494 \quad \langle \mathbf{s}(\phi), \phi \rangle > 0 \text{ and } \langle \mathbf{s}(\phi'), \phi' \rangle > 0.$$

495 We necessary have $\phi' = -\phi$ and $\mathbf{s}(\phi) = \mathbf{s}(\phi')$ so we deduce that

$$496 \quad \langle \mathbf{s}(\phi'), \phi' \rangle > = -\langle \mathbf{s}(\phi), \phi \rangle > 0$$

497 which is a contradiction, and we can conclude our proof.

498 We give now an algorithm to obtain such an oriented SVD:

499 **Algorithm 2 (Oriented SVD)**

500 • **Inputs:** $m \leq n$ and $\mathbf{S} \in \text{Mat}_{n,m}^0(\mathbb{R})$ of rank r .

501 • **Output:** Unique matrices Φ and Ψ for an oriented SVD of \mathbf{S} .

502 1. Compute a SVD of \mathbf{S} so that to obtain spatial unit vectors ϕ_1, \dots, ϕ_r and temporal
503 unit vectors ψ_1, \dots, ψ_r .

504 2. Consider the column vectors $\mathbf{s}_1, \dots, \mathbf{s}_m$ of \mathbf{S} .

505 3. For $i = 1, \dots, r$ define

$$506 \quad \varepsilon_i := \frac{\langle \mathbf{s}(\phi_i), \phi_i \rangle}{|\langle \mathbf{s}(\phi_i), \phi_i \rangle|}$$

507 where $\mathbf{s}(\phi_i)$ is the first column vector \mathbf{s} of \mathbf{S} such that $\langle \phi_i, \mathbf{s} \rangle \neq 0$, see (11).

508 4. For $i = 1, \dots, r$, make sign replacement

$$509 \quad \phi_i \leftarrow \varepsilon_i \phi_i, \quad \psi_i \leftarrow \varepsilon_i \psi_i.$$

510 *Example 3* Assume the rank 3 matrix

$$511 \quad \mathbf{S} = \begin{bmatrix} 1 & 0 & 1 \\ -1 & 1 & 0 \\ 0 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} = [\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3]$$

512 where a unit spatial vector corresponding to the largest singular value is given by (with 5
513 digits)

$$514 \quad \phi_1 = \begin{bmatrix} -0.31145 \\ 0.41763 \\ 0.74265 \\ -0.28294 \\ -0.31145 \end{bmatrix}$$

515 and we can check that $\mathbf{s}(\phi_1) = \mathbf{s}_1$ with $\langle \phi_1, \mathbf{s}_1 \rangle < 0$ so that we consider $-\phi_1$ instead of
516 ϕ_1 , and so on.

517 *3.2 Space–Time interpolation algorithm*

518 In this subsection, we define a Space–Time interpolation on any family of POD of mode
 519 p taken from *generic* snapshot matrices (see an overview in Figure 5). Such interpolation
 520 captures both the spatial and temporal part of such matrices, which is necessary from the
 521 point of view of mechanical equations, but we will also need to define a specific *mixed part*
 522 of each POD (see lemma 3). The method is formalized for the general case of interest in
 523 mechanical applications of an arbitrary number of pre-computed POD basis N , and an
 524 arbitrary number of parameters N_p per training point. In the following, the method is
 525 described for an arbitrary number of basis N and a single parameter $N_p = 1$ by applying
 526 a univariate Lagrange interpolation. However, its generalization in a multivariate case is
 527 straightforward using an appropriate multivariate interpolation scheme [?].

528 Take back parameter values $\lambda_1 < \dots < \lambda_N$, corresponding to snapshot matrices
 529 $\mathbf{S}^{(1)}, \dots, \mathbf{S}^{(N)}$ in $\text{Mat}_{n,m}(\mathbb{R})$, with $n = 3N_s$ and $m = N_t$. To make use of the oriented
 530 SVD, let us suppose:

531 **Genericity assumption:** All snapshot matrices $\mathbf{S}^{(1)}, \dots, \mathbf{S}^{(N)}$ have distinct non-zero
 532 singular values.

533 Take now p to be some integer (less or equal than the minimum rank of all matrices
 534 $\mathbf{S}^{(k)}$). Using the oriented SVD given by Algorithm 2, we can consider a POD of mode p
 535 on each matrix $\mathbf{S}^{(k)}$:

$$536 \quad \mathbf{S}_p^{(k)} := \mathbf{\Phi}_p^{(k)} \mathbf{\Sigma}_p^{(k)} \mathbf{\Psi}_p^{(k)T} \in \text{Mat}_{n,m}(\mathbb{R}) \quad (15)$$

537 where Σ_k corresponds to singular values, and $\Phi_p^{(k)}$ as well as $\Psi_p^{(k)}$ uniquely define points
 538 in a compact Stiefel manifold:

$$539 \quad \Phi_p^{(k)} := [\phi_1^{(k)}, \dots, \phi_p^{(k)}] \in \mathcal{St}_c(p, n), \quad \Psi_p^{(k)} := [\psi_1^{(k)}, \dots, \psi_p^{(k)}] \in \mathcal{St}_c(p, m), \quad (16)$$

540 Recall that in previous equation, $\phi_1^{(k)}, \dots, \phi_p^{(k)}$ (resp. $\psi_1^{(k)}, \dots, \psi_p^{(k)}$) correspond to spatial
 541 oriented eigenvectors (resp. temporal ones) of $\mathbf{S}^{(k)}$.

542 Using the target Algorithm 1 first for the spatial matrices $\Phi_p^{(k)}$ and then for the
 543 temporal matrices $\Psi_p^{(k)}$, we obtain two curves

$$544 \quad \lambda \mapsto \Phi(\lambda), \quad \lambda \mapsto \Psi(\lambda).$$

545 Now, our goal is to produce an interpolation curve between the matrices $\mathbf{S}_p^{(k)}$, taking into
 546 account both spatial and temporal curves defined above. Such a curve is given by

$$547 \quad \lambda \mapsto \mathbf{S}(\lambda) := \Phi(\lambda)\mathbf{M}(\lambda)\Psi(\lambda)^T \text{ with } \mathbf{S}(\lambda_k) = \mathbf{S}_p^{(k)}. \quad (17)$$

548
 549 **Lemma 3** For a curve defined (17) to be an interpolation curve between the matrices $\mathbf{S}_p^{(k)}$,
 550 we necessary have

$$551 \quad \mathbf{M}(\lambda_k) = \Phi(\lambda_k)^T \mathbf{S}_p^{(k)} \Psi(\lambda_k). \quad (18)$$

552 *Proof* To satisfy (17), we necessary have

$$553 \quad \mathbf{S}(\lambda_k) = \Phi(\lambda_k)\mathbf{M}(\lambda_k)\Psi(\lambda_k)^T = \mathbf{S}_p^{(k)}, \quad (19)$$

554 where, by construction, we have $\Phi(\lambda_k) \in \mathcal{St}_c(p, n)$ and $\Psi(\lambda_k) \in \mathcal{St}_c(p, m)$ (see target
 555 Algorithm 1). We thus have

$$556 \quad \Phi(\lambda_k)^T \Phi(\lambda_k) = \Psi(\lambda_k)^T \Psi(\lambda_k) = \mathbf{I}_p,$$

so that by the left and right multiplication of (19) we obtain formula (18) for the mixed part.

Now we have:

Lemma 4 Let $\lambda \mapsto \mathbf{M}(\lambda) \in \text{Mat}_{p,p}(\mathbb{R})$ be any interpolated curve between the mixed part matrices

$$\mathbf{M}_k := \Phi(\lambda_k)^T \mathbf{S}_p^{(k)} \Psi(\lambda_k) \in \text{Mat}_{p,p}(\mathbb{R})$$

so that $\mathbf{M}(\lambda_k) = \mathbf{M}_k$ for $k = 1, \dots, N$. Then, using the curve $\lambda \mapsto \Phi(\lambda)$ (resp. $\lambda \mapsto \Psi(\lambda)$) defined by the target Algorithm 1 applied on the matrices $\Phi_p^{(k)}$ (resp. $\Psi_p^{(k)}$), the curve

$$\kappa : \lambda \mapsto \Phi(\lambda) \mathbf{M}(\lambda) \Psi(\lambda)^T \quad (20)$$

is an interpolated curve between the matrices $\mathbf{S}_p^{(1)}, \dots, \mathbf{S}_p^{(N)}$, so that $\kappa(\lambda_k) = \mathbf{S}_p^{(k)}$ for each $k = 1, \dots, N$.

Proof We need to check that $\kappa(\lambda_k) = \mathbf{S}_p^{(k)}$ for each $k = 1, \dots, N$. Now:

$$\begin{aligned} \kappa(\lambda_k) &= \Phi(\lambda_k) \mathbf{M}(\lambda_k) \Psi(\lambda_k)^T = \Phi(\lambda_k) \mathbf{M}_k \Psi(\lambda_k)^T \\ &= \Phi(\lambda_k) \Phi(\lambda_k)^T \mathbf{S}_p^{(k)} \Psi(\lambda_k) \Psi(\lambda_k)^T \\ &= \Phi(\lambda_k) \Phi(\lambda_k)^T \underbrace{\Phi_p^{(k)} \sum_p^{(k)} (\Psi_p^{(k)})^T}_{\mathbf{S}_p^{(k)}} \Psi(\lambda_k) \Psi(\lambda_k)^T \end{aligned}$$

where $\Phi(\lambda_k) \Phi(\lambda_k)^T$ corresponds to the projection matrix on the subspace $\mathbf{m}_k := \pi(\Phi(\lambda_k)) = \pi(\Phi_k)$ (see Remark 5) so that

$$\Phi(\lambda_k) \Phi(\lambda_k)^T \Phi_p^{(k)} = \Phi_p^{(k)} \quad (21)$$

and the same being true for the temporal part, we obtain the proof of the lemma.

572 The Space–Time interpolation algorithm is now given by:

573 **Algorithm 3 (Space–Time interpolation)**

574 • **Inputs:**

575 – Generic matrices $\mathbf{S}^{(1)}, \dots, \mathbf{S}^{(N)}$ in $\text{Mat}_{n,m}^0(\mathbb{R})$ ($m \leq n$), corresponding to parameter
576 values $\lambda_1 < \dots < \lambda_N$.

577 – A reference parameter value λ_{i_0} with $i_0 \in \{1, \dots, N\}$.

578 – A mode $p \leq m$.

579 – A parameter value $\tilde{\lambda}$.

580 • **Output:** A matrix $\tilde{\mathbf{S}} \in \text{Mat}_{n,m}(\mathbb{R})$.

581 1. Compute an oriented SVD on each matrix $\mathbf{S}^{(k)}$ and write a POD of mode p

582
$$\mathbf{S}_p^{(k)} := \Phi_p \Sigma_p^{(k)} (\Psi_p^{(k)})^T \in \text{Mat}_{n,m}(\mathbb{R})$$

583 with $\Phi_p^{(k)} \in \mathcal{S}t_c(p, n)$ and $\Psi_p^{(k)} \in \mathcal{S}t_c(p, m)$ uniquely defined.

584 2. Consider the target Algorithm 1 applied to the spatial parts $\Phi_p^{(1)}, \dots, \Phi_p^{(N)}$, reference
585 parameter value λ_{i_0} and each of the $N + 1$ parameter values $\lambda_1, \dots, \lambda_N, \tilde{\lambda}$, so from (8)

586 we can define matrices in $\mathcal{S}t_c(p, n)$:

587
$$\Phi(\lambda_k) := \mathbf{Y}(\lambda_k), \quad \Phi(\tilde{\lambda}) := \mathbf{Y}(\tilde{\lambda}). \quad (22)$$

588 3. Consider the target Algorithm 1 applied to the temporal parts $\Psi_p^{(1)}, \dots, \Psi_p^{(N)}$ reference
589 parameter value λ_{i_0} and each of the $N + 1$ parameter values $\lambda_1, \dots, \lambda_N, \tilde{\lambda}$, so from (8)

590 we can define matrices in $\mathcal{S}t_c(p, m)$:

591
$$\Psi(\lambda_k) := \mathbf{Y}(\lambda_k), \quad \Psi(\tilde{\lambda}) := \mathbf{Y}(\tilde{\lambda}). \quad (23)$$

592 4. For each $k = 1, \dots, N$, define the square matrix of the mixed part

$$593 \quad \mathbf{M}_k := \boldsymbol{\Phi}(\lambda_k)^T \mathbf{S}_p^{(k)} \boldsymbol{\Psi}(\lambda_k) \in \text{Mat}_{p,p}(\mathbb{R}). \quad (24)$$

594 5. Use a standard interpolation on square matrices $\mathbf{M}_1, \dots, \mathbf{M}_N$, for instance:

$$595 \quad \mathbf{M}(\tilde{\lambda}) := \sum_{i=1}^N \prod_{i \neq j} \frac{\tilde{\lambda} - \lambda_j}{\lambda_i - \lambda_j} \mathbf{M}_i \quad (25)$$

596 6. Using the spatial part $\boldsymbol{\Phi}(\tilde{\lambda}) \in \mathcal{S}t_c(p, n)$ from (22), the temporal part $\boldsymbol{\Psi}(\tilde{\lambda}) \in \mathcal{S}t_c(p, m)$

597 from (23), and the mixed part $\mathbf{M}(\tilde{\lambda}) \in \text{Mat}_{p,p}(\mathbb{R})$ from (25), the interpolated snapshot

598 matrix corresponding to $\tilde{\lambda}$ is finally given by

$$599 \quad \tilde{\mathbf{S}} := \boldsymbol{\Phi}(\tilde{\lambda}) \mathbf{M}(\tilde{\lambda}) \boldsymbol{\Psi}(\tilde{\lambda})^T \in \text{Mat}_{n,m}(\mathbb{R}).$$

600 4 Rigid-Viscoplastic FEM Formulation

601 The main defining characteristic of the RVP formulation is that it neglects the elasticity
602 effects. This *idealization* is based on the fact that elastic components of strain remain
603 small as compared with irreversible strains. This means that the additive decomposition
604 of the total strain-rate tensor $\dot{\varepsilon}_{ij} = \dot{\varepsilon}_{ij}^e + \dot{\varepsilon}_{ij}^p$ simplifies to $\dot{\varepsilon}_{ij} = \dot{\varepsilon}_{ij}^p$, where $\dot{\varepsilon}_{ij}^e$ is the
605 elastic component of the strain-rate tensor, $\dot{\varepsilon}_{ij}^p$ is the plastic component and $\dot{\varepsilon}_{ij}$ is the
606 total strain-rate tensor. Therefore, the RVP formulation turns out to be very similar to
607 fluid flow problems, and it is also called *flow formulation* [44]. Although it is not possible
608 to calculate the residual stresses and the spring-back effect, the flow formulation presents
609 several advantages. Unlike the elastoplastic FEM, the RVP formulation, even though more
610 approximate, is more stable, simpler to be implemented in computer codes, and can use
611 relatively larger time increments, thus improving the computational efficiency. A thorough
612 overview of the foundation of the theory can be found in [34, 1].

613 *4.1 Governing Field Equations*

614 Classical rigid viscoplastic problems consider the plastic deformation of an isotropic body
 615 occupying a domain $\Omega \subset \mathbb{R}^3$. The domain Ω and its boundary $\partial\Omega$ represent the current
 616 configuration of a body according to the *Updated Lagrangian* formulation. The governing
 617 equations that have to be satisfied are:

618 (a) Equilibrium condition:

619
$$\sigma_{ij,j} = 0$$

620 (b) Compatibility conditions:

621
$$\dot{\varepsilon}_{ij} = \frac{1}{2}(v_{i,j} + v_{j,i})$$

622 (c) Yield criterion:

623
$$\bar{\sigma} := \left(\frac{2}{3} \sigma'_{ij} \sigma'_{ij} \right)^{\frac{1}{2}} = \bar{\sigma}(\bar{\varepsilon}, \dot{\bar{\varepsilon}}, T)$$

624 (d) Constitutive equations:

625
$$\sigma'_{ij} = \frac{2}{3} \frac{\bar{\sigma}}{\bar{\varepsilon}} \dot{\varepsilon}_{ij}, \quad \dot{\bar{\varepsilon}} = \left(\frac{2}{3} \dot{\varepsilon}_{ij} \dot{\varepsilon}_{ij} \right)^{\frac{1}{2}} \quad (26)$$

626 (e) Incompressibility condition:

627
$$\dot{\varepsilon}_v := \dot{\varepsilon}_{kk} = 0$$

(f) Boundary conditions:

$$\begin{aligned} \mathbf{v} &= \hat{\mathbf{v}} && \text{on} && \partial\Omega_v \\ \mathbf{F} &= \hat{\mathbf{F}} && \text{on} && \partial\Omega_F \\ \text{friction and contact} &&& \text{on} && \partial\Omega_c \end{aligned}$$

628 In the above equations $\boldsymbol{\sigma} = (\sigma_{ij})$ is the stress tensor, $\dot{\boldsymbol{\epsilon}} = (\dot{\epsilon}_{ij})$ is the strain rate tensor,
 629 v_i are velocity components, $\bar{\sigma}$ is the effective stress, $\dot{\bar{\epsilon}}$ is the second invariant of $\dot{\boldsymbol{\epsilon}}$ called
 630 effective strain rate, and $\boldsymbol{\sigma}' = (\sigma'_{ij})$ is the deviatoric stress tensor defined by $\sigma'_{ij} = \sigma_{ij} -$
 631 $\delta_{ij}\sigma_{kk}/3$.

632 The hat symbol $\hat{}$ denotes prescribed values. Generally, the boundary $\partial\Omega$ consists
 633 of three distinct parts: over $\partial\Omega_v$ velocity conditions are prescribed (essential boundary
 634 conditions), $\partial\Omega_F$ is the part where the traction conditions are imposed in the form of
 635 nodal point forces (natural boundary conditions), while the boundary conditions along
 636 $\partial\Omega_c$ are mixed, and neither the velocity nor the force can be described. Therefore, we
 637 have the disjoint union:

$$638 \quad \partial\Omega = \partial\Omega_v \cup \partial\Omega_F \cup \partial\Omega_c \quad (27)$$

639 4.2 Variational form

640 In a variational formulation, the functional Π (energy rate) is defined by an integral form
 641 in accordance with the virtual work-rate principle

$$642 \quad \Pi(v) := \int_{\Omega} \bar{\sigma} \dot{\bar{\epsilon}} dV - \int_{\partial\Omega_F} F_i v_i dS \quad (28)$$

643 where the first term in (28) represents the internal deformation work-rate, whereas the
 644 second term represents the work-rate done by the external forces. F_i denotes prescribed
 645 surface tractions on the boundary surface $\partial\Omega_F$. Recalling the MarKov-Hill [45,46] varia-
 646 tional principle, among all virtual (admissible) continuous and continuously differentiable
 647 velocity fields v_i satisfying the conditions of compatibility and incompressibility, as well

648 as the velocity boundary conditions, the real velocity field gives to the functional Π a
 649 stationary value, i.e., the first-order variation vanishes. Moreover, in order to relax the
 650 incompressibility constraint condition $\dot{\epsilon}_v = \dot{\epsilon}_{kk} = 0$ on an admissible velocity field, a
 651 classical penalized form is used

$$652 \quad \delta \Pi := \int_{\Omega} \bar{\sigma} \delta \dot{\epsilon} dV + \frac{1}{2} \int_{\Omega} K \dot{\epsilon}_v \delta \dot{\epsilon}_v dV - \int_{\partial \Omega_F} F_i \delta v_i dS = 0 \quad (29)$$

653 where K is a large positive constant which penalizes the dilatational strain-rate com-
 654 ponent. It can be shown that the mean stress is $\sigma_m = K \dot{\epsilon}_{kk}$.

655 *Remark 6* A limitation of the Updated Lagrangian method for large deformation problems
 656 is the excessive element distortion. To this end, remeshing processes are necessary to
 657 simulate unconstrained plastic flows. A mesh generation process is activated in case of
 658 zero or negative determinant of the Jacobian matrix, or due to various element quality
 659 criteria. Then, a new mesh is calculated conforming to the current state of the geometry
 660 followed by an interpolation of the state variables between the old and the newly generated
 661 mesh. Thus, the information of the remapping process has to adequately be transferred
 662 to the ROM basis obtained using the POD snapshot method. We remark that at this first
 663 attempt, we avoid remeshings of the workpiece during the course of the simulation. This
 664 topic will be addressed in a future investigation.

665 4.3 Discretization and iteration

666 The discretization of the functional follows the standard procedure of the finite element
 667 method. Eq. (29) is expressed in terms of nodal point velocities v_i and their variations

668 δv_i . Using the variational principle

$$669 \quad \delta \Pi = \sum_{m=1}^M \frac{\partial \Pi^{(m)}}{\partial v_i} \delta v_i = 0, \quad i = 1, 2, \dots, 2N_s, \quad (30)$$

670 where δv_i are arbitrary except that they must be zero to satisfy the corresponding essential
671 boundary conditions, and M denotes the number of elements. From the arbitrariness of
672 δv_i , a set of algebraic equations (stiffness equations) are obtained

$$673 \quad \frac{\partial \Pi}{\partial v_i} = \sum_{m=1}^M \frac{\partial \Pi^{(m)}}{\partial v_i} = 0. \quad (31)$$

674 As the resulting algebraic equations are highly nonlinear, they linearized by the Taylor
675 expansion near an assumed velocity field $\mathbf{v} = \mathbf{v}_0$ as

$$676 \quad \left. \frac{\partial \Pi}{\partial v_i} \right|_{\mathbf{v}=\mathbf{v}_0} + \left. \frac{\partial^2 \Pi}{\partial v_i \partial v_j} \right|_{\mathbf{v}=\mathbf{v}_0} \Delta v_j = 0 \quad (32)$$

677 where the first factor of the second term is also known as the Jacobian of the system
678 (Hessian matrix), and Δv_j is a first-order correction of the velocity component v_j . Solv-
679 ing (32) with respect to Δv_j , the assumed velocity field is updated by the form (written
680 in vector notation)

$$681 \quad \mathbf{v}^{(i)} = \mathbf{v}^{(i-1)} + \alpha (\Delta \mathbf{v})^{(i)} \quad (33)$$

682 where $0 \leq \alpha \leq 1$ and i is the iteration step. The solution is obtained by the Direct
683 iteration method [34, 47] and/or by Newton-Raphson type methods. The iteration process
684 is repeated until the following described convergence criteria are satisfied simultaneously

$$685 \quad \frac{\|\Delta \mathbf{v}\|_{L_2}}{\|\mathbf{v}\|_{L_2}} \leq e_1, \quad \left\| \frac{\partial \Pi}{\partial \mathbf{v}} \right\|_{L_2} \leq e_2 \quad (34)$$

686 namely, the velocity error norm and the norm of the residual equations, where e_1 and
 687 e_2 are sufficiently small specified tolerance numbers.

688 4.4 Heat Transfer Analysis

689 In the present model, a thermodynamically sound derivation is adopted using the conser-
 690 vation of energy

$$691 \quad -\rho c \frac{\partial T}{\partial t} + k \nabla^2 T + \xi \bar{\sigma} \dot{\varepsilon} = 0 \quad (35)$$

692 where ρc is the volume-specific heat of the material, $\xi \bar{\sigma} \dot{\varepsilon}$ represents the work heat
 693 rate per unit volume due to plastic deformation, k is the thermal conductivity, T is the
 694 temperature and ξ is a coefficient that presents the fraction of the deformation energy
 695 dissipated into heat also known as the Taylor-Quinney coefficient.

696 In a weak form, and using the divergence theorem

$$697 \quad - \int_{\Omega} \xi \bar{\sigma} \dot{\varepsilon} \delta T dV + \int_{\Omega} k \nabla T \delta(\nabla T) dV + \int_{\Omega} \rho c \frac{\partial T}{\partial t} \delta T dV - \int_{\partial\Omega} q_n \delta T dS = 0 \quad (36)$$

698 where

$$699 \quad q_n := k \frac{\partial T}{\partial n} \quad (37)$$

700 is the heat flux across the boundary $\partial\Omega$ and n denotes the unit normal vector to the
 701 boundary surface $\partial\Omega$.

702 In standard finite element books, e.g. [48], it can be seen that the heat balance equa-
 703 tions such as (36), upon finite element discretization are reduced to the form:

$$704 \quad \mathbf{C}\dot{\mathbf{T}} + \mathbf{K}\mathbf{T} = \mathbf{Q} \quad (38)$$

705 where \mathbf{C} is the heat capacity matrix, \mathbf{K} denotes the heat conduction matrix, \mathbf{Q} is
 706 the heat flux vector, \mathbf{T} is the vector of nodal point temperatures, and $\dot{\mathbf{T}}$ is the rate of
 707 temperature increase vector of nodal points.

708 The theory necessary to integrate (38) can be found in numerical analysis books [49,
 709 50]. It suffices to say that one-step time integration is used. The convergence of a scheme
 710 requires consistency and stability. Consistency is satisfied by a general time integration
 711 scheme

$$712 \quad {}^{t+\Delta t}\mathbf{T} = {}^t\mathbf{T} + \Delta t[(1 - \theta){}^t\dot{\mathbf{T}} + \theta{}^{t+\Delta t}\dot{\mathbf{T}}] \quad (39)$$

713 where θ is a parameter varying between 0 and 1 ($\theta = 0$: Forward difference, $\theta = 1/2$:
 714 Crank-Nicholson, $\theta = 2/3$: Galerkin, $\theta = 1$: Backward difference).

715 *Remark 7 Unconditional stability* is obtained for $\theta \geq 0.5$. This is important, because it is
 716 desirable to take time steps as large as the deformation formulation allows, since this is
 717 the most expensive part of the process.

718 4.5 Computational Procedure for Thermo-Mechanical Analysis

719 For solving *coupled* thermomechanical problems, two different approaches can be used.
 720 In the traditional *monolithic* approach, a single solver is in charge of the solution of

721 the entire system of equations. In an alternative approach, the mechanical and thermal
 722 solvers deal respectively with the viscoplastic flow and the thermal field equations. Thus,
 723 in the so-called *staggered solution* procedure used here, the state of the system is advanced
 724 by sequentially executing and exchange information between these two solvers [51]. The
 725 equations for the mechanical analysis and the temperature calculation are *strongly coupled*,
 726 thereby making necessary the simultaneous solution of the finite element counterparts [34,
 727 52, 53].

728 5 Numerical Investigations

729 The purpose of this section is to evaluate the performance of the ST POD interpolation
 730 using the velocity and temperature fields during the course of the simulation of the forming
 731 process. As a benchmark test case, a rectangular cross-section bar is compressed between
 732 two parallel flat dies under the condition of a constant shear friction factor m at the
 733 die-workpiece interface. The initial workpiece has dimensions $h = 20$ mm (height) and
 734 $w = 20$ mm (width). Plane strain conditions are considered. Due to the symmetry of the
 735 problem, only one quarter of the cross-section is analyzed. The velocity of the upper and
 736 the lower die is set to $v = 1$ mm/s. The initial temperature of the die and the workpiece
 737 is set to $T = 25$ °C. The bar is compressed until a 35% reduction in height is achieved.
 738 The final simulation state is accomplished in 7-time steps with a constant time increment
 739 $\Delta t = 0.5$ s. One can observe the complexity of the nonuniform deformation presented by
 740 the barreling of the free surface (Figure 6). In our calculations, we employ a conventional
 741 rate-dependent power law to describe the material flow stress equation

$$742 \quad \bar{\sigma}(\dot{\varepsilon}) = 1000\dot{\varepsilon}^{0.1} \quad (\text{MPa}) \quad (40)$$

743 The solution convergence is assumed when the velocity error norm and the force error
 744 norm (34) becomes less than 10^{-6} . The type of element used is the linear isoparametric
 745 rectangular element with four-point integration. However, one point integration is used
 746 for the dilatation term, the second integral of the functional in (29). This is known as the
 747 reduced integration scheme which imposes the volume constancy averaged over the linear
 748 rectangular element. The computational grid composed of 100 elements interconnected at
 749 $N_s = 121$ nodes with 2 degrees of freedom, resulting in a global stiffness matrix of size
 750 242×242 . For the rigid-viscoplastic analysis, the limiting strain rate $\dot{\epsilon}_0$ is chosen to be
 751 0.01 and the penalty constant (or bulk modulus) K is set to 10^5 .

752 Among the various models of friction, the one proposed in [54] is adapted to model
 753 the sliding contact at the tool-workpiece interface. This model allows the variation of the
 754 tangential traction with the relative velocity at the tool-workpiece interface

$$755 \quad \mathbf{t}_f = -mk \frac{\mathbf{v}_s}{|\mathbf{v}_s|} \simeq -mk \left\{ \frac{2}{\pi} \arctan \left(\frac{|\mathbf{v}_s|}{v_0} \right) \right\} \frac{\mathbf{v}_s}{|\mathbf{v}_s|}$$

756 where \mathbf{v}_s is the relative velocity in the tangential direction between the tool and the
 757 workpiece, and v_0 is a positive constant several orders of magnitude smaller than \mathbf{v}_s ; m is
 758 the friction factor ($0 < m < 1$) and k is the material shear yield stress $k = \bar{\sigma}/\sqrt{3}$. For the
 759 compression tests considered here, the relative tangential velocity at the tool-workpiece
 760 interface at the beginning of deformation is zero. The present analysis assumes that the
 761 friction factor remains constant throughout compression. Investigations on frictional shear
 762 stress measurements over the interface between a cylindrical workpiece and a die during
 763 plastic compression are reported in [55]. The basic characteristics of algorithms used in
 764 the RVP FEM analysis are summarized in Table 1.

Basic characteristics of algorithms in RVP FEM

Type of problem	Two dimensional, plane strain, rigid viscoplastic material flow, isotropic, homogeneous
Thermomechanical problem solution	Loose coupling (staggered) - Backward Euler difference ($\theta = 1$)
Type of elements	4-node quadrilateral isoparametric elements, bilinear shape functions
Flow stress equation	Power law: $\bar{\sigma}(\dot{\bar{\epsilon}}) = c\dot{\bar{\epsilon}}^p$, c, p constants
Iteration method	Direct, BFGS with line search
Remeshing	N/A
Boundary conditions	Sliding friction on S_c

Table 1: Numerical algorithms.

765 *Remark 8* Note that during the course of the simulation we avoid remeshing of the workpiece. As
766 discussed in [56], remeshing techniques can be taken into account provided that mesh transfer
767 operations are applied to the reduced-basis.

768 *5.1 Mechanical field*

769 The first case for numerical illustration of the method considers the velocity field during the
770 simulation of the forming process using the shear friction factor m as the investigated parameter.
771 From now on, let the shear friction factor m denoted as λ for convenience with the previous
772 sections notation. For the numerical study, the following training points are selected $\lambda \in A_t =$
773 $\{0.1, 0.5, 0.9\}$. The choice made here, is to use a minimum number of sampling points equi-

774 distributed over the parametric range. The target point is set to $\tilde{\lambda} = 0.3$. See the FEM solutions
 775 for the training and target points at the final state of the computation in Figure 6.

776 For each parametric simulation, a sequence of snapshots uniformly distributed over time using
 777 an increment of $\Delta t = 0.5$ s is extracted for all nodes of the workpiece. The space-time snapshot
 778 matrices $\mathbf{S}^{(i)} \in \text{Mat}_{2N_s, N_t}(\mathbb{R})$ with $2N_s = 242$ and $N_t = 7$, corresponding to parameter values
 779 λ_i , are associated with the nodal velocity field in x and y directions.

780 For the parametric Space-Time interpolation, the snapshot matrix $\tilde{\mathbf{S}}$ of mode p corresponding
 781 to the target point $\tilde{\lambda}$ is computed via the target Algorithm 3. The target Algorithm 1 is applied
 782 to the spatial $\Phi_p^{(1)}, \dots, \Phi_p^{(N)}$ and temporal parts $\Psi_p^{(1)}, \dots, \Psi_p^{(N)}$, with reference parameter value
 783 $\lambda_{i_0} = 0.5$. In order to assess the interpolation accuracy, the snapshot matrix $\tilde{\mathbf{S}}$ is compared
 784 against the high-fidelity FEM solution by introducing the following a posteriori errors. Using the
 785 interpolated and the HF-FEM snapshot matrices $\tilde{\mathbf{S}}$ and \mathbf{S}^{FEM} , respectively, the relative L_2 -error
 786 measure is defined as

$$787 \quad e_{L_2}(\tilde{\mathbf{s}}_i) := \frac{\|\tilde{\mathbf{s}}_i - \mathbf{s}_i^{\text{FEM}}\|_{L_2}}{\|\mathbf{s}_i^{\text{FEM}}\|_{L_2}}, \quad i = 1, \dots, p \leq N_t. \quad (41)$$

788 Additionally, the relative Frobenius error norm of $\tilde{\mathbf{S}}$ and \mathbf{S}^{FEM} is defined as

$$789 \quad e_F(\tilde{\mathbf{S}}) := \|\tilde{\mathbf{S}} - \mathbf{S}^{\text{FEM}}\|_F / \|\mathbf{S}^{\text{FEM}}\|_F. \quad (42)$$

790 The eigenvalue spectrum of snapshot matrices $\mathbf{S}^{(i)}$ corresponding to training points $\lambda_i \in \Lambda_t$
 791 is exhibited in a semi-log scale in Figure 7. We can observe that the distance between the first
 792 and the last eigenvalue is from 5 up to 6 orders of magnitude. Moreover, the percentage of energy
 793 $\mathcal{E}(k) = \sum_{i=1}^k \sigma_i^2 / \sum_{i=1}^{N_t} \sigma_i^2$ captured from the POD modes is shown in Figure 8. It is evident that
 794 most of the 99.9% of the total energy is contained by the first two POD modes.

795 The relative L_2 -error norm $e_{L_2}(\tilde{\mathbf{s}}_i)$ (see (41)) between the interpolated and the HF-FEM
 796 solution for various POD modes is displayed in Figure 9. In general, the relative error for all
 797 POD modes lie within a range of 0.0175 up to 0.038. It can be observed that the interpolated ST

798 POD solution delivers good accuracy and is reliable enough to predict the velocity field for the
 799 investigated target point.

800 *Remark 9* In the case of using $p = 7$ POD modes for the temporal basis interpolation, the Grass-
 801 mannian manifold $\mathcal{G}(p, p)$ reduces to one point, so it is not relevant to use the target Algorithm 1:
 802 any new parameter value will give rise to the same matrix Ψ_{i_0} in the associated compact Stiefel
 803 manifold, corresponding to the reference point.

804 Additionally, the position vector error $e_{L_2}(\tilde{\mathbf{x}}(t)) = \|\tilde{\mathbf{x}}(t) - \mathbf{x}^{\text{FEM}}(t)\|_{L_2}$ at the nodal points is
 805 computed for $p = 2, 3, 5$ and 7 POD modes, where $\tilde{\mathbf{x}}(t)$ and $\mathbf{x}^{\text{FEM}}(t)$ denotes the position vector
 806 of the ST POD and the high-fidelity FEM solutions, respectively, at the time increments during
 807 the deformation. Figure 10 presents the local error $e_{L_2}(\tilde{\mathbf{x}}(t))$ superimposed at the final loading
 808 state $t = 0.35$ s obtained from the high-fidelity FEM solution. Different patterns of the spatial
 809 error distribution can be observed concerning the number of POD modes p . It is interesting to
 810 observe that in both cases, the maximum error is located near the upper-right location of the
 811 deforming workpiece.

812 The evolution of the deformation process can be also represented using the time-displacement
 813 histories of some selected nodes of the workpiece (Figure 11). The ST POD predictions are
 814 compared against the high-fidelity FEM counterpart solution using $p = 2$ POD modes. Again, it
 815 can be observed that the interpolated ST POD solution is accurate and reliable to predict the
 816 evolution of the displacement field for the investigated target point during the forming process.

817 For the preceding numerical investigations, the ST POD efficiency is demonstrated using a
 818 single target point, i.e., $\tilde{\lambda} = 0.3$. To further assess the interpolation performance, a new target
 819 point is now considered, $\tilde{\lambda} = 0.8$. Interpolation is performed using the same set of training points
 820 $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$, with reference parameter value $\lambda_{i_0} = 0.5$. The relative L_2 -error norm
 821 $e_{L_2}(\tilde{\mathbf{s}}_i)$ for various POD modes p corresponding to target point $\tilde{\lambda} = 0.8$ is shown in Figure 12.

822 Again, one can observe that the relative error lies within a narrow range of the values, i.e., 0.014
 823 up to 0.026.

824 5.2 Temperature field

825 To further investigate the performance of the proposed ST POD interpolation, the temperature
 826 field obtained from the coupled thermomechanical simulation of the forming process is considered.
 827 Again, for the temperature field, we consider the shear friction factor m as the investigated system
 828 parameter. The training points selected for the mechanical field analysis are also used in this study,
 829 i.e., $\lambda \in A_t = \{0.1, 0.5, 0.9\}$. The target point is set to $\tilde{\lambda} = 0.3$. For each parametric problem,
 830 snapshots are uniformly distributed over time using an increment step size $\Delta t = 0.5$ s. The final
 831 deformation state is reached at $t = 0.35$ s. The space-time snapshot matrices $\mathbf{S}^{(i)} \in \text{Mat}_{N_s, N_t}(\mathbb{R})$
 832 of size 121×7 , corresponding to λ_i , are associated with nodal temperatures. We will now compare
 833 the Space-Time interpolation (see Algorithm 3) against the high-fidelity FEM solution. Again, for
 834 the target Algorithm 1 applied to the spatial $\Phi_p^{(1)}, \dots, \Phi_p^{(N)}$ and temporal parts $\Psi_p^{(1)}, \dots, \Psi_p^{(N)}$,
 835 the reference parameter value $\lambda_{i_0} = 0.5$ is used.

836 Figure 13 presents the temperature profiles at the final compression state obtained using
 837 different values of the shear friction factor m (represented by parameter λ). The temperature
 838 rises due to plastic work conversion to heat assuming a constant value for the Taylor-Quinney
 839 coefficient $\xi = 0.9$. In all cases, the maximum temperature is located at the center of the workpiece
 840 with values ranging from $T = 89.5$ °C up to $T = 98$ °C.

841 The eigenvalue spectrum of snapshot matrices $\mathbf{S}^{(i)}$ corresponding to training points $\lambda_i \in A_t$
 842 is shown in a semi-log scale in Figure 14. We can observe that the distance between the first and
 843 the last eigenvalue of the curves is of the order of 5 up to 6 orders of magnitude. Moreover, the
 844 system energy $\mathcal{E}(k) = \sum_{i=1}^k \sigma_i^2 / \sum_{i=1}^{N_t} \sigma_i^2$ captured from the POD modes is shown in Figure 15.
 845 Most of the 99.9% of the total energy is contained by the first two POD modes.

846 The relative L_2 -error norm $e_{L_2}(\tilde{\mathbf{s}}_i)$ (41) between the interpolated and the HF-FEM snapshot
 847 matrices $\tilde{\mathbf{S}}$ and \mathbf{S}^{FEM} , respectively, for various modes p is shown in Figure 16. Additionally, the
 848 Frobenius relative error norm (42) for the POD modes is presented in Figure 17. In general, the
 849 obtained results are found to have less than 1% relative error for POD modes $p > 1$ and therefore
 850 are acceptable as fast near real-time numerical predictions.

851 Finally, Figure 18 shows the ST POD time-temperature histories for some selected nodes
 852 of the workpiece using $p = 7$ modes. The predictions are compared against the high-fidelity
 853 counterpart solution, and it is difficult to distinguish differences among these plots. It is revealed
 854 that the interpolated ST POD solution delivers good accuracy for all selected nodes.

855 5.3 Computational complexity

856 The computational cost of the ST POD interpolation scales with the computational complexity
 857 of SVD and the matrix operations in the target ST Algorithm 3. It is evident, that the cost
 858 of ST POD interpolation will be lower compared to the standard POD Galerkin nonlinear ap-
 859 proaches and even lower than the full order FEM solution. The coupled thermomechanical FEM
 860 simulation for the target point takes 35.123 seconds in wall-clock time. On the other hand, the
 861 ST interpolation for the mechanical problem using a ROM POD basis of mode $p = 4$ results in
 862 0.147 seconds in wall-clock time. The ST interpolation for the thermal problem using a ROM
 863 POD basis of mode $p = 4$ results in 0.153 seconds in wall-clock time. Therefore, the total ST
 864 interpolation takes 0.3 seconds in wall-clock time corresponding to a time speed-up of 116.96. All
 865 experiments in this section were implemented in Matlab and run on a 4th Generation Intel(R)
 866 Core(TM) i7-4600U CPU @ 2.10GHz, 8GB RAM, 250 GB SSD, Debian 9 x64.

867 **6 Conclusions**

868 A novel non-intrusive Space-Time POD basis interpolation scheme on compact Stiefel manifolds
869 is developed and applied to parametric high nonlinear metal forming problems. Apart from the
870 separate interpolation of POD spatial and temporal basis on associated Grassmannian mani-
871 folds, an interpolation function is defined on a set of parametric snapshot matrices. This function
872 results from curves, which are defined on compact Stiefel manifolds both for space and the tem-
873 poral part, and also the use of some mixed part encoded by a square matrix. This latter matrix
874 provides a link between the interpolated space and temporal basis for the construction of the
875 target ROM snapshot matrix. To prove the efficiency of the method it has been used a coupled
876 thermomechanical rigid-viscoplastic FEM formulation which is integrated into the manufactur-
877 ing industry in a variety of applications. The performed numerical investigations have considered
878 the reconstruction of the ROM snapshot matrices both of the velocity and the temperature
879 fields. Moreover, the error norms of the Space-Time POD interpolated ROM models concern-
880 ing the associated high-fidelity FEM counterpart solutions are validating the accuracy of the
881 proposed interpolation scheme. In conclusion, the overall results demonstrate the potential use
882 of the proposed ST POD interpolation scheme for near real-time parametric simulations using
883 off-line computed ROM POD databases, supporting thus manufacturing industries to accelerate
884 design-to-production timespans, and thereby reducing costs while ensuring the design of superior
885 processes.

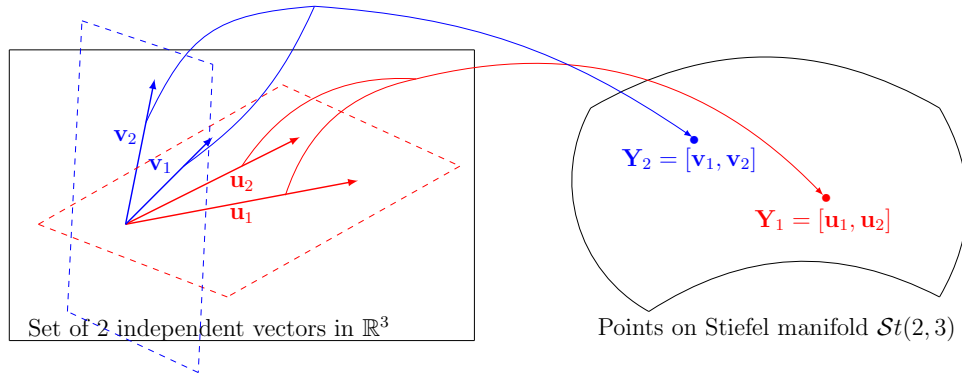


Fig. 1: Points on Stiefel manifold. The linearly independent vectors in \mathbb{R}^3 spanning the red and blue planes correspond to points in $St_c(2,3)$.

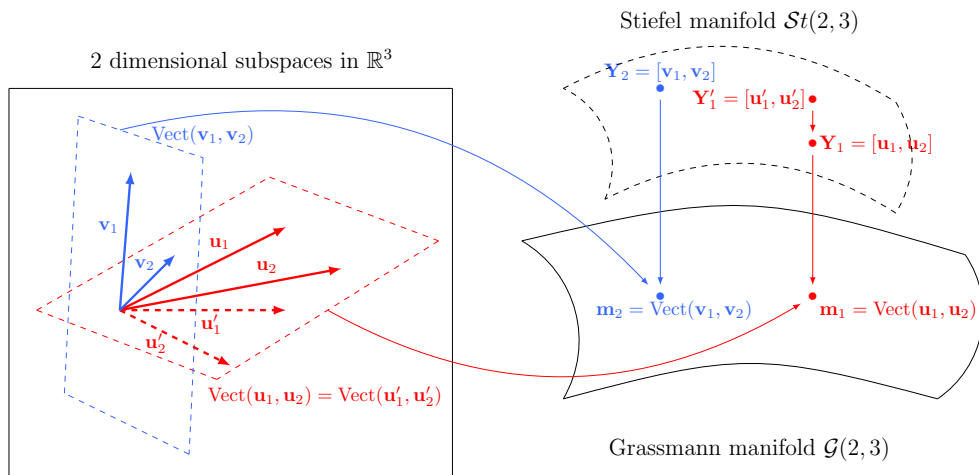


Fig. 2: Points on Stiefel $St(2,3)$ and Grassmann manifold $\mathcal{G}(2,3)$.

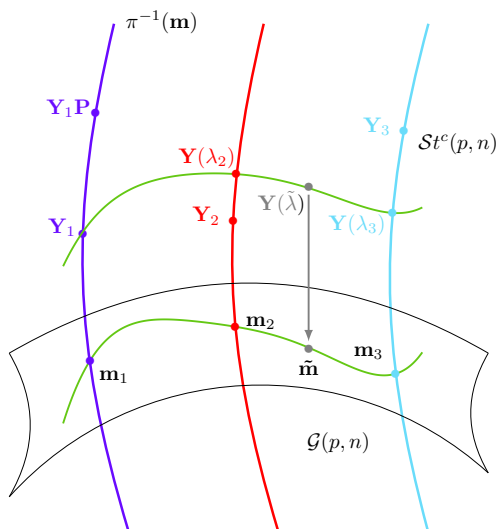


Fig. 3: There is a natural projection $\pi : St_c(p, n) \rightarrow \mathcal{G}(p, n)$ from the compact Stiefel manifold $St_c(p, n)$ to the Grassmannian $\mathcal{G}(p, n)$ of p -dimensional subspaces in \mathbb{R}^n which sends a p -frame to the subspace spanned by that frame. The fiber over a given point \mathbf{m} on $\mathcal{G}(p, n)$ is the set of all orthonormal p -frames spanning the subspace \mathbf{m} . Computations on $St_c(p, n)$ using the target Algorithm 1 for $\lambda := \lambda_k$, lead to some matrix $\mathbf{Y}(\lambda_k)$ generally different from \mathbf{Y}_k (except for the reference point), and thus do not produce an interpolation on the points $\mathbf{Y}_1, \dots, \mathbf{Y}_N$.

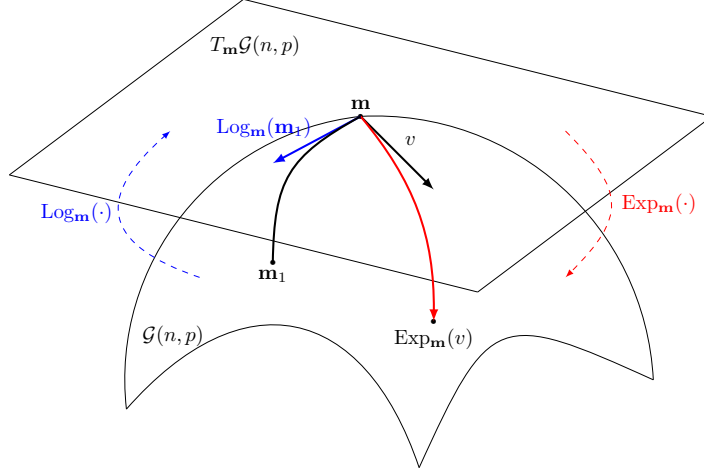


Fig. 4: The exponential Exp_m and the logarithm Log_m map on the Grassmann manifold $\mathcal{G}(p, n)$.

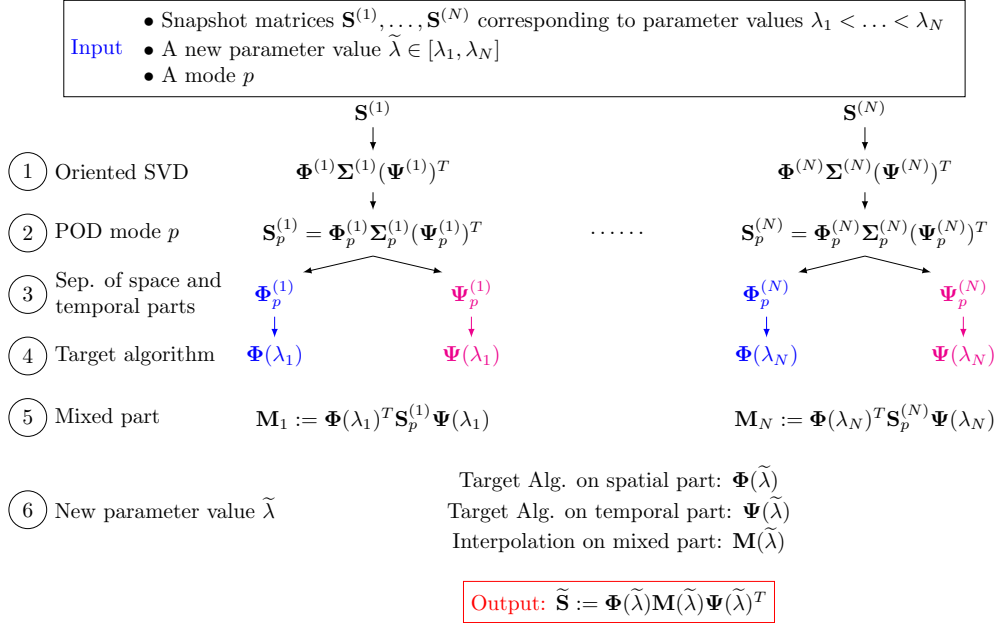


Fig. 5: The Space-Time Algorithm.

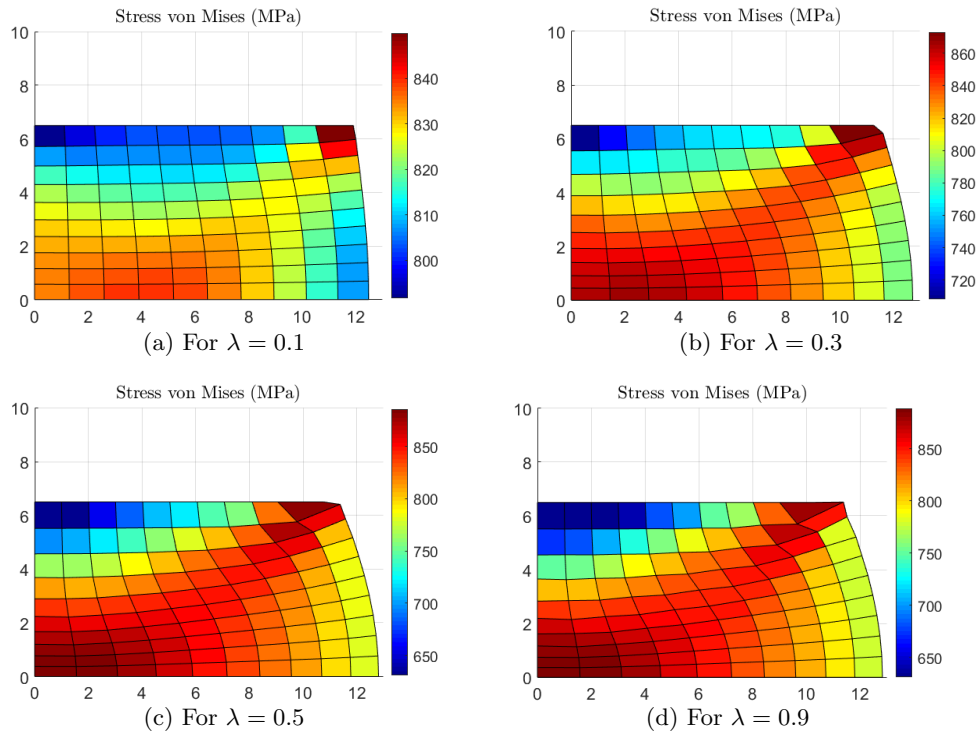


Fig. 6: Deformation patterns of the benchmark metal forming example using different values for the shear friction factor m represented by the parameter λ .

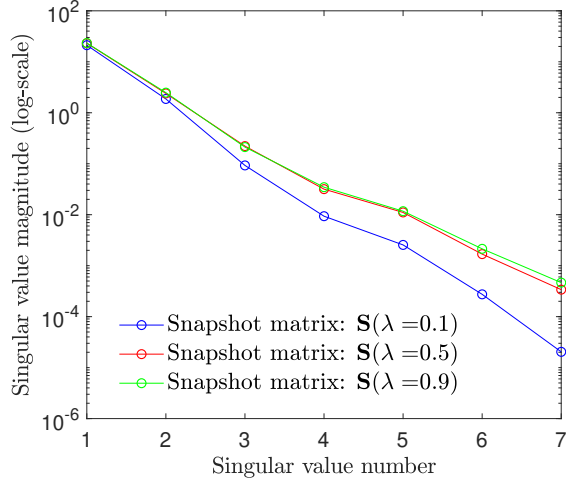


Fig. 7: The eigenvalue spectrum of snapshot matrices $\mathbf{S}^{(i)}$ corresponding to training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$.

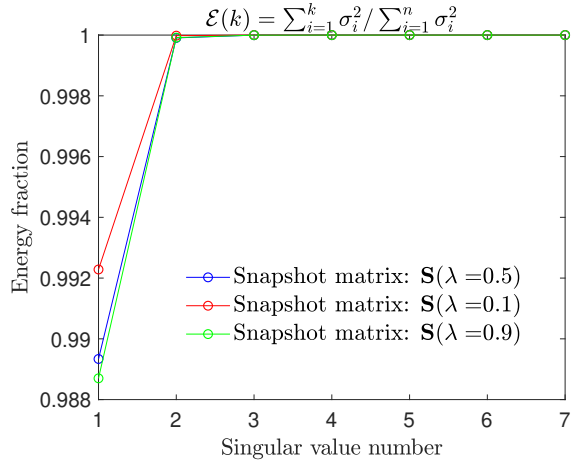


Fig. 8: Energy captured by the singular values of snapshot matrices $\mathbf{S}^{(i)}$ corresponding to training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$.

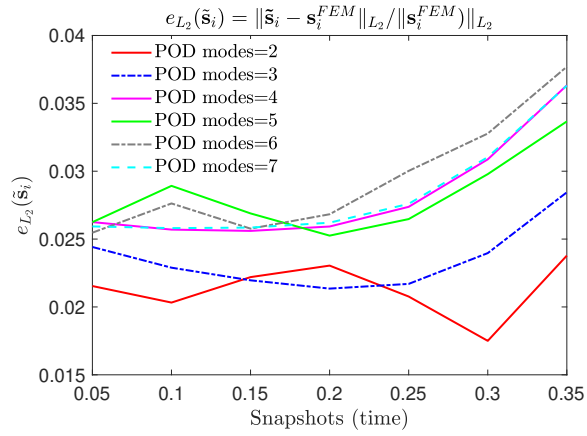


Fig. 9: Performance of ST POD interpolation using the relative L_2 -error norm $e_{L_2}(\tilde{\mathbf{s}}_i)$ for various modes p ; training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$; reference parameter value $\lambda_{i_0} = 0.5$; target point $\tilde{\lambda} = 0.3$.

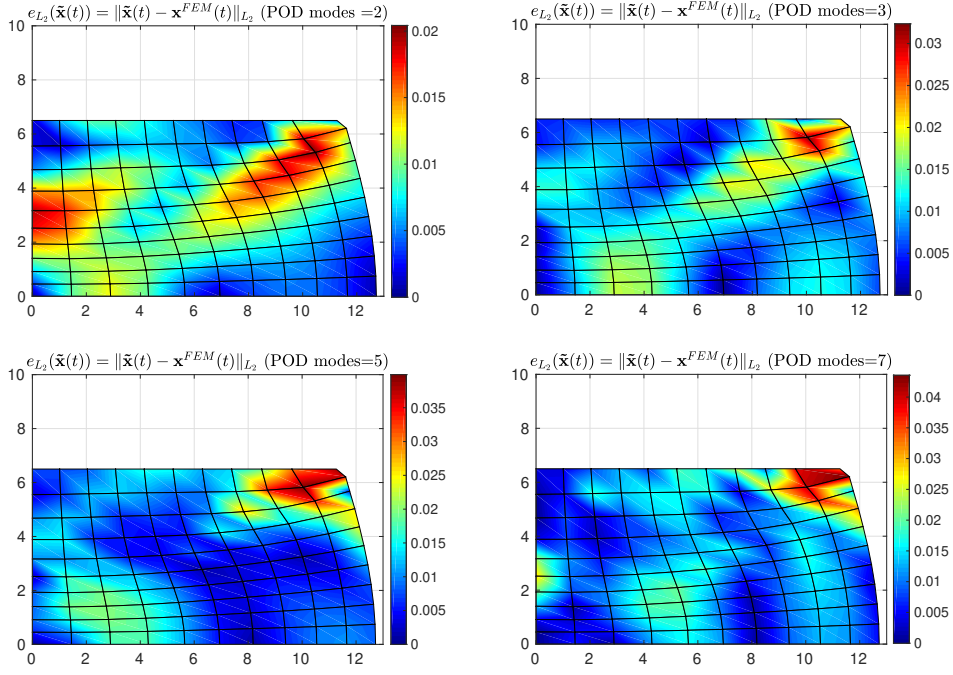


Fig. 10: The position vector error $e_{L_2}(\tilde{\mathbf{x}}(t)) = \|\tilde{\mathbf{x}}(t) - \mathbf{x}^{FEM}(t)\|_{L_2}$ of the nodal points at the final deformation state $t = 0.35$ s superimposed on the high-fidelity FEM solution; POD modes $p = \{2, 3, 5, 7\}$; training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$; reference parameter value $\lambda_{i_0} = 0.5$; target point $\tilde{\lambda} = 0.3$.

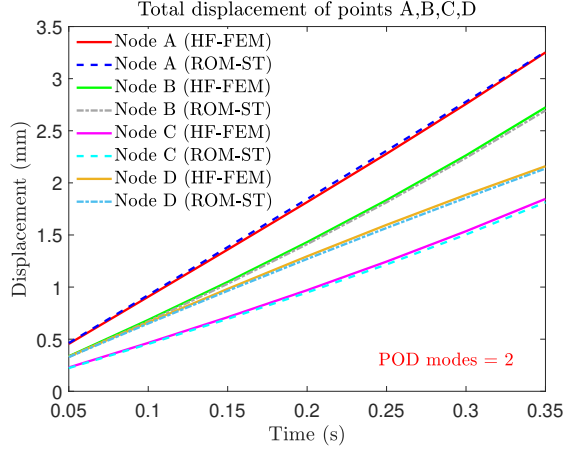


Fig. 11: Comparison of the total displacement of selected nodes against the high-fidelity FEM solution; training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$; reference parameter value $\lambda_{i_0} = 0.5$; target point $\tilde{\lambda} = 0.3$; POD modes $p = 2$.

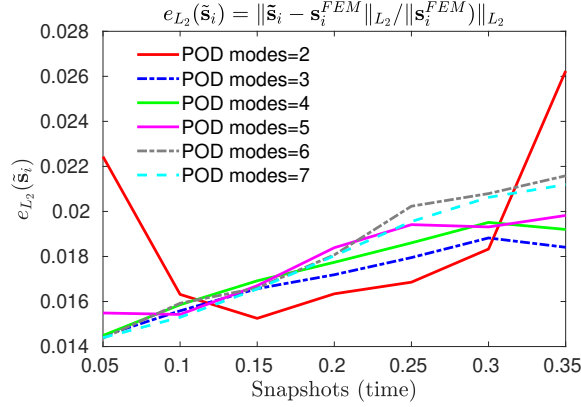


Fig. 12: Performance of ST POD interpolation using the relative L_2 -error norm $e_{L_2}(\tilde{\mathbf{s}}_i)$ for various POD modes p ; training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$; reference parameter value $\lambda_{i_0} = 0.5$; target point $\tilde{\lambda} = 0.8$.

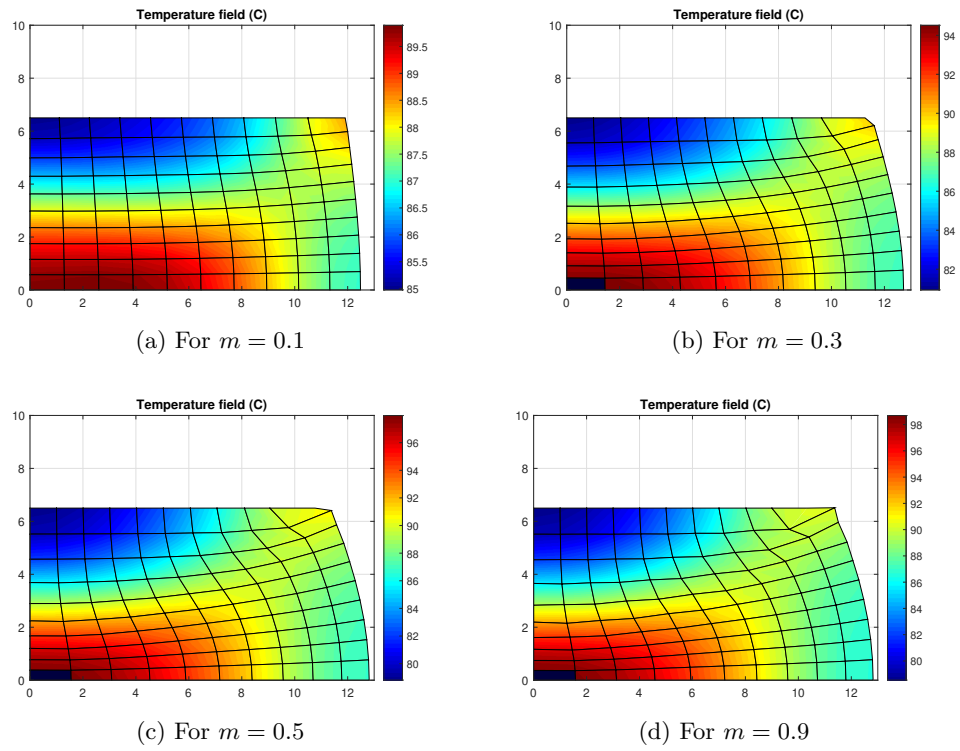


Fig. 13: Temperature profiles at the final compression state $t = 0.35$ s obtained using different values of the shear friction factor m represented by parameter λ .

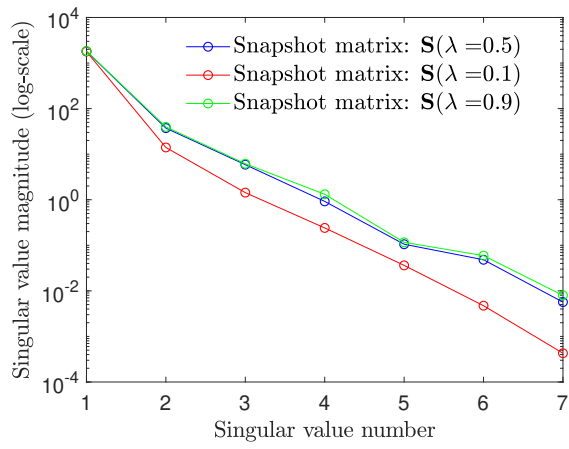


Fig. 14: The eigenvalue spectrum of snapshot matrices $\mathbf{S}^{(i)}$ corresponding to training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$.

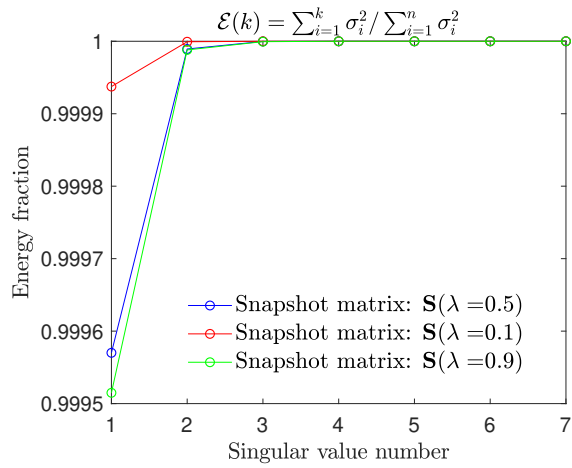


Fig. 15: Energy captured by the singular values of snapshot matrices $\mathbf{S}^{(i)}$ corresponding to training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$.

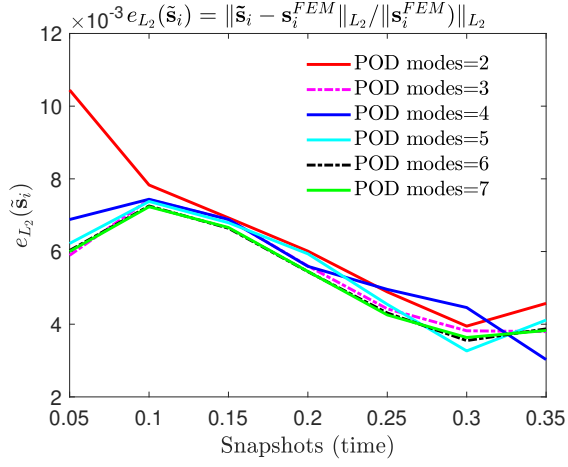


Fig. 16: Performance of ST POD interpolation using the relative L_2 -error norm $e_{L_2}(\tilde{\mathbf{s}}_i)$ for various POD modes p ; training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$; reference parameter value $\lambda_{i_0} = 0.5$; target point $\tilde{\lambda} = 0.3$.

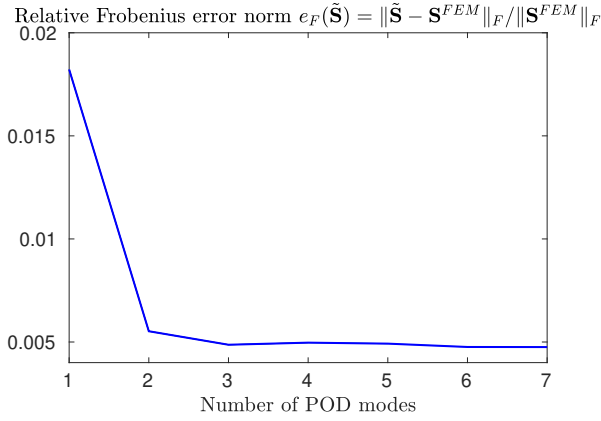


Fig. 17: Performance of the POD interpolation using the relative Frobenius error norm $e_F(\tilde{\mathbf{S}})$ against the number of POD modes p ; training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$; reference parameter value $\lambda_{i_0} = 0.5$; target point $\tilde{\lambda} = 0.3$.

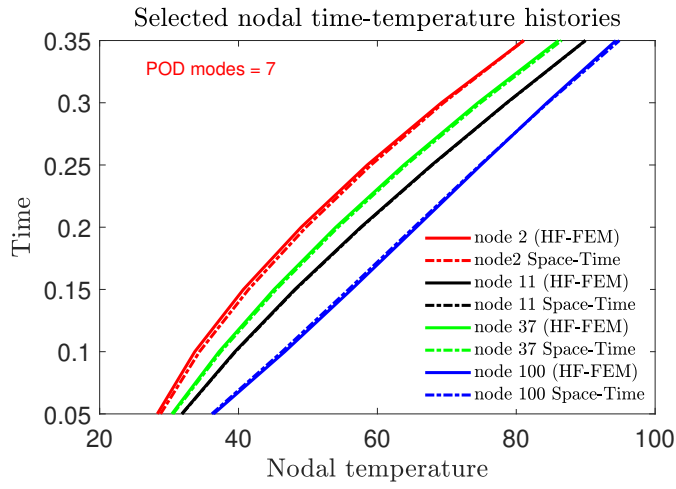


Fig. 18: Temperature evolution of selected nodal points validated against the high-fidelity FEM solution; ST POD and HF-FEM solutions virtually coincide; training points $\lambda \in \Lambda_t = \{0.1, 0.5, 0.9\}$; reference parameter value $\lambda_{i_0} = 0.5$; target point $\tilde{\lambda} = 0.3$; POD modes $p = 7$.

886 **References**

- 887 1. J.-L. Chenot. Recent contributions to the finite element modelling of metal forming processes.
888 *Journal of Materials Processing Technology*, 34(1-4):9–18, sep 1992.
- 889 2. Z. Gronostajski, Z. Pater, L. Madej, A. Gontarz, L. Lisiecki, A. Łukaszek-Sotek,
890 J. Łuksza, S. Mróz, Z. Muskalski, W. Muzykiewicz, M. Pietrzyk, R.E. Śliwa, J. Tomczak,
891 S. Wiewiórowska, G. Winiarski, J. Zasadziński, and S. Ziólkiewicz. Recent development
892 trends in metal forming. *Archives of Civil and Mechanical Engineering*, 19(3):898–941, may
893 2019.
- 894 3. Francisco Chinesta, Pierre Ladeveze, and Elías Cueto. A short review on model order re-
895 duction based on proper generalized decomposition. *Archives of Computational Methods in*
896 *Engineering*, 18(4):395–404, oct 2011.
- 897 4. C. Allery, A. Hamdouni, D. Ryckelynck, and N. Verdon. A priori reduction method for
898 solving the two-dimensional burgers’ equations. *Applied Mathematics and Computation*,
899 217(15):6671–6679, apr 2011.
- 900 5. Philip Holmes, John L Lumley, Gahl Berkooz, and Clarence W Rowley. *Turbulence, coherent*
901 *structures, dynamical systems and symmetry*. Cambridge university press, 2012.
- 902 6. Thibault Henri and Jean-Pierre Yvon. Convergence estimates of POD-galerkin methods
903 for parabolic problems. In *IFIP International Federation for Information Processing*, pages
904 295–306. Kluwer Academic Publishers, 2005.
- 905 7. Nadine Aubry. On the hidden beauty of the proper orthogonal decomposition. *Theoretical*
906 *and Computational Fluid Dynamics*, 2(5-6):339–352, aug 1991.
- 907 8. Kari Karhunen. Zur spektraltheorie stochastischer prozesse. *Ann. Acad. Sci. Fennicae, AI*,
908 34, 1946.
- 909 9. M. Loève. Elementary probability theory. In *Probability Theory I*, pages 1–52. Springer New
910 York, 1977.

- 911 10. Gene H Golub and CFV Loan. *Matrix Computations, 3rd edn., vol. 1*. JHU Press, 1996.
- 912 11. Ian T Jolliffe. Springer series in statistics. *Principal component analysis*, 29, 2002.
- 913 12. Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary*
914 *reviews: computational statistics*, 2(4):433–459, 2010.
- 915 13. J. Edward Jackson. Principal components and factor analysis: Part i principal components.
916 *Journal of Quality Technology*, 12(4):201–213, oct 1980.
- 917 14. J. Edward Jackson. Principal components and factor analysis: Part II—additional topics
918 related to principal components. *Journal of Quality Technology*, 13(1):46–58, jan 1981.
- 919 15. Patricia Astrid, Siep Weiland, Karen Willcox, and Ton Backx. Missing point estimation in
920 models described by proper orthogonal decomposition. *IEEE Transactions on Automatic*
921 *Control*, 53(10):2237–2251, nov 2008.
- 922 16. Annika Radermacher and Stefanie Reese. POD-based model reduction with empirical in-
923 terpolation applied to nonlinear elasticity. *International Journal for Numerical Methods in*
924 *Engineering*, 107(6):477–495, dec 2015.
- 925 17. Saifon Chaturantabut and Danny C. Sorensen. Nonlinear model reduction via discrete em-
926 pirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, jan 2010.
- 927 18. R. Everson and L. Sirovich. Karhunen–loève procedure for gappy data. *Journal of the Optical*
928 *Society of America A*, 12(8):1657, aug 1995.
- 929 19. Kevin Carlberg, Charbel Farhat, Julien Cortial, and David Amsallem. The GNAT method
930 for nonlinear model reduction: Effective implementation and application to computational
931 fluid dynamics and turbulent flows. *Journal of Computational Physics*, 242:623–647, jun
932 2013.
- 933 20. David Amsallem, Julien Cortial, Kevin Carlberg, and Charbel Farhat. A method for inter-
934 polating on manifolds structural dynamics reduced-order models. *International journal for*
935 *numerical methods in engineering*, 80(9):1241–1258, 2009.

- 936 21. Rolando Mosquera Meza. *Interpolation sur les variétés grassmanniennes et applications à la*
937 *réduction de modèles en mécanique*. PhD thesis, La Rochelle, 2018.
- 938 22. Silvère Bonnabel and Rodolphe Sepulchre. Riemannian metric and geometric mean for posi-
939 tive semidefinite matrices of fixed rank. *SIAM Journal on Matrix Analysis and Applications*,
940 31(3):1055–1070, jan 2010.
- 941 23. John M Lee. Smooth manifolds. In *Introduction to Smooth Manifolds*, pages 1–31. Springer,
942 2013.
- 943 24. Sylvestre Gallot, Dominique Hulin, and Jacques Lafontaine. *Riemannian geometry*, volume 2.
944 Springer, 1990.
- 945 25. Rolando Mosquera, , Aziz Hamdouni, Abdallah El Hamidi, and Cyrille Allery. POD basis
946 interpolation via inverse distance weighting on grassmann manifolds. *Discrete & Continuous*
947 *Dynamical Systems - S*, 12(6):1743–1759, 2019.
- 948 26. G. Muhlbach. The general neville-aitken-algorithm and some applications. *Numerische*
949 *Mathematik*, 31(1):97–110, mar 1978.
- 950 27. Y. Lu, N. Blal, and A. Gravouil. Space–time POD based computational vademecums for
951 parametric studies: application to thermo-mechanical problems. *Advanced Modeling and*
952 *Simulation in Engineering Sciences*, 5(1), feb 2018.
- 953 28. M. Oulghelou and C. Allery. Non intrusive method for parametric model order reduction
954 using a bi-calibrated interpolation on the grassmann manifold. *Journal of Computational*
955 *Physics*, 426:109924, feb 2021.
- 956 29. Vilas Shinde, Elisabeth Longatte, Franck Baj, Yannick Hoarau, and Marianna Braza. A
957 galerkin-free model reduction approach for the navier–stokes equations. *Journal of Compu-*
958 *tational Physics*, 309:148–163, mar 2016.
- 959 30. C. Audouze, F. De Vuyst, and P. B. Nair. Reduced-order modeling of parameterized PDEs
960 using time-space-parameter principal component analysis. *International Journal for Numer-*

- 961 *ical Methods in Engineering*, 80(8):1025–1057, nov 2009.
- 962 31. Youngsoo Choi and Kevin Carlberg. Space–time least-squares petrov–galerkin projection
963 for nonlinear model reduction. *SIAM Journal on Scientific Computing*, 41(1):A26–A58, jan
964 2019.
- 965 32. Youngsoo Choi, Peter Brown, William Arrighi, Robert Anderson, and Kevin Huynh.
966 Space–time reduced order model for large-scale linear dynamical systems with application to
967 boltzmann transport problems. *Journal of Computational Physics*, 424:109845, jan 2021.
- 968 33. Christophe Audouze, Florian De Vuyst, and Prasanth B. Nair. Nonintrusive reduced-order
969 modeling of parametrized time-dependent partial differential equations. *Numerical Methods
970 for Partial Differential Equations*, 29(5):1587–1628, feb 2013.
- 971 34. Shiro Kobayashi, Shirō Kobayashi, Soo-Ik Oh, and Taylan Altan. *Metal forming and the
972 finite-element method*, volume 4. Oxford University Press on Demand, 1989.
- 973 35. C. H. Lee and S. Kobayashi. New solutions to rigid-plastic deformation problems using a
974 matrix method. *Journal of Engineering for Industry*, 95(3):865–873, aug 1973.
- 975 36. Shiro Kobayashi. Rigid-plastic finite element analysis of axisymmetric metal forming pro-
976 cesses. *Numerical Modeling of Manuf. Process (ASME, New York, 1977)*, pages 49–65, 1977.
- 977 37. ZQ Feng and G De Saxcé. Rigid-plastic implicit integration scheme for analysis of metal
978 forming. *European Journal of Mechanics, A/Solids*, 15(1):51–66, 1996.
- 979 38. O. Friderikos. Two-dimensional rigid-plastic fem simulation of metal forming processes in
980 matlab. Proceedings of the 4th International Conference on Manufacturing and Materials
981 Engineering (ICMMEN), 3-5 October, Thessaloniki, Greece, 2011.
- 982 39. Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with
983 orthogonality constraints. *SIAM journal on Matrix Analysis and Applications*, 20(2):303–
984 353, 1998.

- 985 40. P.-A. Absil, R. Mahony, and R. Sepulchre. Riemannian geometry of grassmann manifolds
986 with a view on algorithmic computation. *Acta Applicandae Mathematicae*, 80(2):199–220,
987 jan 2004.
- 988 41. S. E. Kozlov. Geometry of the real grassmannian manifolds. parts i, ii. *Zapiski Nauchnykh*
989 *Seminarov POMI*, 246:84–107, 1997.
- 990 42. Yung-Chow Wong. Differential geometry of grassmann manifolds. *Proceedings of the National*
991 *Academy of Sciences of the United States of America*, 57(3):589, 1967.
- 992 43. S. Berceanu. On the geometry of complex grassmann manifold, its noncompact dual and
993 coherent states. *Bulletin of the Belgian Mathematical Society - Simon Stevin*, 4(2):205–243,
994 1997.
- 995 44. Rodney Hill. *The mathematical theory of plasticity*, volume 11. Oxford university press, 1998.
- 996 45. A.A. Markov. *On variational principles in the theory of plasticity*. Division of Applied
997 Mathematics, Brown University, 1948.
- 998 46. R Hill. A variational principle of maximum plastic work in classical plasticity. *The Quarterly*
999 *Journal of Mechanics and Applied Mathematics*, 1(1):18–28, 1948.
- 1000 47. S.I. Oh. Finite element analysis of metal forming processes with arbitrarily shaped dies.
1001 *International Journal of Mechanical Sciences*, 24(8):479–493, jan 1982.
- 1002 48. O. C. Zienkiewicz and P. N. Godbole. Flow of plastic and visco-plastic solids with special
1003 reference to extrusion and forming processes. *International Journal for Numerical Methods*
1004 *in Engineering*, 8(1):1–16, 1974.
- 1005 49. Anthony Ralston and Philip Rabinowitz. *A first course in numerical analysis*. Courier
1006 Corporation, 2001.
- 1007 50. Germund Dahlquist and Åke Björck. *Numerical Methods in Scientific Computing, Volume I*.
1008 Society for Industrial and Applied Mathematics, jan 2008.

- 1009 51. C.A. Felippa and K.C. Park. Staggered transient analysis procedures for coupled mechanical
1010 systems: Formulation. *Computer Methods in Applied Mechanics and Engineering*, 24(1):61–
1011 111, oct 1980.
- 1012 52. N. Rebelo and S. Kobayashi. A coupled analysis of viscoplastic deformation and heat trans-
1013 fer—i. *International Journal of Mechanical Sciences*, 22(11):699–705, jan 1980.
- 1014 53. N. Rebelo and S. Kobayashi. A coupled analysis of viscoplastic deformation and heat trans-
1015 fer—II. *International Journal of Mechanical Sciences*, 22(11):707–718, jan 1980.
- 1016 54. CC Chen. Rigid-plastic finite-element analysis of ring compression. *Applications of numerical*
1017 *methods of forming processes*, 1978.
- 1018 55. G.T. van Rooyen and W.A. Backofen. A study of interface friction in plastic compression.
1019 *International Journal of Mechanical Sciences*, 1(1):1–27, jan 1960.
- 1020 56. D. Ryckelynck. Hyper-reduction of mechanical models involving internal variables. *Interna-*
1021 *tional Journal for Numerical Methods in Engineering*, 77(1):75–89, jan 2009.