Finite strain data-driven computational mechanics. : From tailored data to adaptive solvers for multiscale simulations

Auriane Platzer

THESE DE DOCTORAT DE

L'ÉCOLE CENTRALE DE NANTES

ECOLE DOCTORALE N° 602
Sciences pour l'Ingénieur
Spécialité : Mécanique des Solides, des Matériaux, des Structures et des Surfaces.

Par
Auriane Platzer

Mécanique numérique en grandes transformations pilotée par les données

De la génération de données sur mesure à une stratégie adaptative de calcul multiéchelle

Thèse présentée et soutenue à l'Ecole Centrale de Nantes, le 15 décembre 2020
Unité de recherche : UMR 6183, Institut de Recherche en Génie Civil et Mécanique (GeM)

Rapporteurs avant soutenance :

Julien Yvonnet	Professeur des universités, Université Gustave Eiffel, Marne-la-Vallée
Marc-André Keip	Professeur, Université de Stuttgart (Allemagne)

Composition du Jury :

Président :	David Ryckelynck	Professeur, Mines Paris Tech
Examinateurs :	Stefanie Reese	Professeure, RWTH Aachen Université (Allemagne)
	Piotr Breitkopf	Ingénieur de recherche CNRS, Université Technologique de Compiègne
	Michael Ortiz	Professeur, California Institute of Technology (États-Unis)
Dir. de thèse :	Laurent Stainier	Professeur des universités, Ecole Centrale de Nantes
Co-encadrant :	Adrien Leygue	Chargé de recherche CNRS, Ecole Centrale de Nantes

P H D T H E S I S

to obtain the title of

PhD in Engineering Sciences

of the École Centrale de Nantes

defended by

Auriane Platzer

Finite strain data-driven computational mechanics

From tailored data to adaptive solvers for multiscale simulations

À ma mère.

Remerciements

Mes premiers remerciements vont aux rapporteurs de cette thèse, Julien Yvonnet et Marc-André Keip. On dit souvent que les thèses ne sont jamais vraiment lues, si ce n'est par leurs rapporteurs·rices. Je mesure le privilège d'avoir eu mes travaux rapportés avec rigueur par ces deux scientifiques et les remercie chaleureusement pour leur travail. Je remercie également tous les autres membres du jury pour leurs questions, remarques et leur intérêt enthousiaste pour le sujet : merci à David Ryckelynck d'avoir accepté de présider ce jury, à Stefanie Reese pour sa rigueur bienveillante, à Piotr Breitkopf pour son accompagnement et ses encouragements, et à Michael Ortiz pour son suivi parfois taquin mais toujours chaleureux et sa vivacité scientifique.

Cette thèse n'aurait évidemment pas pu aboutir sans la confiance et la maturité scientifique de mes encadrants Laurent Stainier et Adrien Leygue. Merci de m'avoir transmis sans réserve votre savoir et votre passion pour la Science. Laurent, merci également pour m'avoir donné l'opportunité d'enseigner pendant ma thèse, ce qui a rendu cette aventure d'autant plus enrichissante. Merci Adrien de m'avoir accompagnée « depuis que je suis toute petite » et de ton amitié, précieuse.

Il est également des professeurs, des mentors, qui ne sont pas innocents dans mon parcours : merci à Erwan Verron et Bertrand Huneau, duo de choc dont le prosélytisme doctoral n'a d'égal que leur passion pour les boissons houblonnées ; merci à Michel Coret d'avoir tenté de m'initier à l'activité expérimentale avec autant d'affection que d'échecs de ma part ; merci à Christophe Binetruy et Sébastien Comas-Cardona dont j'apprécie et connais l'intégrité scientifique et l'humanité au travers des autres doctorant·es qu'ils ont encadré·es. Vous maintenez contre vents et marrées une atmosphère chaleureuse et engagée dans laquelle l'ensemble des doctorant·es peut s'épanouir. Merci à elles et eux pour m'avoir accueillie et encouragée, pour les parties de coinche endiablées et pour tout ce que vous avez partagé avec moi. Je ne vous cite pas nommément mais si vous « habitez » (ou avez habité) dans le bâtiment T ou F, recevez ces remerciements. Enfin, merci à tous les membres d'Unité Centrale, magnifique exemple d'intelligence collective, et aux copaines qui en font partie.

Pour terminer, je veux exprimer toute ma gratitude à mes ami·es pour leur soutien et leur amour indéfectibles, avec une mention particulière à Lola, qui est toujours là depuis si longtemps et à Anceline, Ella, Priscillia et Télie, dont l'amitié m'honore. Nous sommes aussi et avant tout le produit d'une famille et je remercie la mienne dont le soutien affectif et matériel m'a offert le luxe de faire une thèse. Merci

à Florestan d'être le grand frère sans qui je ne me serais certainement pas lancée là-dedans, ni dans grand chose d'autre d'ailleurs. Merci au relecteur de l'ombre pour sa rigueur, son estime et bien plus encore. Ma dernière pensée s'adresse à deux femmes d'exception qui manquent au monde, ma grand-mère Nicole Platzer et ma mère Valérie Mamoux, dont j'espère honorer la détermination et l'immense humanité par ce travail, et dans le futur.

Contents

List of Acronyms

General introduction

General introduction

Computational mechanics is a field in which a large amount of *data* is both consumed and produced. On the one hand, the recent developments in experimental measurement techniques have provided rich databases for the identification process of constitutive models used in finite element simulations. In that case, data are usually discarded once the parameters are identified. On the other hand, heavy computations, such as multiscale analyses, produce a huge amount of discrete values of displacements, strains, stresses, or strain energy densities. Again, if databases resulting from computational micromechanics can be stored, they are often used to train a meta-model, and only this model is subsequently evaluated in macroscopic simulations, without any further reference to the original microscopic data. The constitutive model, understood in the broad sense as the relationship (not necessarily explicit) between the constitutive variables (typically strain and stress), then acts as a bottleneck between upstream and downstream material data.

In contrast, Kirchdoerfer and Ortiz (2016) recently introduced a model-free computing paradigm, so called Data-Driven Computational Mechanics (DDCM), which incorporates data *directly* into the computation, thus bypassing the need for a constitutive model. This seminal work was consolidated by the mathematical framework proposed in Conti et al. (2018), demonstrating that DDCM encompasses the classical definition of the solid mechanics Boundary Value Problem (BVP).

The first part of this thesis (Part A) is dedicated to recalling the motivation and principles of the approach. In short, the approach relies on the fundamental separation between the characterization of the material response and the satisfaction of essential constraints and conservation laws: while the former is most likely to be approximately captured (*e.g.* due to noise in experimental data), the latter ones must be exactly satisfied. In solid mechanics, the typical constitutive variables involved are the strain and the stress. The BVP is therefore reformulated as a minimization of a distance function between two types of strain-stress pair fields: one, representing the material response, takes value in a database resulting from experimental or numerical acquisition, and the other, verifying both compatibility and equilibrium equations, represents the admissible mechanical state of the body. The former strain-stress field is referred to as the *material state* and belongs to a so-called *material data set*, while the latter *mechanical state* belongs to a so-called *constraint set*. Both sets are subspaces of the so-called *phase space*, which collects all possible strain-stress pair fields.

In this model-free approach, the material response is only represented by a discrete set of material data points: no interpolation, approximation nor modeling of the data set is ever performed. Likewise, the material response can not be extrapolated from missing data points, contrary to the constitutive model which is often used well beyond its domain of identification. Ideally, the material database should then comprise an almost infinite number of points, covering every possible mechanical state, in order to replace a constitutive model. However, the number

of material data points that can be acquired either experimentally or through numerical simulations is obviously limited. Hence, the arrangement and coverage of material data points in phase space are critical for the prediction of the mechanical response. In particular, the use of raw material data raises crucial questions such as "richness" or "density" of the database. The former can be understood as the adequacy of the database to a large variety of mechanical sollicitations. The latter refers not only to the size of the database (number of points) but also to notions of clustering or sparsity of material data points in certain regions: *e.g.*, should data points be evenly spaced or clustered in relevant areas of phase space? In addition, the curse of dimensionality can prevent the database from spanning the whole phase space: using symmetric strain-stress tensor pairs, the dimension of the phase space equals the number of independent components in each tensor, *i.e.* $6 + 6 = 12$ for three-dimensional problems. Yet, it may not be necessary for the material database to span the whole phase space; instead it may preferably span an appropriate region of it. Our work then addresses the following questions:

- In what manner should material data points be distributed in phase space to accurately predict the mechanical response of a structure?
- How can we control this distribution to generate tailored databases?

The present thesis investigates these questions from two points of view, which are each presented in a part of the manuscript:

- the extension of DDCM to large strain (Part B),
- the application of DDCM to multiscale simulation of heterogeneous materials (Part C).

In Part B, data coverage is investigated in the context of finite strain elasticity. In particular, the choice of work conjugate strain-stress pairs has a direct impact on the dimensionality of the phase space and on the formulation of the problem. Here, we explore two possibilities: (i) the Lagrangian formulation of Nguyen and Keip (2018), and (ii) the nominal formulation of Conti et al. (2018). In the former, the work conjugate strain-stress pairs are the Green-Lagrange strain–second Piola-Kirchhoff stress symmetric tensors, while in the latter, the deformation gradient–first Piola-Kirchhoff stress non-symmetric tensors are favored. For both formulations, we develop a finite element solver, which is an original contribution in the case of the nominal formulation. In this context, the material database is synthetic: we generate it from sampling a known constitutive model. This method has several advantages:

- it allows for assessing the developed data-driven solvers by comparing the results with classical finite element analysis, taken as a reference solution (Chapter III);

- it allows for (almost) unlimited exploration of material databases, in particular to evaluate the necessary mechanical content and spread in phase space (Chapter IV);
- it allows for a controlled generation of a database suitable for a genuinely three-dimensional computation, which is beyond the current capacity of experimental acquisition techniques (Chapter V).

The last part of our work (Part C) focuses on applying DDCM to multiscale simulation of heterogeneous materials. For this application, the databases are no longer synthetic but generated from computational micromechanics. The material database is constituted with macroscopic strain-stress pairs, obtained from the volume average of the microscopic fields in a Representative Volume Element of the microstructure. It is established that the number of microscopic evaluations is usually a critical aspect in multiscale schemes such as multilevel finite element approach (FE^2), popularized by Feyel (1999). Here, we propose to use data-driven solvers as a tool to accelerate FE^2 simulations. As mentioned by Kirchdoerfer and Ortiz (2016), as distance-minimizing schemes, data-driven solvers indeed provide "error measures which highlight data regions that require additional resolution". We use this specific feature of DDCM to select the relevant microscopic evaluations to perform, based on the adequacy of the material database to the current macroscopic mechanical response. We then develop, in the last chapter of this thesis (Chapter VI) a first adaptive strategy for multiscale simulations, restricted to small strain (nonlinear) elasticity.

Part A

Computational mechanics and data-driven approaches

Chapter I

State of the art

Contents

1 Basics of nonlinear computational mechanics

Nonlinear computational mechanics refers to the "study of the numerical analysis of nonlinear continua using a computer, [which] comprises nonlinear continuum mechanics together with the numerical schemes for solving the resulting governing equations" and is recognized as "the only route by which the behavior of a complex component subject to complex loading can be successfully simulated" (Bonet and Wood, 2008). Basics ingredients of (nonlinear) continuum mechanics are kinematics, stress and equilibrium, and constitutive behavior. In this section, we revisit the governing equations of a Boundary Value Problem (BVP) in (nonlinear) continuum mechanics and their discretization by the Finite Element (FE) method, to serve as a basis to the definitions and notations used throughout this thesis. The following is widely inspired by the textbook of Bonet and Wood (2008) and we refer the interested reader to it for more details on the derivations.

Notations and conventions Throughout the text, Einstein's summation convention is used with indices ranging from 1 to 3, unless otherwise indicated:

$$A_{ik}B_{kj} = \sum_{k=1}^{3} A_{ik}B_{kj}. \tag{I.1}$$

Considering vectors $\boldsymbol{u}$, $\boldsymbol{v}$, $\boldsymbol{w} \in \mathbb{R}^3$, second-order tensors $\boldsymbol{A}$, $\boldsymbol{B} \in \mathbb{R}^{3\times3}$, a third-order tensor $\boldsymbol{\mathcal{D}} \in \mathbb{R}^{3\times3\times3}$, and a fourth-order tensor $\mathbb{H} \in \mathbb{R}^{3\times3\times3\times3}$, the standard inner and outer products are expressed as:

$$\boldsymbol{u}\cdot\boldsymbol{v} = u_i v_j; \qquad (\boldsymbol{u}\otimes\boldsymbol{v})\,\boldsymbol{w} = (\boldsymbol{w}\cdot\boldsymbol{v})\,\boldsymbol{u}; \tag{I.2}$$

$$\boldsymbol{A}\boldsymbol{u} = A_{ij}u_j\,\boldsymbol{e}_i; \qquad \boldsymbol{u}\cdot\boldsymbol{A} = u_i A_{ij}\,\boldsymbol{e}_j; \tag{I.3}$$

$$\boldsymbol{A}\boldsymbol{B} = A_{ik}B_{kj}\,(\boldsymbol{e}_i\otimes\boldsymbol{e}_j); \qquad \boldsymbol{A}:\boldsymbol{B} = \mathrm{tr}\left(\boldsymbol{A}\boldsymbol{B}^T\right) = A_{ij}B_{ij}; \tag{I.4}$$

$$\boldsymbol{\mathcal{D}}:(\boldsymbol{u}\otimes\boldsymbol{v}) = (\boldsymbol{\mathcal{D}}\boldsymbol{v})\boldsymbol{u}; \qquad \boldsymbol{\mathcal{D}}:\boldsymbol{B} = D_{ijk}B_{jk}\,\boldsymbol{e}_i; \tag{I.5}$$

$$\mathbb{H}:\boldsymbol{A} = H_{ijkl}A_{kl}\,(\boldsymbol{e}_i\otimes\boldsymbol{e}_j); \tag{I.6}$$

with $(\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3)$ the canonical basis of $\mathbb{R}^3$.

The second- and symmetric fourth-order tensors are respectively written as

$$\boldsymbol{I} = \delta_{ij}\,(\boldsymbol{e}_i\otimes\boldsymbol{e}_j); \qquad \mathbb{I}^{\mathrm{SYM}} = \frac{1}{2}\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right)(\boldsymbol{e}_i\otimes\boldsymbol{e}_j\otimes\boldsymbol{e}_k\otimes\boldsymbol{e}_l), \tag{I.7}$$

with δ_{ij} the Kronecker's symbol: $\delta_{ij} = 1$ if $i = j$, $\delta_{ij} = 0$ otherwise.

1.1 Governing equations of nonlinear continuum mechanics

1.1.1 Motion and kinematics

The general motion of a deformable body is described by the transformation mapping ϕ that relates the initial coordinate $\boldsymbol{X}$ (at time $t = 0$) to the current coordinate $\boldsymbol{x}$ (at time t) as

$$\boldsymbol{x} = \boldsymbol{\phi}(\boldsymbol{X}, t), \tag{I.8}$$

and the displacement field is defined by

$$\boldsymbol{u} = \boldsymbol{x} - \boldsymbol{X}. \tag{I.9}$$

In infinitesimal deformation theory, the displacement $\boldsymbol{u}$ is assumed to be small in comparison with the dimensions of the body. The deformation of the body is then entirely measured by the linearized "small strain" tensor

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(\boldsymbol{\nabla}\boldsymbol{u} + \boldsymbol{\nabla}^T\boldsymbol{u}\right). \tag{I.10}$$

where $\boldsymbol{\nabla}\bullet = \partial \bullet / \partial \boldsymbol{x}$ denotes the gradient with respect to the current coordinates. In finite strain, the magnitude of the displacement field can be of the order or even exceed the initial dimensions of the body. The small strain $\boldsymbol{\varepsilon}$ does not account for all geometrical changes. More general strain (and stress) measures must therefore be defined. Kinematic and stress descriptions can either be referenced to the initial state of the body (classically undeformed and referred to as the reference state) or to its deformed state; the former is referred to as the *material* or *Lagrangian* description whereas the latter is referred to as the *spatial* or *Eulerian* description. Regardless of the chosen setting, all strain measures rely on the deformation gradient tensor $\boldsymbol{F}$:

$$\boldsymbol{F} = \boldsymbol{\nabla}_0\boldsymbol{\phi} = \boldsymbol{I} + \boldsymbol{\nabla}_0\boldsymbol{u}, \tag{I.11}$$

where $\boldsymbol{\nabla}_0\bullet = \partial \bullet / \partial \boldsymbol{X}$ denotes the gradient with respect to the initial coordinates. The deformation gradient tensor transforms infinitesimal lengths, surface areas and volumes. In particular, it is worth mentioning that the Jacobian J measures the volume change during deformation and hence, is positive:

$$J = \det \boldsymbol{F} > 0. \tag{I.12}$$

Strain tensors measure the change in the scalar product of two elemental vectors, from the initial configuration to the current configuration, and are expressed either with reference to material or spatial coordinates, leading to the following quantities:

- Lagrangian strain tensor: the Green-Lagrange strain tensor $\boldsymbol{E}$ is defined thanks to the right Cauchy-Green deformation tensor $\boldsymbol{C}$ as

$$\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F}, \tag{I.13}$$

$$\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{C} - \boldsymbol{I}\right). \tag{I.14}$$

- Eulerian strain tensor: the Euler-Almansi strain tensor $\boldsymbol{e}$ is defined thanks to the left Cauchy-Green deformation tensor $\boldsymbol{b}$ as

$$\boldsymbol{b} = \boldsymbol{F}\boldsymbol{F}^T, \tag{I.15}$$

$$\boldsymbol{e} = \frac{1}{2}\left(\boldsymbol{I} - \boldsymbol{b}^{-1}\right). \tag{I.16}$$

Throughout the text, we refer to the above displacement-strain relationships as *compatibility* equations. Note that, classically, compatibility equations refer to the mathematical conditions that $\boldsymbol{\varepsilon}$ (resp. $\boldsymbol{F}$) must satisfy to match a small strain (resp. deformation gradient) tensor. Practically, $\boldsymbol{\varepsilon}$ and $\boldsymbol{F}$ must be integrable, and there must exist a displacement field from which they come from[1].

Concurrently, the relative change of velocity $\boldsymbol{v}(\boldsymbol{x},t) = \partial\boldsymbol{x}/\partial t$ between two neighboring particles in the spatial configuration is measured by the velocity gradient tensor $\boldsymbol{l}$ as

$$\boldsymbol{l} = \boldsymbol{\nabla}\boldsymbol{v}. \tag{I.17}$$

From the spatial point of view, the rate of deformation tensor $\boldsymbol{d}$ is defined as the symmetric part of $\boldsymbol{l}$:

$$\boldsymbol{d} = \frac{1}{2}\left(\boldsymbol{l} + \boldsymbol{l}^T\right), \tag{I.18}$$

the antisymmetric part being the spin tensor $\boldsymbol{w} = 1/2\left(\boldsymbol{l} - \boldsymbol{l}^T\right)$. In addition, the time derivative of the deformation gradient writes

$$\dot{\boldsymbol{F}} =_{\text{def}} \frac{\partial \boldsymbol{F}}{\partial t} = \frac{\partial \boldsymbol{v}}{\partial \boldsymbol{X}} \tag{I.19}$$

and allows for the definition of the material strain rate tensor $\dot{\boldsymbol{E}}$ to write as:

$$\dot{\boldsymbol{E}} = \frac{1}{2}\left(\dot{\boldsymbol{F}}^T\boldsymbol{F} + \boldsymbol{F}^T\dot{\boldsymbol{F}}\right). \tag{I.20}$$

Both the material and spatial strain rates are related through:

$$\boldsymbol{d} = \boldsymbol{F}^{-T}\dot{\boldsymbol{E}}\boldsymbol{F}^{-1} \;\Leftrightarrow\; \dot{\boldsymbol{E}} = \boldsymbol{F}^T\boldsymbol{d}\boldsymbol{F}. \tag{I.21}$$

1.1.2 Stress and equilibrium

In finite strain setting, the different stress measures can be introduced in several ways. Here, we follow the same organization as in Bonet and Wood (2008): first, we define the Cauchy stress tensor $\boldsymbol{\sigma}$ as the force per unit area in the current configuration. Second, other measures are introduced using work conjugacy by manipulating the principle of virtual work, such as the first and second Piola-Kirchhoff stress tensors $\boldsymbol{P}$ and $\boldsymbol{S}$.

[1] In small strain, the compatibility conditions write

$$\boldsymbol{\nabla} \times (\boldsymbol{\nabla} \times \boldsymbol{\varepsilon}) = \boldsymbol{0},$$

and in finite strain

$$\boldsymbol{\nabla}_0 \times \boldsymbol{F} = \boldsymbol{0}.$$

Note that compatibility conditions are necessary when $\boldsymbol{\varepsilon}$ (resp. $\boldsymbol{F}$) derives from a continuous, single-valued, displacement field as in Eq. (I.10) (resp. Eq. (I.11)).

For this purpose, let us consider the general BVP illustrated in Fig. I.1. The body occupies a domain $\Omega_0 \in \mathbb{R}^3$ in the reference configuration (in its undeformed state) and $\Omega \in \mathbb{R}^3$ under quasi-static loading, *i.e.* under the action of body forces $\boldsymbol{f}$ per unit deformed volume, traction forces $\boldsymbol{t}$ per unit deformed area of the boundary Γ_N and a displacement field $\boldsymbol{u}_D$ prescribed on the boundary Γ_D. The boundary $\partial\Omega$ of the domain in the current configuration is such that $\Gamma_N \cup \Gamma_D = \partial\Omega$ and $\Gamma_D \cap \Gamma_N = \emptyset$. The latter boundary conditions is referred to as Dirichlet boundary conditions and writes

$$\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{u}_D, \, \forall \boldsymbol{x} \in \Gamma_D. \tag{I.22}$$

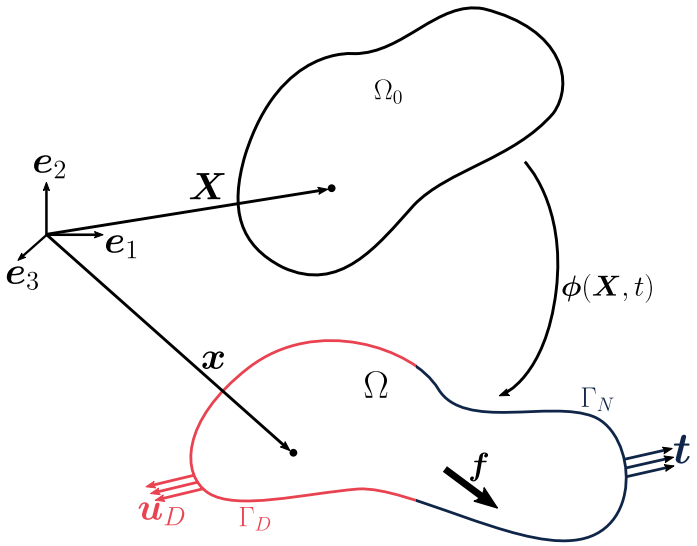


Figure I.1 – A general nonlinear continuum mechanics BVP involving reference Ω_0 and current Ω configurations.

First, the Cauchy stress tensor relates the traction vector $\boldsymbol{t}$ to the outer normal vector $\boldsymbol{n}$ of a deformed unit area as

$$\boldsymbol{t}(\boldsymbol{x}) = \boldsymbol{\sigma}(\boldsymbol{x})\boldsymbol{n}(\boldsymbol{x}), \, \forall \boldsymbol{x} \in \Gamma_N. \tag{I.23}$$

Eq. (I.23) is referred to as Neumann boundary conditions and can also be understood as an equilibrium equation on the boundary of the body. By contrast, the translational equilibrium, derived from the conservation of momentum *inside* the body, writes

$$\operatorname{div} \boldsymbol{\sigma}(\boldsymbol{x}) + \boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}, \, \forall \boldsymbol{x} \in \Omega \tag{I.24}$$

in the absence of inertial forces. The conservation of angular momentum in the body yields the well-known symmetry of the Cauchy stress tensor, conveniently expressed

using the Levi-Civita third-order tensor $\boldsymbol{\mathcal{E}}$ ($\boldsymbol{\mathcal{E}}_{ijk} = 1$ if i, j, k is an even permutation of $\{1, 2, 3\}$, -1 if it is odd, and 0 if any indices are repeated) as the vector equation

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^T \Leftrightarrow \boldsymbol{\mathcal{E}} : \boldsymbol{\sigma}^T = 0. \tag{I.25}$$

The principle of virtual work is classically derived by multiplying the local spatial equilibrium equation Eq. (I.24) by an arbitrary virtual velocity from the current position of the body $\delta\boldsymbol{v}$, integrating over the domain Ω and using the Gauss theorem. Expressing the virtual velocity gradient in terms of the symmetric virtual rate of deformation $\delta\boldsymbol{d} = \frac{1}{2}\left(\boldsymbol{\nabla}\delta\boldsymbol{v} + \boldsymbol{\nabla}^T\delta\boldsymbol{v}\right)$ and using the symmetry of the Cauchy stress tensor, the spatial virtual work equation writes

$$\delta W = \int_\Omega \boldsymbol{\sigma} : \delta\boldsymbol{d}\,\mathrm{d}v - \int_\Omega \boldsymbol{f}\cdot\delta\boldsymbol{v}\,\mathrm{d}v - \int_{\partial\Omega} \boldsymbol{t}\cdot\delta\boldsymbol{v}\,\mathrm{d}a = 0. \tag{I.26}$$

Alternative work conjugate pairs other than $(\boldsymbol{d}, \boldsymbol{\sigma})$ can be defined by expressing the principle of virtual work Eq. (I.26) in the material configuration. We write $\boldsymbol{f}_0 = J\boldsymbol{f}$ the body forces per unit reference volume and $\boldsymbol{t}_0 = \boldsymbol{t}(\,\mathrm{d}a/\,\mathrm{d}A)$ the traction vector per unit reference area[2]. The change of variables $\mathrm{d}V = J\,\mathrm{d}V$ and some manipulations gives a new expression of the virtual work equation Eq. (I.26),

$$\delta W = \int_{\Omega_0} \boldsymbol{P} : \delta\dot{\boldsymbol{F}}\,\mathrm{d}V - \int_{\Omega_0} \boldsymbol{f}_0\cdot\delta\boldsymbol{v}\,\mathrm{d}V - \int_{\partial\Omega_0} \boldsymbol{t}_0\cdot\delta\boldsymbol{v}\,\mathrm{d}A = 0, \tag{I.27}$$

which defines the first Piola-Kirchhoff stress tensor as work conjugate to the rate of the deformation gradient $\dot{\boldsymbol{F}}$:

$$\boldsymbol{P} = J\boldsymbol{\sigma}\boldsymbol{F}^{-T}. \tag{I.28}$$

The term "nominal" or "engineering" stress is also employed to refer to the first Piola-Kirchhoff stress tensor and is rooted in its physical meaning: while $\boldsymbol{\sigma}$ measures the traction force per unit *deformed* area, it can easily be shown that $\boldsymbol{P}$ measures the traction force per unit *reference* area, *i.e.* a sort of *nominal* traction force (Bonet and Wood, 2008). The nominal equivalent local equilibrium equations to Eqs. (I.23) to (I.25), then write:

$$\boldsymbol{P}(\boldsymbol{X})\boldsymbol{N}(\boldsymbol{X}) = \boldsymbol{t}_0, \qquad \forall \boldsymbol{X} \in \Gamma_{0_N} \tag{I.29}$$

$$\mathrm{DIV}\,\boldsymbol{P}(\boldsymbol{X}) + \boldsymbol{f}_0(\boldsymbol{X}) = \boldsymbol{0}, \qquad \forall \boldsymbol{X} \in \Omega_0, \tag{I.30}$$

$$\boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}\boldsymbol{P}^T\right)(\boldsymbol{X}) = \boldsymbol{0}, \qquad \forall \boldsymbol{X} \in \Omega_0, \tag{I.31}$$

where $\boldsymbol{N}$ is the outer normal of a unit area on the boundary Γ_{0_N} of the reference domain corresponding to the boundary Γ_N of the current domain and DIV denotes the divergence operator with respect to the initial coordinates.

[2]The area ratio is given by the Nanson's formula as

$$\frac{\mathrm{d}a}{\mathrm{d}A} = \frac{J}{\sqrt{\boldsymbol{n}\cdot\boldsymbol{b}\boldsymbol{n}}} = J\sqrt{\boldsymbol{N}\cdot\boldsymbol{C}^{-1}\boldsymbol{N}},$$

where $\boldsymbol{N}$ is the outer normal in the reference configuration.

As the deformation gradient tensor $\boldsymbol{F}$, the first Piola-Kirchhoff stress tensor $\boldsymbol{P}$ is an nonsymmetric two point tensor which relates reference and deformed configurations. It can be more convenient to use the totally material symmetric stress tensor $\boldsymbol{S}$, known as the second Piola-Kirchhoff stress tensor. Performing a pull-back operation on the spatial element of force $\mathrm{d}\boldsymbol{p} = \boldsymbol{t}\,\mathrm{d}a = \boldsymbol{\sigma}\boldsymbol{n}\,\mathrm{d}a = \boldsymbol{P}\boldsymbol{N}\,\mathrm{d}A$ yields the material force vector $\mathrm{d}\boldsymbol{p}_0 = \boldsymbol{t}_0\,\mathrm{d}A = \boldsymbol{F}^{-1}\,\mathrm{d}\boldsymbol{p}$, which defines $\boldsymbol{S}$ as $\mathrm{d}\boldsymbol{p}_0 = \boldsymbol{S}\boldsymbol{N}\,\mathrm{d}A$, *i.e.*

$$\boldsymbol{S} = \boldsymbol{F}^{-1}\boldsymbol{P}, \tag{I.32}$$

or

$$\boldsymbol{S} = J\boldsymbol{F}^{-1}\sigma\boldsymbol{F}^{-T}. \tag{I.33}$$

Now, injecting the second Piola-Kirchhoff – Cauchy stress relationship Eq. (I.33) and the strain rates relationship Eq. (I.21) into the spatial virtual work Eq. (I.26), we obtain the material virtual work equation

$$\delta W = \int_{\Omega_0} \boldsymbol{S} : \delta\dot{\boldsymbol{E}}\,\mathrm{d}V - \int_{\Omega_0} \boldsymbol{f}_0 \cdot \delta\boldsymbol{v}\,\mathrm{d}V - \int_{\partial\Omega_0} \boldsymbol{t}_0 \cdot \delta\boldsymbol{v}\,\mathrm{d}A = 0, \tag{I.34}$$

which shows that $(\dot{\boldsymbol{E}}, \boldsymbol{S})$ is another valid work conjugate pair.

The virtual work δW, regardless of the chosen form (material, nominal or spatial), is split into two terms: (i) the internal virtual work δW^{int} defined by the first integral involving the contraction between strain and stress, (ii) the external virtual work δW^{ext} containing the last two integrals involving the body and traction forces (see Eqs. (I.26), (I.27) and (I.34)).

We summarize the definitions and governing equations for kinematics and equilibrium of nonlinear quasi-static continuum mechanics in Table I.1; both material and spatial descriptions are linked through nominal quantities. The unknowns of a three-dimensional mechanical problem, namely the displacement, strain and stress fields, then comprise $3+9+9=21$ independent variables. The governing equations derived so far provide 15 independent equations (see details in brackets in Table I.1). To close the problem, one then need 6 additional independent equations, which are provided by a constitutive model describing the mechanical response of the material.

1.1.3 Constitutive equations: special case of hyperelasticity

Constitutive equations relate the primary variables (transformation mapping $\boldsymbol{\phi}$) and their gradients (gradient of deformation tensor $\boldsymbol{F}$) to the constitutive variables (stress tensor). They generally result from models, either motivated by physical phenomena or phenomenological observations. These relations involve parameters which are often considered in the end as material properties. Even if a great freedom is allowed in modeling, some physical principles must be verified by the constitutive models:

Table I.1 – Kinematic and static quantities and governing equations of nonlinear continuum mechanics in the material, nominal and spatial descriptions. The symbols $\boldsymbol{u}$ and $\boldsymbol{v}$ denote the displacement and velocity fields respectively.

		Material	**Nominal**	**Spatial**
Kinematics (compatibility)	(9)	$\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{C} - \boldsymbol{I}\right)$ with $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F}$	$\boldsymbol{F} = \boldsymbol{I} + \boldsymbol{\nabla}_0\boldsymbol{u}$	$\boldsymbol{e} = \frac{1}{2}\left(\boldsymbol{I} - \boldsymbol{b}^{-1}\right)$ with $\boldsymbol{b} = \boldsymbol{F}\boldsymbol{F}^T$
Strain rates		$\dot{\boldsymbol{E}} = \frac{1}{2}\left(\dot{\boldsymbol{F}}^T\boldsymbol{F} + \boldsymbol{F}^T\dot{\boldsymbol{F}}\right)$	$\dot{\boldsymbol{F}} = \frac{\partial \boldsymbol{v}}{\partial \boldsymbol{X}}$	$\boldsymbol{d} = \frac{1}{2}\left(\boldsymbol{\nabla}\boldsymbol{v} + \boldsymbol{\nabla}^T\boldsymbol{v}\right)$
Translational equilibrium (local form)	(3)	-	$\mathrm{DIV}\,\boldsymbol{P} + \boldsymbol{f}_0 = \boldsymbol{0}$	$\mathrm{div}\,\boldsymbol{\sigma} + \boldsymbol{f} = \boldsymbol{0}$
Rotational equilibrium (local form)	(3)	$\boldsymbol{\mathcal{E}} : \boldsymbol{S}^T = \boldsymbol{0}$	$\boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}\boldsymbol{P}^T\right) = \boldsymbol{0}$	$\boldsymbol{\mathcal{E}} : \boldsymbol{\sigma}^T = \boldsymbol{0}$
Work conjugate pairs		$\left(\dot{\boldsymbol{E}}, \boldsymbol{S}\right)$	$\left(\dot{\boldsymbol{F}}, \boldsymbol{P}\right)$	$(\boldsymbol{d}, \boldsymbol{\sigma})$
Principle of virtual work $\delta W = \delta W^{\mathrm{int}} - \delta W^{\mathrm{ext}} = 0$ (weak form)		$\delta W^{\mathrm{int}} = \int_{\Omega_0} \boldsymbol{S} : \delta\dot{\boldsymbol{E}}\,\mathrm{d}V$ $\delta W^{\mathrm{ext}} = \int_{\Omega_0} \boldsymbol{f}_0 \cdot \delta\boldsymbol{v}\,\mathrm{d}V - \int_{\partial\Omega_0} \boldsymbol{t}_0 \cdot \delta\boldsymbol{v}\,\mathrm{d}A.$	$\delta W^{\mathrm{int}} = \int_{\Omega_0} \boldsymbol{P} : \delta\dot{\boldsymbol{F}}\,\mathrm{d}V$ $\delta W^{\mathrm{ext}} = \int_{\Omega_0} \boldsymbol{f}_0 \cdot \delta\boldsymbol{v}\,\mathrm{d}V - \int_{\partial\Omega_0} \boldsymbol{t}_0 \cdot \delta\boldsymbol{v}\,\mathrm{d}A.$	$\delta W^{\mathrm{int}} = \int_{\Omega} \boldsymbol{\sigma} : \delta\boldsymbol{d}\,\mathrm{d}v$ $\delta W^{\mathrm{ext}} = \int_{\Omega} \boldsymbol{f} \cdot \delta\boldsymbol{v}\,\mathrm{d}v - \int_{\partial\Omega} \boldsymbol{t} \cdot \delta\boldsymbol{v}\,\mathrm{d}a.$

physical admissibility: respect of physical principles, such as non-decreasing net entropy;

determinism: constitutive variables at time t depend on primary variables at previous times ($\tau \leq t$);

locality: constitutive variables at material point $\boldsymbol{X}$ depend on primary variables within a close neighborhood of $\boldsymbol{X}$;

objectivity: constitutive equations must be frame-invariant.

A fairly simple class of constitutive equations which apply to finite strain are derived within the *hyperelasticity* theory. Such constitutive models conveniently describe the response of materials that exhibit large reversible deformations, such as elastomers, gels or some biological tissues. Hyperelasticty often serves as a basis for more complex constitutive models in finite strain, such as elastoplasticity, viscoelasticity, *etc.* In the present work, we focused on large elastic deformations only. Hence, we briefly recall the basic postulates of hyperelasticity, which provide the reference solution to compare with the data-driven computations conducted in this thesis.

First, the first Piola-Kirchhoff stress tensor $\boldsymbol{P}$ derives from an potential Ψ representing the stored elastic energy density:

$$\boldsymbol{P}\left(\boldsymbol{F}(\boldsymbol{X}), \boldsymbol{X}\right) = \frac{\partial \Psi\left(\boldsymbol{F}(\boldsymbol{X}), \boldsymbol{X}\right)}{\partial \boldsymbol{F}}. \tag{I.35}$$

Second, taking into account the objectivity principle implies that Ψ only depends on the stretch component $\boldsymbol{U}$ of the right polar decomposition of the deformation gradient tensor $\boldsymbol{F} = \boldsymbol{R}\boldsymbol{U}$, as the rotation component $\boldsymbol{R}$ represents rigid body rotations. Then, the constitutive relation Eq. (I.35) is more favorably expressed in terms of the second Piola-Kirchhoff stress tensor and the right Cauchy-Green stretch tensor $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F} = \boldsymbol{U}^2$ as

$$\boldsymbol{S}\left(\boldsymbol{C}(\boldsymbol{X}), \boldsymbol{X}\right) = 2\frac{\partial \Psi\left(\boldsymbol{C}(\boldsymbol{X}), \boldsymbol{X}\right)}{\partial \boldsymbol{C}} = \frac{\partial \Psi}{\partial \boldsymbol{E}}. \tag{I.36}$$

Third, in the case of isotropic materials, the constitutive equation Eq. (I.36) furthers simplifies to a function of the invariants of $\boldsymbol{C}$:

$$\boldsymbol{S}\left(I_C, II_C, III_C, \boldsymbol{X}\right) = 2\frac{\partial \Psi\left(I_C, II_C, III_C, \boldsymbol{X}\right)}{\partial \boldsymbol{C}}, \tag{I.37}$$

where the invariants of $\boldsymbol{C}$ are defined as[3]

$$I_C = \operatorname{tr} \boldsymbol{C} = \boldsymbol{C} : \boldsymbol{I}, \tag{I.38}$$

$$II_C = \operatorname{tr} \boldsymbol{C}^2 = \boldsymbol{C} : \boldsymbol{C}, \tag{I.39}$$

$$III_C = \det \boldsymbol{C} = J^2. \tag{I.40}$$

The derivative chain rule yields a compact form of the isotropic hyperelastic constitutive equation in the material description as

$$\boldsymbol{S} = 2\frac{\partial \Psi}{\partial I_C}\boldsymbol{I} + 4\frac{\partial \Psi}{\partial II_C}\boldsymbol{C} + 2J^2\frac{\partial \Psi}{\partial III_C}\boldsymbol{C}^{-1}. \tag{I.41}$$

Its spatial counterpart is readily derived using Eq. (I.33) and recalling that the invariants of $\boldsymbol{C}$ are the same as that of $\boldsymbol{b}$:

$$\boldsymbol{\sigma} = 2J^{-1}\frac{\partial \Psi}{\partial I_b}\boldsymbol{b} + 4J^{-1}\frac{\partial \Psi}{\partial II_b}\boldsymbol{b}^2 + 2\frac{\partial \Psi}{\partial III_b}\boldsymbol{I}. \tag{I.42}$$

To derive a hyperelastic model, it then suffices to propose a strain energy density function Ψ as a function of the stretch invariants (I_C, II_C, III_C).

The relationship Eq. (I.36) is generally nonlinear. As detailed in the next paragraph, its incorporation in the principle of virtual work Eq. (I.34) and its discretization with a finite element approximation yields a nonlinear system of algebraic equations. In computational mechanics, the system is usually addressed with an iterative technique which requires knowledge of the linearized constitutive equation. The linearization is a systematic process which is based on the concept of directional derivative, extensively presented in Bonet and Wood (2008); Holzapfel (2000) *inter alia*. Here, we merely provide the general definition below.

Definition I.1. *Directional derivative. Let consider a general multi-dimensional functional $\boldsymbol{\mathcal{F}}(\boldsymbol{x})$ with $\boldsymbol{x}$ being a list of unknown variables or functions. The directional derivative of $\boldsymbol{\mathcal{F}}(\boldsymbol{x})$ at $\boldsymbol{x}_0$ in the direction of $\boldsymbol{u}$ is defined as*

$$D\boldsymbol{\mathcal{F}}(\boldsymbol{x}_0)[\boldsymbol{u}] = \left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\right|_{\epsilon=0} \boldsymbol{\mathcal{F}}\left(\boldsymbol{x}_0 + \epsilon\boldsymbol{u}\right). \tag{I.43}$$

The linearization of the relationship between $\boldsymbol{S}$ and $\boldsymbol{E}$, given by Eq. (I.36), with respect to an increment $\boldsymbol{u}$ in the current configuration writes

$$D\boldsymbol{S}(\boldsymbol{x})[\boldsymbol{u}] = \mathbb{C} : D\boldsymbol{E}(\boldsymbol{x})[\boldsymbol{u}], \tag{I.44}$$

where the symmetric fourth-order tensor $\mathbb{C}$, known as the Lagrangian or material elasticity tensor is defined as

$$\mathbb{C} = \frac{\partial \boldsymbol{S}}{\partial \boldsymbol{E}} = 2\frac{\partial \boldsymbol{S}}{\partial \boldsymbol{C}} = \frac{\partial^2 \Psi}{\partial \boldsymbol{C} \partial \boldsymbol{C}}. \tag{I.45}$$

[3]The following alternate definition of the second invariant II_C is also commonly used for its more physical meaning:

$$II_C = \frac{1}{2}\left(I_C^2 - \boldsymbol{C} : \boldsymbol{C}\right).$$

In the rest of the manuscript, we precise which one is adopted whenever necessary.

The partial derivative $\partial \boldsymbol{S}/\partial \boldsymbol{E}$ in Eq. (I.45) is defined in indicial notation by [4]

$$\frac{\partial \boldsymbol{S}}{\partial \boldsymbol{E}} = \frac{\partial S_{IJ}}{\partial E_{KL}} \left(\boldsymbol{e}_I \otimes \boldsymbol{e}_J \otimes \boldsymbol{e}_K \otimes \boldsymbol{e}_L\right). \tag{I.46}$$

The equivalent Eulerian or spatial elasticity tensor $\mathbb{D}$ is obtained by the interpretation of the constitutive relation Eq. (I.44) into a rate form. It relates the Truesdell rate of the Cauchy stress tensor

$$\boldsymbol{\sigma}^{\circ} = J^{-1} \boldsymbol{F} \dot{\boldsymbol{S}} \boldsymbol{F}^T \tag{I.47}$$

to the rate of deformation tensor $\boldsymbol{d}$ as

$$\boldsymbol{\sigma}^{\circ} = \mathbb{D} : \boldsymbol{d}. \tag{I.48}$$

The spatial elasticity tensor $\mathbb{D}$ is related to its material counterpart $\mathbb{C}$ by

$$\mathbb{D} = J^{-1} F_{iI} F_{jJ} F_{kK} F_{lL} C_{IJKL} \left(\boldsymbol{e}_i \otimes \boldsymbol{e}_j \otimes \boldsymbol{e}_k \otimes \boldsymbol{e}_l\right). \tag{I.49}$$

Note that in Eq. (I.49) and in the rest of the manuscript, lower case indices i, j, k, l refer to the spatial coordinate system while upper case indices I, J, K, L refer to the material coordinate system.

In like manner, a nominal elasticity tensor $\mathbb{M}$ can be defined from the linearization of the relationship Eq. (I.35) as

$$D\boldsymbol{P}(\boldsymbol{x})[\boldsymbol{u}] = \mathbb{M} : D\boldsymbol{F}(\boldsymbol{x})[\boldsymbol{u}] \tag{I.50}$$

with

$$\begin{aligned} \mathbb{M} &= \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{F}} = M_{iJkL} \left(\boldsymbol{e}_i \otimes \boldsymbol{e}_J \otimes \boldsymbol{e}_k \otimes \boldsymbol{e}_L\right) \\ M_{iJkL} &= \frac{\partial P_{iJ}}{\partial F_{kL}} = \delta_{ik} S_{JL} + F_{iI} C_{IJKL} F_{kK}, \end{aligned} \tag{I.51}$$

where the directional derivative of the deformation gradient in the direction $\boldsymbol{u}$ is simply given by

$$D\boldsymbol{F}[\boldsymbol{u}] = \boldsymbol{\nabla}_0 \boldsymbol{u}. \tag{I.52}$$

[4] We remark to the interested reader that there exists an alternate definition for the partial derivative of a second-order tensor with respect to another second-order tensor, and hence for the fourth-order elasticity tensor, introduced by Itskov (2000). This requires an entirely new framework of tensor algebra and tensor analysis extensively presented in the textbook Itskov (2015). As these conventions do not change the physical results, we followed the commonly adopted definition of Bonet and Wood (2008) in the present work.

1.1.4 Linearized equilibrium equations

The weak form of the equilibrium given in Table I.1 is nonlinear with respect to both the geometry, due to finite strain measures, and the material, through the constitutive model. As aforementioned, the numerical procedure employed in computational mechanics relies on the linearized version of the virtual work, defined at given transformation mapping $\boldsymbol{\phi}$ and virtual velocity $\delta\boldsymbol{v}$, in the direction of an increment $\boldsymbol{u}$. Here, we do not give the detailed manipulations that yield this linearization; they can be found in Chapter 8 of (Bonet and Wood, 2008). The linearization of the equilibrium is considered in terms of an internal and external components as

$$\begin{aligned} D\delta W\left(\boldsymbol{\phi},\delta\boldsymbol{v}\right)\left[\boldsymbol{u}\right] &= D\delta\left(W^{\text{int}}-W^{\text{ext}}\right)\left(\boldsymbol{\phi},\delta\boldsymbol{v}\right)\left[\boldsymbol{u}\right] \\ &= D\delta W^{\text{int}}\left(\boldsymbol{\phi},\delta\boldsymbol{v}\right)\left[\boldsymbol{u}\right]-D\delta W^{\text{ext}}\left(\boldsymbol{\phi},\delta\boldsymbol{v}\right)\left[\boldsymbol{u}\right]. \end{aligned} \tag{I.53}$$

The material, nominal and spatial forms of the linearized internal virtual work $D\delta W^{\text{int}}\left(\boldsymbol{\phi},\delta\boldsymbol{v}\right)\left[\boldsymbol{u}\right]$ are given in Table I.2 where a hyperelastic constitutive relation of the form Eqs. (I.44) and (I.48) is considered. The expression of the linearized external virtual work is not given here, as it depends on the kind of traction forces applied. In particular, the case of uniform normal pressure is detailed in the Chapter 8 of (Bonet and Wood, 2008). Here, we only mention that the linearized external virtual work is null when the body forces are independent of the motion. Specifically, in the case of gravity loading, the external virtual work writes

$$W^{\text{ext}}_{\boldsymbol{f}}\left(\boldsymbol{\phi},\delta\boldsymbol{v}\right)=\int_{\Omega}\rho\boldsymbol{g}\cdot\delta\boldsymbol{v}\,\mathrm{d}V, \tag{I.54a}$$

and hence

$$D\delta W^{\text{ext}}_{\boldsymbol{f}}\left(\boldsymbol{\phi},\delta\boldsymbol{v}\right)\left[\boldsymbol{u}\right]=0, \tag{I.54b}$$

with ρ the density of the material and $\boldsymbol{g}$ the acceleration due to gravity.

1.1.5 Special case of linear elasticity

To close this section, we briefly give the linearized or small strain counterpart of the previous equations. First, all strain measures are reduced to the small strain tensor $\boldsymbol{\varepsilon}$ and the rate of deformation tensor $\boldsymbol{d}$ is simply the time derivative of $\boldsymbol{\varepsilon}$

$$\boldsymbol{d}=\frac{1}{2}\left(\boldsymbol{\nabla}\boldsymbol{v}+\boldsymbol{\nabla}^{T}\boldsymbol{v}\right)=\dot{\boldsymbol{\varepsilon}} \tag{I.58}$$

Second, all stress measures are reduced to the Cauchy stress tensor $\boldsymbol{\sigma}$. The reference and current configuration being indistinguishable, only one set of local equilibrium equations remains:

$$\boldsymbol{\sigma}\boldsymbol{n}=\boldsymbol{t}, \qquad \forall\boldsymbol{x}\in\Gamma_{N}, \tag{I.59}$$

$$\operatorname{div}\boldsymbol{\sigma}+\boldsymbol{f}=\boldsymbol{0}, \qquad \forall\boldsymbol{x}\in\Omega, \tag{I.60}$$

$$\boldsymbol{\mathcal{E}}:\boldsymbol{\sigma}=\boldsymbol{0}, \qquad \forall\boldsymbol{x}\in\Omega, \tag{I.61}$$

Table I.2 – Expressions of the linearized internal virtual work $D\delta W^{\text{int}}(\boldsymbol{\phi}, \delta\boldsymbol{v})[\boldsymbol{u}]$ in the material, nominal and spatial descriptions.

Description	**Linearized internal virtual work**	
Material	$D\delta W^{\text{int}}(\boldsymbol{\phi}, \delta\boldsymbol{v})[\boldsymbol{u}] = \int_{\Omega_0} D\boldsymbol{E}[\delta\boldsymbol{v}] : \mathbb{C} : D\boldsymbol{E}[\boldsymbol{u}] \, \mathrm{d}V + \int_{\Omega_0} \boldsymbol{S} : \left(\boldsymbol{\nabla}_0^T\boldsymbol{u} \; \boldsymbol{\nabla}_0\delta\boldsymbol{v}\right) \, \mathrm{d}V$	(I.55)
Nominal	$D\delta W^{\text{int}}(\boldsymbol{\phi}, \delta\boldsymbol{v})[\boldsymbol{u}] = \int_{\Omega_0} \boldsymbol{\nabla}_0\delta\boldsymbol{v} : \mathbb{M} : \boldsymbol{\nabla}_0\boldsymbol{u} \, \mathrm{d}V$	(I.56)
Spatial	$D\delta W^{\text{int}}(\boldsymbol{\phi}, \delta\boldsymbol{v})[\boldsymbol{u}] = \int_{\Omega} \delta\boldsymbol{d} : \mathbb{D} : \boldsymbol{\varepsilon} \, \mathrm{d}V + \int_{\Omega} \boldsymbol{\sigma} : \left(\boldsymbol{\nabla}^T\boldsymbol{u} \; \boldsymbol{\nabla}\delta\boldsymbol{v}\right) \, \mathrm{d}v$	(I.57)

which are summarized in the linearized principle of virtual work as

$$\delta\hat{W} = \int_\Omega \boldsymbol{\sigma} : \boldsymbol{\varepsilon} \, \mathrm{d}v - \int_\Omega \boldsymbol{f} \cdot \delta\boldsymbol{v} \, \mathrm{d}v - \int_{\partial\Omega} \boldsymbol{t} \cdot \delta\boldsymbol{v} \, \mathrm{d}a. \tag{I.62}$$

The constitutive relation is also linearized and reduces to the standard linear elastic solid (Hooke's law) defined by the constant fourth-order elasticity tensor $\mathbb{H}$, such that

$$\boldsymbol{\sigma} = \mathbb{H} : \boldsymbol{\varepsilon}. \tag{I.63}$$

In the case of isotropy, only two independent parameters suffice to describe $\mathbb{H}$, and Hooke's law takes one of the following forms:

$$\boldsymbol{\sigma} = \lambda \operatorname{tr} \boldsymbol{\varepsilon} \boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}, \tag{I.64}$$

or

$$\boldsymbol{\sigma} = 3\kappa \operatorname{vol} \boldsymbol{\varepsilon} + 2\mu \operatorname{dev} \boldsymbol{\varepsilon}, \tag{I.65}$$

or

$$\boldsymbol{\varepsilon} = \frac{1+\nu}{E}\boldsymbol{\sigma} - \frac{\nu}{E} \operatorname{tr} \boldsymbol{\sigma} \boldsymbol{I}, \tag{I.66}$$

where λ and μ are the Lamé constants, κ the bulk modulus accounting for the compressibility of the material, ν the Poisson's ratio and E the Young's modulus. The 2^{nd} Lamé constant μ is also the shear modulus. The operators $\operatorname{vol} \bullet$ and $\operatorname{dev} \bullet$ acting on second-order tensors are defined as

$$\operatorname{vol} \bullet = \frac{1}{3} \operatorname{tr} \bullet \, \boldsymbol{I}, \tag{I.67}$$

$$\operatorname{dev} \bullet = \bullet - \operatorname{vol} \bullet. \tag{I.68}$$

1.2 Finite element discretization of nonlinear continuum mechanics

In the governing equations of nonlinear continuum mechanics reminded above, two sources of nonlinearity co-exist:

geometric nonlinearity arises when "changes in geometry have a significant effect on the load deformation behavior", which includes large strain, deformation-dependent boundary conditions and loading, or geometric instabilities (Bonet and Wood, 2008);

material nonlinearity occurs when the constitutive equation relating strain and stress is nonlinear, which includes stiffening effects for instance.

Both nonlinearities render the partial differential equations impossible to solve by-hand, apart in simple problems. That is why numerical analysis, which relies on a discretization of the domain, is an essential tool to calculate the mechanical response of complex structures subject to complex loading conditions. The most commonly used discretization technique in solid mechanics is the FE method which Bonet and Wood (2008) summarize as follows:

> It is a procedure whereby the continuum behavior described at an infinity of points is approximated in terms of a finite number of points, called *nodes*, located at specific points in the continuum. These nodes are used to define regions, called *finite elements*, over which both the geometry and the primary variables in the governing equations are approximated.

To complete this definition, we add that the finite elements constitute the compact support of a set of approximation or basis or shape functions used to approximate the solution (a shape function is nonzero only over the domain occupied by the attached finite element). The FE method is based upon integral formulations of partial differential equations. In mechanics, it thus uses the weak form Eqs. (I.26), (I.27) and (I.34) of the governing equations. After discretization, these are rendered into a nonlinear system of algebraic equations. In the following, we briefly revisit the discretized form of these equations to provide the Newton-Raphson procedure that is commonly used to solve such a nonlinear system.

1.2.1 Discretized equilibrium equations

The discretization is herein restricted to isoparametric elements, meaning that the approximation functions for the geometry and for the primary variable (displacement field) are the same. Let us consider a finite element mesh approximating the body in Ω_0 with N nodes and M elements. Each element e comprises N_e nodes and occupies a domain $\Omega_0^{(e)}$ ($\Omega^{(e)}$) in the reference (current) configuration. The position $\boldsymbol{X}$ ($\boldsymbol{x}$) of a material point in the reference (current) configuration of element e is interpolated from the positions $\boldsymbol{X}_a$ of the nodes with the nodal shape functions $\mathcal{N}_a(\boldsymbol{\xi})$ as

$$\boldsymbol{X} = \sum_{a=1}^{N_e} \mathcal{N}_a(\boldsymbol{\xi}) \boldsymbol{X}_a, \tag{I.69}$$

$$\boldsymbol{x} = \sum_{a=1}^{N_e} \mathcal{N}_a(\boldsymbol{\xi}) \boldsymbol{x}_a, \tag{I.70}$$

where $\boldsymbol{\xi}$ is the isoparametric coordinate of the integration point in the unit reference element. Consistently, the displacement field $\boldsymbol{u}$ and the virtual velocity field $\delta\boldsymbol{v}$ are

interpolated as

$$\boldsymbol{u}(\boldsymbol{X}) = \sum_{a=1}^{N_e} \mathcal{N}_a(\boldsymbol{\xi})\boldsymbol{u}_a, \tag{I.71}$$

$$\delta\boldsymbol{v} = \sum_{a=1}^{N_e} \mathcal{N}_a(\boldsymbol{\xi})\delta\boldsymbol{v}_a. \tag{I.72}$$

The gradients of the primary variables are then easily derived as

$$\boldsymbol{F} = \boldsymbol{\nabla}_0\boldsymbol{x} = \sum_{a=1}^{N_e} \boldsymbol{x}_a \otimes \boldsymbol{\nabla}_0\mathcal{N}_a, \tag{I.73}$$

$$\delta\boldsymbol{d} = \frac{1}{2}\sum_{a=1}^{N_e} \left(\delta\boldsymbol{v}_a \otimes \boldsymbol{\nabla}_0\mathcal{N}_a + \boldsymbol{\nabla}_0\mathcal{N}_a \otimes \delta\boldsymbol{v}_a\right), \tag{I.74}$$

where the gradient of the shape functions with respect to the spatial and material configuration are obtained from their (known) derivatives with respect to the isoparametric coordinates as

$$\boldsymbol{\nabla}_0\mathcal{N}_a = \frac{\partial\mathcal{N}_a}{\partial\boldsymbol{\xi}}\left(\frac{\partial\boldsymbol{X}}{\partial\boldsymbol{\xi}}\right)^{-1}, \qquad \text{with} \qquad \frac{\partial\boldsymbol{X}}{\partial\boldsymbol{\xi}} = \sum_{a=1}^{N_e} \boldsymbol{X}_a \otimes \boldsymbol{\nabla}_\xi\mathcal{N}_a, \tag{I.75}$$

$$\boldsymbol{\nabla}\mathcal{N}_a = \frac{\partial\mathcal{N}_a}{\partial\boldsymbol{\xi}}\left(\frac{\partial\boldsymbol{x}}{\partial\boldsymbol{\xi}}\right)^{-1}, \qquad \text{with} \qquad \frac{\partial\boldsymbol{x}}{\partial\boldsymbol{\xi}} = \sum_{a=1}^{N_e} \boldsymbol{x}_a \otimes \boldsymbol{\nabla}_\xi\mathcal{N}_a. \tag{I.76}$$

Injecting this discretization into each form of the principle of virtual work yields the discretized virtual work equation

$$\delta\boldsymbol{v}_a \cdot \boldsymbol{R}_a = 0, \forall a = 1 \,..\, N, \tag{I.77}$$

which involves the nodal residual force vector

$$\boldsymbol{R}_a = \boldsymbol{T}_a^{\text{int}} - \boldsymbol{T}_a^{\text{ext}} \tag{I.78}$$

representing the balance between the internal force vector $\boldsymbol{T}_a^{\text{int}}$ and the external force vector $\boldsymbol{T}_a^{\text{ext}}$. The former is equivalently evaluated on all configurations, depending on the choice of implementation; the different forms are then listed in Table I.3, in which the summation symbol $\sum_{\substack{e=1\\e\ni a}}^{M_a} \bullet$ denotes the assembly of the quantity $\bullet$ at node a over the M_a elements of the mesh containing node a, labeled $e \ni a$. The reader is referred to Bonet and Wood (2008) for the detailed derivations of these expressions and the assembly procedure. The latter is preferably expressed in the spatial configuration as it allows for taking into account deformation-dependent loading such as applied normal pressure more easily. Specifically,

$$\boldsymbol{T}_a^{\text{ext}} = \sum_{\substack{e=1\\e\ni a}}^{M_a} \left(\int_{\Omega^{(e)}} \mathcal{N}_a\boldsymbol{f}\,\mathrm{d}v + \int_{\partial\Omega^{(e)}} \mathcal{N}_a\boldsymbol{t}\,\mathrm{d}a\right). \tag{I.79}$$

Table I.3 – Finite element discretizations of the internal force vector $\boldsymbol{T}_a^{\text{int}}$ assembled at node a, using isoparametric elements, in the material, nominal and spatial form.

Description	Discretized nodal force vectors	
Material	$\boldsymbol{T}_a^{\text{int}} = \sum_{\substack{e=1 \\ e \ni a}}^{M_a} \int_{\Omega_0^{(e)}} \boldsymbol{F}\boldsymbol{S}\boldsymbol{\nabla}_0 \mathcal{N}_a \, \mathrm{d}V$	(I.80)
Nominal	$\boldsymbol{T}_a^{\text{int}} = \sum_{\substack{e=1 \\ e \ni a}}^{M_a} \int_{\Omega_0^{(e)}} \boldsymbol{P}\boldsymbol{\nabla}_0 \mathcal{N}_a \, \mathrm{d}V$	(I.81)
Spatial (or linearized kinematics)	$\boldsymbol{T}_a^{\text{int}} = \sum_{\substack{e=1 \\ e \ni a}}^{M_a} \int_{\Omega^{(e)}} \boldsymbol{\sigma}\boldsymbol{\nabla} \mathcal{N}_a \, \mathrm{d}v$	(I.82)

The virtual nodal velocities $\delta\boldsymbol{v}$ being arbitrary, Eq. (I.77) reduces to a nonlinear system of equations expressed by the nodal residual force vectors as

$$\boldsymbol{R}_a = \boldsymbol{0}, \forall a = 1 \,..\, N. \tag{I.83}$$

1.2.2 Solution to the nonlinear discrete problem

The discretized version of the principle of virtual work then yields a nonlinear system of equations Eq. (I.78) which represents the balance between internal and external forces at every node on the structure, with the nodal current positions (or displacements) as the unknowns. The most commonly used numerical scheme to solve these nonlinear equations is the Newton-Raphson iterative procedure which requires the linearization of the residual in the direction of an incremental displacement $\boldsymbol{u}$. A typical iteration k of the algorithm writes:

$$\delta W(\boldsymbol{\phi}_k, \delta\boldsymbol{v}) + D\delta W(\boldsymbol{\phi}_k, \delta\boldsymbol{v})[\boldsymbol{u}] = 0; \quad \boldsymbol{\phi}_{k+1} = \phi_k + \boldsymbol{u}, \tag{I.84}$$

where

$$D\delta W(\boldsymbol{\phi}_k, \delta\boldsymbol{v})[\boldsymbol{u}] = D\delta W^{\text{int}}(\boldsymbol{\phi}_k, \delta\boldsymbol{v})[\boldsymbol{u}] - D\delta W^{\text{ext}}(\boldsymbol{\phi}_k, \delta\boldsymbol{v})[\boldsymbol{u}]$$

with $D\delta W^{\text{int}}(\boldsymbol{\phi}_k, \delta\boldsymbol{v})[\boldsymbol{u}]$ as in Eqs. (I.55) to (I.57). Using the finite element discretization of the geometry and the primary variables together with the linearized virtual work equations given in Eq. (I.53) and Table I.2, Eq. (I.84) is rendered in

the following discrete form:

$$\boldsymbol{R}_a + \sum_{b=1}^{N_a} \boldsymbol{K}_{ab}\boldsymbol{u}_b = 0, \forall a = 1 \,..\, N, \tag{I.85}$$

with

$$\boldsymbol{K}_{ab} = \left(\boldsymbol{K}_{ab}^{\text{int}} - \boldsymbol{K}_{ab}^{\text{ext}}\right) \tag{I.86}$$

where $\boldsymbol{K}_{ab}^{\text{int}}$ (respectively $\boldsymbol{K}_{ab}^{\text{ext}}$) is the internal (respectively external) tangent stiffness matrix of the system and N_a denotes the number of nodes connected to node a. The tangent stiffness matrix $\boldsymbol{K}_{ab}$ measures the change in forces at node a due to change in the current position of node b. The explicit formulation of the external stiffness matrix $\boldsymbol{K}_{ab}^{\text{ext}}$ is left aside here since it depends on the type of loading. However, a general expression of the internal stiffness matrix comprises two component: (i) a constitutive component $\boldsymbol{K}_{ab}^{c}$ corresponding to the term of the linearized internal work which involves the constitutive relationship, (ii) an initial stress component $\boldsymbol{K}_{ab}^{\sigma}$ resulting from the additional stress term of the linearized internal work which derives from the nonlinearity of the material and spatial strain measures:

$$\boldsymbol{K}_{ab}^{\text{int}} = \boldsymbol{K}_{ab}^{c} + \boldsymbol{K}_{ab}^{\sigma}. \tag{I.87}$$

We gather the expressions of the internal tangent stiffness matrix $\boldsymbol{K}_{ab}^{\text{int}}$ in the material, nominal and spatial descriptions in Table I.4.

A typical Newton-Raphson iteration k in a finite element code then writes:

$$\boldsymbol{R}(\boldsymbol{x}_k) + \boldsymbol{K}(\boldsymbol{x}_k)\boldsymbol{u} = 0, \text{ and } \boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{u}, \tag{I.93}$$

where $\boldsymbol{R} = \left[\boldsymbol{R}_1^T, \boldsymbol{R}_2^T, \cdots, \boldsymbol{R}_N^T\right]^T$ is the complete residual force vector gathering all nodal equivalent forces, $\boldsymbol{u} = \left[\boldsymbol{u}_1^T, \boldsymbol{u}_2^T, \cdots, \boldsymbol{u}_N^T\right]^T$ the complete nodal displacements vector and $\boldsymbol{K}$ the complete tangent stiffness matrix defined by assembling the nodal components as

$$\boldsymbol{K} = \begin{pmatrix} \boldsymbol{K}_{11} & \boldsymbol{K}_{12} & \cdots & \boldsymbol{K}_{1N} \\ \boldsymbol{K}_{21} & \boldsymbol{K}_{22} & \cdots & \boldsymbol{K}_{2N} \\ \vdots & \ddots & \vdots & \\ \boldsymbol{K}_{N1} & \boldsymbol{K}_{N2} & \cdots & \boldsymbol{K}_{NN} \end{pmatrix} \tag{I.94}$$

1.2.3 Discrete evaluation of integrals

As a final remark, all finite element quantities presented above require the evaluation of integrals over elementary domains. However, the underlying mathematical expressions of the integrands are generally too complex to perform an exact integration. The FE method thus also comprises an appropriate integration or Gauss

Table I.4 – Finite element discretizations of the internal tangent stiffness matrix components, assembled at all nodes b which are connected to node a, using isoparametric elements, in the material, nominal and spatial form. In Eqs. (I.81), (I.88) and (I.92) and throughout the manuscript, the summation symbol $\sum_{\substack{e=1\\e\ni a,b}}^{M_{ab}} \bullet$ denotes the assembly of the quantity $\bullet$ over the M_a elements of the mesh containing both nodes a and b, labeled $e \ni a, b$.

Description	**Constitutive component**	**Initial stress component**
Material	$[\boldsymbol{K}^c_{ab}]_{ij} = \sum_{\substack{e=1\\e\ni a,b}}^{M_{ab}} \int_{\Omega_0^{(e)}} F_{iI} \frac{\partial \mathcal{N}_a}{\partial X_J} C_{IJKL} \frac{\partial \mathcal{N}_a}{\partial X_K} F_{jL} \, \mathrm{d}V$ (I.88)	$\boldsymbol{K}^\sigma_a = \sum_{\substack{e=1\\e\ni a,b}}^{M_{ab}} \int_{\Omega_0^{(e)}} (\boldsymbol{\nabla}_0 \mathcal{N}_a \cdot \boldsymbol{S} \boldsymbol{\nabla}_0 \mathcal{N}_b) \, \boldsymbol{I} \, \mathrm{d}V$ (I.89)
Nominal	$[\boldsymbol{K}^c_{ab}]_{ij} = \sum_{\substack{e=1\\e\ni a,b}}^{M_{ab}} \int_{\Omega_0^{(e)}} F_{iI} \frac{\partial \mathcal{N}_a}{\partial X_K} M_{iKjL} \frac{\partial \mathcal{N}_a}{\partial X_L} \, \mathrm{d}V$ (I.90)	Not defined: the compatibility equation Eq. (I.11) is linear. Hidden in the stress term of $\mathbb{M}$ (see Eq. (I.50)).
Spatial	$[\boldsymbol{K}^c_{ab}]_{ij} = \sum_{\substack{e=1\\e\ni a,b}}^{M_{ab}} \int_{\Omega^{(e)}} \frac{\partial \mathcal{N}_a}{\partial x_k} D_{ikjl} \frac{\partial \mathcal{N}_a}{\partial x_l} \, \mathrm{d}v$ (I.91)	$\boldsymbol{K}^\sigma_a = \sum_{\substack{e=1\\e\ni a,b}}^{M_{ab}} \int_{\Omega^{(e)}} (\boldsymbol{\nabla} \mathcal{N}_a \cdot \boldsymbol{\sigma} \boldsymbol{\nabla} \mathcal{N}_b) \, \boldsymbol{I} \, \mathrm{d}v$ (I.92)

quadrature rule, so as to replace integrals by weighted sums of the discrete evaluations of the fields at integration points. Every integral of a functional $\boldsymbol{\mathcal{F}}$ over the elementary domain $\Omega^{(e)}$ of the current configuration is then computed as

$$\int_{\Omega^{(e)}} \boldsymbol{\mathcal{F}}(\boldsymbol{x})\,\mathrm{d}v = \int_{\hat{\Omega}^{(e)}} \boldsymbol{\mathcal{F}}(\boldsymbol{\xi}) \det\left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\xi}}\right) \mathrm{d}v = \sum_{g=1}^{P_e} w_g \boldsymbol{\mathcal{F}}_g, \tag{I.95}$$

where $\hat{\Omega}^{(e)}$ is the elementary domain in the isoparametric coordinate system, $w_g = \omega_g \det\left(\frac{\partial \boldsymbol{x}_g}{\partial \boldsymbol{\xi}}\right)$ is the current volume associated to integration point g of weight ω_g and coordinate $\boldsymbol{x}_g$ in the current configuration, $\boldsymbol{\mathcal{F}}_g = \boldsymbol{\mathcal{F}}(\boldsymbol{x}_g)$ and P_e the number of integration points in the element. The same applies to the integrals over the elementary domain $\Omega_0^{(e)}$ in the reference configuration:

$$\int_{\Omega_0^{(e)}} \boldsymbol{\mathcal{F}}(\boldsymbol{X})\,\mathrm{d}V = \int_{\hat{\Omega}^{(e)}} \boldsymbol{\mathcal{F}}(\boldsymbol{\xi}) \det\left(\frac{\partial \boldsymbol{X}}{\partial \boldsymbol{\xi}}\right) \mathrm{d}V = \sum_{g=1}^{P_e} w_g \boldsymbol{\mathcal{F}}_g, \tag{I.96}$$

with $w_g = \omega_g \det\left(\frac{\partial \boldsymbol{x}_g}{\partial \boldsymbol{\xi}}\right)$ now being defined as the reference volume associated to integration point g of coordinate $\boldsymbol{X}_g$ in the reference configuration, and $\boldsymbol{\mathcal{F}}_g = \boldsymbol{\mathcal{F}}(\boldsymbol{X}_g)$.

Hence, in the finite element setting, the primary variables (displacement field) are evaluated at the node while their gradient (constitutive variables, strain and stress fields) are evaluated at the integration points. In the rest of the manuscript, the finite element formulation of the BVP will then be directly defined by these discrete unknowns. We shall only refer to the nodes and the integration points, bypassing intermediate computations (integrals) on the elements. Every variational formulation resulting in the integral of a functional $\boldsymbol{\mathcal{F}}$ over the whole domain Ω_0 (and likewise over Ω) is then directly expressed as the weighted sum

$$\int_{\Omega_0} \boldsymbol{\mathcal{F}}(\boldsymbol{X})\,\mathrm{d}V = \sum_{g=1}^{P} w_g \boldsymbol{\mathcal{F}}_g \tag{I.97}$$

with $P = \sum_{e=1}^{M} P_e$ the total number of integration points in the mesh.

1.3 Summary

To summarize, the finite element method applied to nonlinear mechanics of solids consists in solving a nonlinear system of algebraic equations derived from the weak form of equilibrium equations, using the Newton-Raphson scheme based on the tangent stiffness matrix. It is worth noting that the primary variables (positions and displacements) are computed at the nodes and that the constitutive variables (strains and stress) are computed within the elements, discretely at the integration points. The three formulations (material, nominal or spatial) are all equivalent and yield the same results. Nevertheless, each presents specific features which can advocate for their implementation, depending on the user's preferences and the problem:

nominal: In this setting, the displacement-deformation relation Eq. (I.11) is linear, which suppresses the initial stress component of the internal tangent stiffness matrix. In addition, since the integration is performed on the reference configuration, the gradient of the shapes functions can be computed once and for all at the beginning of the calculation (see Eq. (I.75)).

spatial: This formulation is particularly adapted to deformation-dependent loading conditions which require to update the current configuration at each iteration. A convenient matrix formulation is also available which allows for the use of sparse solvers.

material: This description is equally interesting than the nominal formulation in terms of the gradient of the shape functions. In addition, it allows to work with fully symmetric tensors.

1.4 The key concept and central difficulty of constitutive modeling

As discussed above, the FE method applied to nonlinear continuum mechanics is more complex than the standard linear analysis of solids and requires a careful implementation of the chosen formulation. Of course, we only presented the elastic framework; additional derivations and concepts must be introduced in order to predict anelastic behaviors such as plasticity or viscoelasticity. Moreover, multi-physics phenomena such as thermo-mechanics involve supplementary governing equations. Leaving these complex couplings aside, the most difficult aspect of computational mechanics is undoubtedly the derivation, implementation and identification of the constitutive model.

1.4.1 Derivation

In solid mechanics, and particularly for hyperelasticity, most of the constitutive models are derived following the so-called phenomenological approach. This approach consists in "fitting mathematical equations to experimental data", but is "not capable of relating the mechanism of deformation to the underlying physical (microscopic) structure of the material" (Holzapfel, 2000). In this context a constitutive theory is developed in order to give a consistent framework for the elaboration of a model able to reproduce the response of the material. For instance, much effort has been devoted to the development of the finite (hyper)elasticity theory, which involves the principles we recalled in Section 1.1.3 together with the polyconvexity of strain-energy functions (see Holzapfel (2000, Chapter 6) for more explanation and references on this subject). A thermodynamical approach, based on the work of Coleman and Noll (1974) provides the generalized standard materials framework in for anelastic materials using internal variables (Germain et al., 1983; Halphen and Son Nguyen, 1975).

1.4.2 Implementation

The thermo-mechanical framework for the development of constitutive models in then well established. However, as it offers great freedom, one must choose the most suitable model for the material considered among a large variety of propositions: entire scientific journals are exclusively dedicated to the research on constitutive modeling and for rubber-like materials only, several reviews have compared the performances of each model to date (see Boyce and Arruda (2000); Marckmann and Verron (2006); Steinmann et al. (2012); Mihai and Goriely (2017) *inter alia*). In particular, for an engineer faced with a commercial FE software it is not straightforward which constitutive equations to choose that will best predict the excepted behavior of the material in the structure considered. For complex models, dedicated implementations and algorithms can be needed: see from example the book of Simo and Hughes (1998) devoted to the implementation of inelasticity, or the LArge Time INcrement method (LATIN) method introduced by Ladevèze in 1984 and extensively detailed in (Ladevèze, 1996) to deal with nonlinearity. We can also mention the Z-mat material library, part of the Z-set suite, which provides a collection of constitutive models for non-linear material behaviors Chaboche and Cailletaud (1996). The availability of complex constitutive models and the development of associated integrations tools in FE codes or softwares constitutes a large part of research effort in computational mechanics.

1.4.3 Identification

Once the appropriate constitutive model has been selected and made available in standard FE software, it still remains for the engineer to identify the parameters for the considered material, with experimental data. Often, only one strain-stress curve extracted from a uniaxial tensile test is available, but it offers insufficient data to identify numerous parameters. The identification process is then generally an ill-posed problem comprising multiple solutions. The reliability of the model in other scenarios than the training one can then be uncertain: fitting a hyperelastic model on uniaxial tensile test only or using supplementary data from other deformation modes such as pure shear or biaxial tension can yield different values of the parameters. Hence, it results in different predictions for the three-dimensional material behavior (Verron, 2018). As discussed next, the recent developments of experimental techniques and monitoring tools, *e.g.* full-field measurements, addresses these identification issues by providing richer experimental data.

1.4.4 Special case of heterogeneous materials

The difficult process of developing, identifying and implementing a constitutive model is even more so in the case of so-called heterogeneous materials, *i.e.* multiphase materials. Indeed, their effective behavior is strongly dependent on the material response of the constituents themselves and on their arrangement in the microstructure. They are also generally anisotropic and complex microstructural phe-

nomena occur during deformation, impacting the macroscopic response. An entire branch of research on mechanics of materials is focused on the homogenization of microscopic properties into an effective material response of heterogeneous materials. Starting from the Eshelby solution of an ellipsoidal inclusion embedded in an infinite elastic medium and subject to a far-field strain (Eshelby and Peierls, 1957), mean field theory was developed to model more complex microstructures and anelastic behaviors, such as elastoplasicity, based on the work of Hill (1965). Again, the more microstructural phenomena one aims at capturing, the more complex the mean field model. Breaking with this modeling approach the so-called *computational homogenization* was introduced in the late 1990s and has seen various contributions and developments ever since (see for instance the review of Geers et al. (2010)). An application of this approach to elastoviscoplastic materials resulted in the so-called FE^2 scheme introduced by Feyel (1999); Feyel and Chaboche (2000). It consists in a multiscale numerical scheme, which bypasses the need for modeling the effective behavior of the microstructure. Instead, two nested BVPs are concurrently solved using FE analysis: a computation on a Representative Volume Element (RVE) of the microstructure is conducted at each integration point of the macroscopic mesh, with boundary conditions provided by the strain state of the macroscopic point. The macroscopic stress state is subsequently obtained from a volume average of the microscopic stress field in the RVE. This procedure makes use of the constitutive equations at the microscopic level only, as illustrated in Fig. I.2. However, it is

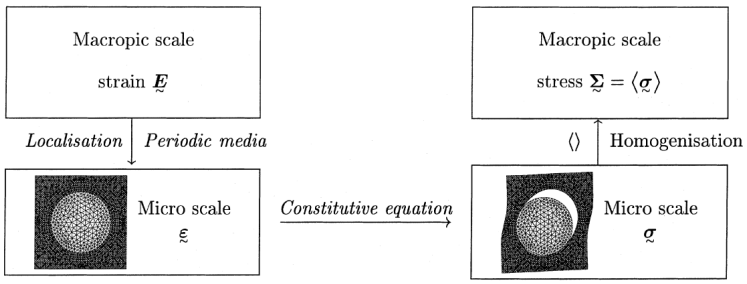


Figure I.2 – Schematic diagram of the FE^2 model. From Feyel (1999).

computationally very demanding, both in CPU time and in memory requirements, and researchers are currently focusing on reducing the cost, either *via* parallel implementation and/or through reduced order modeling concepts.

1.4.5 Summary

To sum up, constitutive models are one of the key ingredients in computational mechanics. As demonstrated, a current major focus in mechanics of materials is the development of constitutive models. However, the more complex the constitutive models, the greater the need for experimental data to calibrate the parameters and also the more carefully their domain of validity must be defined. Material

modeling then appears as one possible entry-point for data science in computational mechanics, as we show next.

2 Data science in computational mechanics

In this section, we take inspiration from the argument Michael Ortiz gave in his conference at the XIII International Conference on Computational Plasticity (COMPLAS), in 2015, to motivate the data-driven computational mechanics approach (Ortiz, 2015).

2.1 Data in computational mechanics

Computational mechanics is a field in which a large amount of *data* is both consumed and produced. On the one side, the recent developments of experimental measurements techniques have provided rich databases to the identification process of the constitutive models used in FE simulations. In particular, identification techniques coupling computational mechanics with full-field measurements such as Finite Element Model Updating, Virtual Field Method, or Constitutive Equation Gap Method have been developed in recent years (see the review of Avril et al. (2008)). On the other side, heavy computations, such as multiscale analyses, produce a huge amount of discrete values of displacements, strains, stresses, or strain energy densities, which can be used in the end to extract knowledge on the overall material behavior. For example, recent achievements in high performance computing has enabled Mosby and Matouš (2016) to perform a simulation consisting of 53.8 Billion finite elements with 28.1 Billion nonlinear equations that was solved on 393 216 computing cores, using a hierarchically parallel implementation of the computational homogenization formulation (Matouš et al., 2017). These kind of massive multiscale computations are intended to constitute "the basis of *Virtual Materials Testing* standards, and to aid in the development of new material formulations with extreme properties" (Mosby and Matouš, 2016).

Any numerical analysis is then nowadays both further down and in the upstream part of a production chain of data, making the field of computational mechanics a perfect candidate for application of data science.

2.2 What is data science and how it fits in the field of computational mechanics

Here, data science is understood as "the study of the generalizable extraction of knowledge from data". "Knowledge" refers to underlying patterns or "insights" in unstructured data; then, "unlike database querying, which asks 'What data satisfies this pattern (query)?', discovery asks 'What patterns satisfy this data?'". Furthermore, "what makes an insight actionable [...] is its predictive power". Data science uses techniques such as data management, machine learning and statistics to derive mathematical models which can be "acted upon with high degree of confidence"

(Dhar, 2013). It is widespread in fields such as marketing, advertising, medical diagnosis, finance or social sciences. To understand its role and where it can be incorporated in solid mechanics and computational mechanics in particular, we must take a look back at the anatomy of a mechanical problem. As aforementioned in Section 1.1, a mechanical BVP is essentially driven by three groups of governing equations: (i) kinematics together with Dirichlet boundary conditions, which derive from geometry, (ii) equilibrium and Neumann conditions, which results from conservation laws, (iii) constitutive models, which represent the behavior of the material. On the first hand, the first two are universally valid, or uncertainty-free, as they rely on general physical principles. On the other hand, the constitutive equations, or material laws, are more questionable. More importantly, this is where computational mechanics *needs data* to calibrate the models. A typical application of data science in the field of computational mechanics is then the modeling of the material response: there, one does need a *predictive* mathematical relation that is based on *knowledge* extracted from *unstructured data*, to fall back on the definition given by Dhar (2013).

2.3 Instances of data science in material modeling and structural computation

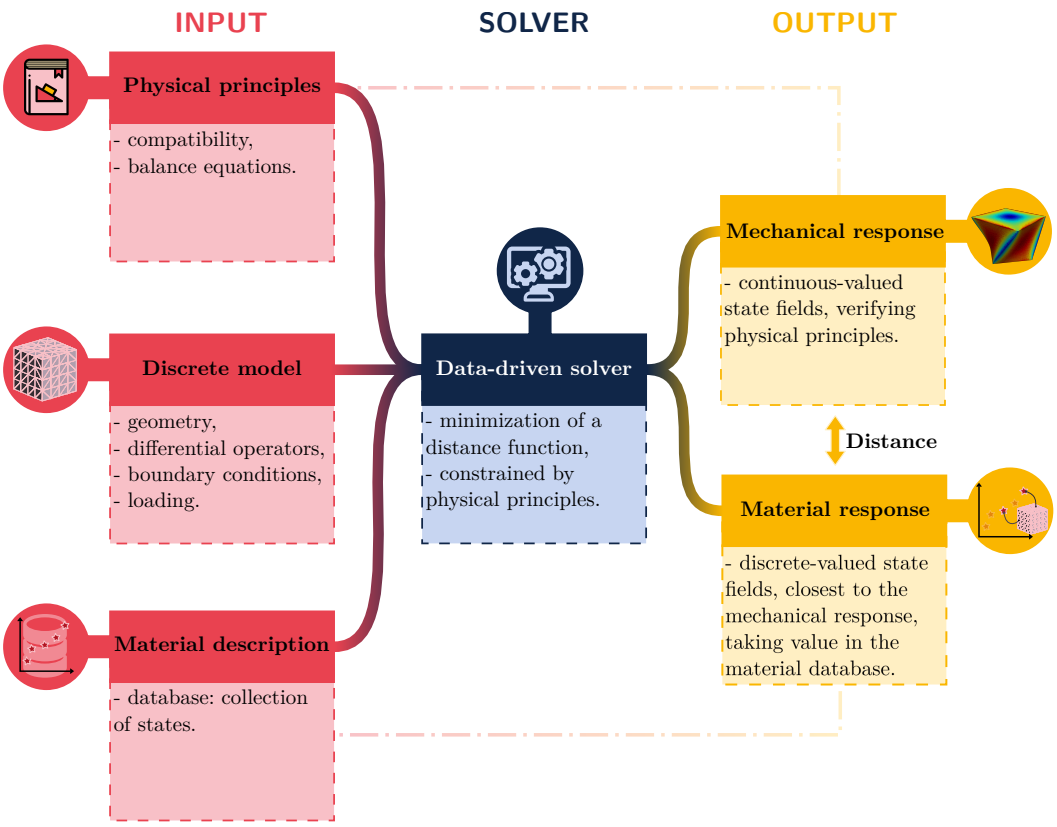
As aforementioned, micro-mechanics of materials is a typical field of applications where computational mechanics actually generates data. It is then a particularly adequate candidate for the use of data science techniques. A first example may be the use of model reduction such as the proper orthogonal decomposition method. It has notably been used by Yvonnet and He (2007) for both localization and homogenization of hyperelastic composites. Other examples can be found in the previously mentioned review of predictive nonlinear theories from multiscale modeling of heterogeneous materials by Matouš et al. (2017). Reduced order models may be understood as *meta*-models of both the material and the structural response in a given problem. Other approaches consists in calibrating a response surface model of the microstructure with digital databases or discrete material maps (see *inter alia* Le et al. (2015) for hyperelastic materials). In these approaches, a constitutive equation is still to be derived beforehand and further calibrated with microstructural data. To directly extract knowledge from the microscopic response, Neural Network (NN)-based material models where first introduced by Ghaboussi J. et al. (1991). Hashash et al. (2004) discussed their implementations in FE analysis and Ling et al. (2016) "incorporated domain knowledge in the machine learning process" by taking into account invariance and symmetries of the problem. One of the most popular and fairly simple machine learning technique is probably dimensional reduction, that seek meaningful low-dimensional structures hidden in high dimensional data, represented for instance by full-field experimental measurements or microscopic simulations. In their review, Matouš et al. (2017) classify Laplacian eigenmaps or Locally linear embedding (LLE) (Roweis and Saul, 2000) as local techniques and kernel-PCA, diffusion maps, or Isomap as global techniques to learn the

manifold where the microscopic response lies. Other applications of manifold learning techniques have recently been introduced for homogeneous materials by Ibañez et al. (2017, 2018), where the constitutive manifold is learned by LLE. For hyperelastic materials, Latorre and Montáns (2014) introduced What-You-Prescribe-is-What-You-Get hyperelastic models where the strain energy density function is constructed with splines based on the resolution of the equilibrium equations of different experiments (see also Crespo et al. (2017)). As a final example of data science integration into computational mechanics, we care to mention the work of Buffiere et al. (2006); Herbig et al. (2011); Rovinelli et al. (2018). It relies on a careful generation of experimental big-data of the evolution of fatigue cracks relative to the local microstructure during *in situ* loading of polycrystalline materials. A Bayesian network is then used to identify an analytical relationship for the crack driving force. In their work, machine learning does not only automatically infer a constitutive relation but is also used as a tool to understand the correlations between the direction of propagation and usual fatigue metrics.

In all the aforementioned instances of data science in computational mechanics or material modeling (which can hardly be taken for an exhaustive list), some sort of constitutive model is still either learned, or postulated and calibrated with relevant techniques and data. We see it as a loss of data: the constitutive model acts as a bottle neck between the upstream data (obtained either experimentally or computationally) and the downstream data generated by FE analysis. As a new paradigm, so-called Data-Driven Computational Mechanics (DDCM), Kirchdoerfer and Ortiz (2016) recently proposed to incorporate the data *directly* into the computation, *i.e.* to replace the constitutive equation with a database of material relevant constitutive variables. In the case of elasticity, the material database is a discrete set of strain-stress data points. This model-free approach roughly consists in minimizing some deviation function to the discrete material response, represented by the material database, under the constraints that the mechanical state in the body satisfies universal physical principles. Their proposition constitutes the basis of this thesis and is detailed in the next chapter.

Chapter II

Data-driven computational mechanics



Contents

1 Presentation of the method

In this section, we present the DDCM approach, first introduced by Kirchdoerfer and Ortiz (2016) and complemented in Conti et al. (2018). We first recall the general idea in Section 1.1. Then, both continuous and FE discrete formulations are presented in Section 1.2. The subsequent data-driven solver is listed in Section 1.3. Finally, we reproduce the original results of Kirchdoerfer and Ortiz (2016) in Section 1.4.

1.1 General idea

As motivated in State of the art, the DDCM approach was proposed by Kirchdoerfer and Ortiz (2016) as a "new computing paradigm", which allows for "bypassing the empirical material modeling step of conventional computing altogether". In lieu of the constitutive model, the material behavior is represented by a database of *material data points*, resulting from experimental acquisition or numerical simulations at finer scales. In solid mechanics, the typical constitutive variables involved are the strain and the stress. For elastic materials, in the linearized kinematics approximation, the material database is then constituted with $(\boldsymbol{\varepsilon}, \boldsymbol{\sigma})$ strain-stress *state* pairs. The approach relies on the fundamental separation between the characterization of the material response and the satisfaction of essential constraints and conservation laws: while the former is most likely to be approximately captured (*e.g.* due to noise in the acquisition of experimental data), the latter ones must be exactly satisfied. The authors then developed a new class of solvers, called *data-driven* solvers, in a small strain elastic framework, which allows for some discrepancy with respect to the material response while satisfying physical principles at every material point in the body. This is achieved by re-formulating the BVP as a minimization of a distance function between two types of strain-stress fields: one, representing the material response, takes value in the material database, and the other, verifying both compatibility and equilibrium equations, represents the mechanical states of the body. The former strain-stress field is referred to as the *material state* and belongs to a so-called *material data set*, while the latter *mechanical state* belongs to a so-called *constraint set*. Both sets are subspaces of the so-called *phase space*, which collects all possible strain-stress fields. The phase space can be opposed to the physical space, where the body deforms and the coordinates of the material points are defined. The data-driven solver then seeks to assign to each material point in the body a strain-stress state that (i) verifies the compatibility and equilibrium constraints of the BVP in physical space, and (ii) is the closest possible from a pre-specified material database in phase space.

The approach was first introduced as a way to directly incorporate *raw* data into the computations, hence developed from a purely numerical, computational mechanics point of view. Specifically, Kirchdoerfer and Ortiz (2016) formulated the data-driven problem in the case of the static equilibrium of nonlinear three-dimensional trusses and of FE discretized linear elastic solids. However data-driven small strain elasticity can also be formulated for continuum mechanics (see the

authors attempt in the last section of Kirchdoerfer and Ortiz (2016)). It is only two years later that Conti et al. (2018) complemented the data-driven paradigm with a continuous formulation. They also provided the attendant mathematical proofs of the existence of solutions together with specific notions of convergence with respect to the material data set. We also care to mention the variational formulation presented in Nguyen et al. (2020).

With the benefit of hindsight, we therefore choose in the next section not to present the method in a chronological manner: we rather first introduce the data-driven paradigm with the continuum mechanics formulation. Then, we summarize the mathematical ground to the approach before recalling the original discrete formulation.

1.2 Continuous and discrete formulation

1.2.1 Continuous formulation

In this section and throughout the manuscript when it is possible, we adopt the formalism of Conti et al. (2018) together with the notations introduced in State of the art. Let us consider an elastic body occupying a domain $\Omega \in \mathbb{R}^3$ under quasi-static loading. The small-strain assumption is adopted; the compatibility and equilibrium governing equations respectively write

$$\boldsymbol{\varepsilon}(\boldsymbol{x}) = \frac{1}{2}\left(\boldsymbol{\nabla}\boldsymbol{u}(\boldsymbol{x}) + \boldsymbol{\nabla}^T\boldsymbol{u}(\boldsymbol{x})\right) \qquad \text{in } \Omega, \tag{II.1a}$$

$$\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{u}_D(\boldsymbol{x}) \qquad \text{on } \Gamma_D, \tag{II.1b}$$

and

$$\operatorname{div}\boldsymbol{\sigma}(\boldsymbol{x}) + \boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0} \qquad \text{in } \Omega, \tag{II.2a}$$

$$\boldsymbol{\sigma}(\boldsymbol{x})\boldsymbol{n}(\boldsymbol{x}) = \boldsymbol{t}(\boldsymbol{x}) \qquad \text{on } \Gamma_N. \tag{II.2b}$$

In elasticity, the above system of equations is classically closed by a constitutive relation of the type

$$\boldsymbol{\sigma}(\boldsymbol{x}) = \check{\boldsymbol{\sigma}}(\boldsymbol{\varepsilon}(\boldsymbol{x})). \tag{II.3}$$

As we previously mentioned, these three groups of governing equations are of two types: universally valid physical principles on the one hand (Eqs. (II.1) and (II.2)), and material-dependent relations on the other hand (Eq. (II.3)). In the data-driven approach, this separation is rooted in the re-formulation of the BVP as distance-minimization problem, presented in the following. Note that the symmetry of the stress tensor is admitted as classically, the conservation of angular momentum is then directly enforced in the definition of the stress field.

Preliminary definitions The strain-stress tensor pair $(\boldsymbol{\varepsilon}(\boldsymbol{x}), \boldsymbol{\sigma}(\boldsymbol{x}))$ is referred to as the *local state* at the material point $\boldsymbol{x}$ of the body Ω. All possible strain-stress pairs sit in a high dimensional space, which is referred to as the *local phase*

space and denoted $\mathcal{Z}_{\text{loc}} = \mathbb{R}^{3\times3}_{\text{sym}} \times \mathbb{R}^{3\times3}_{\text{sym}}$. For instance, in a 3-dimensional problem, the strain and stress tensors each comprise 6 independent variables, which yields a 12-dimensional local phase space. To relate the *physical space* Ω, where the body deforms, to the local phase space, Conti et al. (2018) introduce the *local state function* z as the mapping from every material point $\boldsymbol{x}$ to its local state in $\mathcal{Z}_{\text{loc}}$:

$$\begin{aligned} z :\ & \Omega \mapsto \mathcal{Z}_{\text{loc}} \\ & \boldsymbol{x} \to z(\boldsymbol{x}) = (\boldsymbol{\varepsilon}(\boldsymbol{x}), \boldsymbol{\sigma}(\boldsymbol{x}))\,. \end{aligned} \tag{II.4}$$

We also refer to the local state function as the local state field. The *global phase space* $\mathcal{Z}$ is then merely the collection of all possible local state functions $z : \Omega \to \mathcal{Z}_{\text{loc}}$:

$$\mathcal{Z} = L^2(\Omega, \mathbb{R}^{n\times n}_{\text{sym}}) \times L^2(\Omega, \mathbb{R}^{n\times n}_{\text{sym}}). \tag{II.5}$$

Two different subsets of $\mathcal{Z}$ are defined in the following, dividing material-independent quantities and constitutive quantities.

1. The collection of local state fields verifying universally valid Eqs. (II.1) and (II.2) defines a material-independent *constraint set*, denoted $\mathcal{E}$:

   $$\mathcal{E} = \{z \in \mathcal{Z} \mid \text{Eqs. (II.1) and (II.2)}\} \subset \mathcal{Z}, \tag{II.6}$$

   where the symbol $\mid$ means "subject to" or "such that" throughout the manuscript. We remark that kinematics and equilibrium constraints typically define a continuous *manifold* in phase space.

2. The material behavior is described by a collection of admissible strain-stress pairs, referred to as the *local material data set* $\mathcal{D}_{\text{loc}}$. We also refer to $\mathcal{D}_{\text{loc}}$ as the *material database*. If a constitutive model is known, then the local material data set writes

   $$\check{\mathcal{D}}_{\text{loc}} = \{z(\boldsymbol{x}) \mid \text{Eq. (II.3)}\}\,. \tag{II.7}$$

   However, the data-driven paradigm aims to address cases where the constitutive model Eq. (II.3) is not explicit nor identified; only a raw discrete sampling of the material response is available, obtained *e.g.* through experimental tests or numerical simulations at finer scales. The material data set $\mathcal{D}_{\text{loc}}$ then collects a finite number m of material strain-stress pairs and writes:

   $$\mathcal{D}_{\text{loc}} = \{(\boldsymbol{\varepsilon}_i, \boldsymbol{\sigma}_i) \mid i \in [1\,..\,m]\} \subset \mathcal{Z}_{\text{loc}}. \tag{II.8}$$

   The *global material data set* $\mathcal{D} \subset \mathcal{Z}$ is, in any case, defined by the collection of local state fields taking value in a local material data set:

   $$\mathcal{D} = \{z \in \mathcal{Z} \mid z(\boldsymbol{x}) \in \mathcal{D}_{\text{loc}}\}\,. \tag{II.9}$$

   The local and global definitions of the material data set allow for the possible inhomogeneity of the material: a distinct local material data set can be associated to different phases of an heterogeneous material.

The above definitions are illustrated in Fig. II.1, where the phase space is schematically represented in two dimensions (it is of much higher dimension as previously emphasized). In the small strain setting and for loading conditions independent of the motion, the constraint set is a linear manifold (*i.e.* a line in 2D, in red) resulting from the linearity of the compatibility and equilibrium equations Eqs. (II.1a) and (II.2a). The local material data set is represented here as a discrete sampling of a nonlinear constitutive relation.

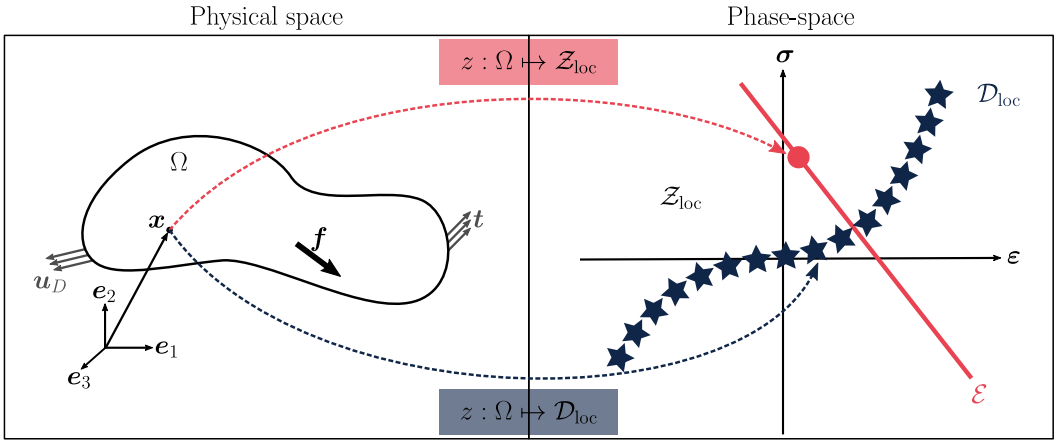


Figure II.1 – Functional spaces defined in the continuous data-driven formulation of Conti et al. (2018).

Reformulation of the boundary value problem The solution of the classical BVP Eqs. (II.1) to (II.3) is defined as the tuple $(\boldsymbol{u}, \boldsymbol{\varepsilon}, \boldsymbol{\sigma})$ of the fields verifying all three equations, at the same time. With the definitions above, this statement can be reformulated as follows: the solution of the classical BVP lies at the intersection $\check{\mathcal{S}}$ between the constraint set of mechanically admissible states and the material data set, defined with a constitutive model:

$$\check{\mathcal{S}} = \mathcal{E} \cap \check{\mathcal{D}}, \tag{II.10}$$

with $\hat{\mathcal{D}} = \left\{ z \in \mathcal{Z} \mid z(\boldsymbol{x}) \in \hat{\mathcal{D}}_{\text{loc}} \right\}$.

Let us now consider the more general case where the material database is discrete, as in Eq. (II.8). Then, the solution set can no longer be determined as previously, since the intersection $\mathcal{E} \cap \mathcal{D}$ is most likely to be empty, as illustrated in Fig. II.2, even if a solution does exist. The data-driven approach then boils down to relaxing this characterization by allowing a certain amount of discrepancy between the solution set and the material data set. The solution is now defined as the *mechanical state* field $z = (\boldsymbol{\varepsilon}, \boldsymbol{\sigma}) \in \mathcal{E}$, verifying both kinematics and equilibrium constraints, that is closest to the material data set:

$$\mathcal{S} = \arg\min_{z \in \mathcal{E}} d(z, \mathcal{D}), \tag{II.11}$$

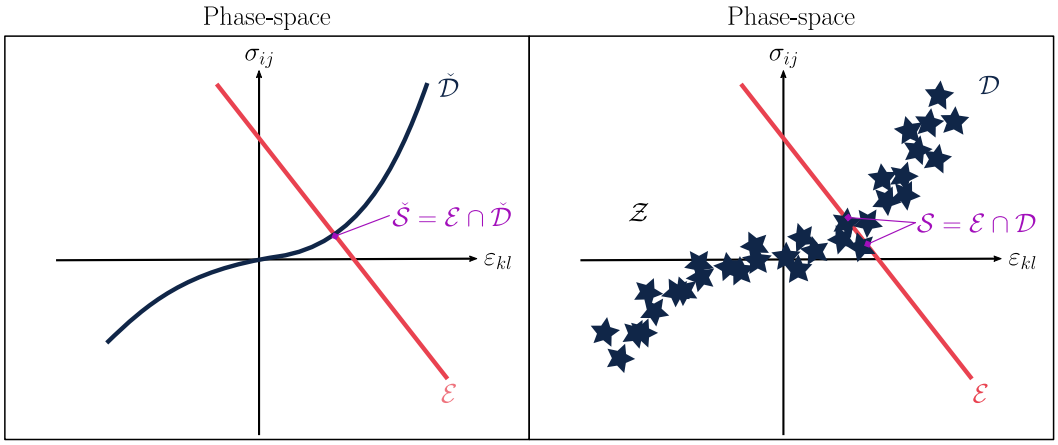


Figure II.2 – Exact and near intersections $\mathcal{S}$ between a possible constraint set $\mathcal{E}$ and (left) a continuous material data set $\check{\mathcal{D}}$, obtained from a constitutive model and (right) a discrete material data set $\mathcal{D}$, obtained from *e.g.* experimental testing. Inspired from Kirchdoerfer and Ortiz (2016).

where d is an appropriate distance defined on the phase space $\mathcal{Z}$. The phase space $\mathcal{Z} = L^2\left(\Omega, \mathbb{R}^{n\times n}_{\text{sym}}\right) \times L^2\left(\Omega, \mathbb{R}^{n\times n}_{\text{sym}}\right)$ is indeed equipped with the metric:

$$\|z\|_{\mathbb{C}} = \left(\int_\Omega \left(\frac{1}{2}(\mathbb{C} : \boldsymbol{\varepsilon}) : \boldsymbol{\varepsilon} + \frac{1}{2}(\mathbb{C}^{-1} : \boldsymbol{\sigma}) : \boldsymbol{\sigma} \right) \mathrm{d}v \right)^{\frac{1}{2}}, \tag{II.12}$$

where $\mathbb{C}$ is a constant fourth-order positive definite tensor. The distance between a local state z and any subset $\mathcal{A}$ of the phase space is follows as

$$d(z, \mathcal{A}) = \min_{a\in\mathcal{A}} \|z - a\|_{\mathbb{C}}. \tag{II.13}$$

The data-driven BVP is then reformulated as the double minimization problem

$$\min_{z\in\mathcal{E}} \min_{z'\in\mathcal{D}} \|z - z'\|_{\mathbb{C}}, \tag{II.14}$$

or, equivalently, using the squared distance, as

$$\min_{z\in\mathcal{E}} \min_{z'\in\mathcal{D}} \|z - z'\|^2_{\mathbb{C}}. \tag{II.15}$$

For a given material data set $\mathcal{D}$, the solution set $\mathcal{S}$ of the data-driven $\mathcal{D}$-problem is then defined as

$$\mathcal{S} = \arg \min_{z=(\boldsymbol{\varepsilon},\boldsymbol{\sigma})\in\mathcal{E}} \min_{z'=(\boldsymbol{\varepsilon}',\boldsymbol{\sigma}')\in\mathcal{D}} \int_\Omega \left(\frac{1}{2} \left(\mathbb{C} : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}')\right) : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}') + \frac{1}{2} \left(\mathbb{C}^{-1} : (\boldsymbol{\sigma} - \boldsymbol{\sigma}')\right) : (\boldsymbol{\sigma} - \boldsymbol{\sigma}') \right) \mathrm{d}v. \tag{II.16}$$

The material part $z' \in \mathcal{D}$ of the solution to Eq. (II.16) is referred to as the *material state* field.

An elementary example, presented in Kirchdoerfer and Ortiz (2017), illustrates the data-driven distance-minimizing paradigm: an elastic bar deforms uniformly under the action of a loading device. We refer to Analytical solution for a one-dimensional bar for the analytical data-driven solution and the attendant geometrical interpretation we derived.

1.2.2 Mathematical proof

In their first contribution, Kirchdoerfer and Ortiz (2016) showed that "the data-driven solutions converge to classical solutions when the data set approximates a limiting constitutive law with increasing fidelity". They particularly developed estimates of convergence rates for the data-driven solution to the reference solution, with respect to the number of states in the material database. In addition, they ensured that the convergence is preserved with spatial discretization (e.g. through FE approximation) provided that the "fidelity of the data set increases appropriately with increasing mesh resolution".

A stronger mathematical proof that the problem Eq. (II.11) is well-posed and recovers the classical solution in the case of linear elasticity is provided by Conti et al. (2018). Indeed, they demonstrate that the data-driven $\mathcal{D}_{\mathbb{C}}$-problem defined as

$$\min_{z \in \mathcal{E}} d\left(z, \mathcal{D}_{\mathbb{C}}\right) \tag{II.17a}$$

where

$$\mathcal{E} = \{z \in \mathcal{Z} \mid \text{Eqs. (II.1) and (II.2)}\} \tag{II.17b}$$

$$\mathcal{D}_{\mathbb{C}} = \{(\boldsymbol{\varepsilon}, \boldsymbol{\sigma}) \in \mathcal{Z} \mid \boldsymbol{\sigma} = \mathbb{C} : \boldsymbol{\varepsilon}\} \tag{II.17c}$$

has a unique solution $(\boldsymbol{\varepsilon}, \boldsymbol{\sigma})$ which coincides with the classical linear elastic solution

$$\boldsymbol{\sigma} = \mathbb{C} : \boldsymbol{\varepsilon}. \tag{II.18}$$

See Conti et al. (2018, Theorem 2.2) for more details.

Furthermore, considering a sequence $(\mathcal{D}_h)$ of material data sets which converges to a limiting material data set $\mathcal{D}$, the authors show that the solutions of the $\mathcal{D}_h$-problems converge to the solution of the $\mathcal{D}$-problem. The data-driven $\mathcal{D}$-problem consists of finding

$$\arg\min_{z \in \mathcal{E}} \; d^2\left(z, \mathcal{D}\right) \tag{II.19}$$

or, equivalently,

$$\arg\min_{z \in \mathcal{Z}} \; \left(d^2\left(z, \mathcal{D}\right) + I_{\mathcal{E}}(z)\right) \tag{II.20}$$

where the indicator function $I_{\mathcal{E}}$ of $\mathcal{E} \subset \mathcal{Z}$ is defined as

$$I_{\mathcal{E}}(z) = \begin{cases} 0, & \text{if } z \in \mathcal{E}, \\ \infty, & \text{otherwise.} \end{cases} \tag{II.21}$$

The convergence of $(\mathcal{D}_h)$ is understood in terms of spread away from the limiting material data set, and in terms of density of material data points, as illustrated in Fig. II.3(a). It is detailed in (Conti et al., 2018, Lemma 2.11):

> Let $\mathcal{Z}$ and $\mathcal{E}$ be as in Eqs. (II.5) and (II.6). Suppose that
>
> $$\mathcal{D}_h = \{z \in \mathcal{Z} \mid z(\boldsymbol{x}) \in \mathcal{D}_{\mathrm{loc},h} \text{ almost everywhere in } \Omega\},$$
>
> for some sequence of local material data sets $\mathcal{D}_{\mathrm{loc},h} \subset \mathbb{R}^{n\times n}_{\mathrm{sym}} \times \mathbb{R}^{n\times n}_{\mathrm{sym}}$. Let
>
> $$\mathcal{D} = \{z \in \mathcal{Z} \mid z(\boldsymbol{x}) \in \mathcal{D}_{\mathrm{loc}} \text{ almost everywhere in } \Omega\},$$
>
> where
>
> $$\mathcal{D}_{\mathrm{loc}} = \left\{(\boldsymbol{\varepsilon}, \boldsymbol{\sigma}) \in \mathbb{R}^{n\times n}_{\mathrm{sym}} \times \mathbb{R}^{n\times n}_{\mathrm{sym}} \mid \boldsymbol{\sigma} = \mathbb{C} : \boldsymbol{\varepsilon}\right\}.$$
>
> Assume that
>
> i) (Fine approximation) There is a sequence $\rho_h \downarrow 0$ such that
>
> $$d(\xi, \mathcal{D}_{\mathrm{loc},h}) \leq \rho_h, \ \forall \xi \in \mathcal{D}_{\mathrm{loc}};$$
>
> ii) (Uniform approximation) There is a sequence $t_h \downarrow 0$ such that
>
> $$d(\xi, \mathcal{D}_{\mathrm{loc}}) \leq t_h, \ \forall \xi \in \mathcal{D}_{\mathrm{loc},h};$$
>
> sequentially
>
> Then, $\mathcal{D} = M - \lim_{h\to\infty} \mathcal{D}_h$ in $\mathcal{Z}$.

The limit operator $M - \lim$ denotes the Mosco convergence of sets, defined in Conti et al. (2018, Definition 2.5, 2.6). Once the convergence of the sequence $(\mathcal{D}_h)$ to the limiting material data set $\mathcal{D}$ is assumed, the convergence of the sequence (z_h) of associated data-driven solutions follows (see Conti et al. (2018, Theorem 2.8)):

> Let $\mathcal{Z}$ be a reflexive, separable Banach space, $\mathcal{D}$ and $(\mathcal{D}_h)$ subsets of $\mathcal{Z}$, $\mathcal{E}$ a weakly sequentially closed subset of $\mathcal{Z}$. Suppose:
>
> i) (Mosco convergence) $\mathcal{D} = M - \lim_{h\to\infty} \mathcal{D}_h$ in $\mathcal{Z}$ and
>
> ii) (Equi-transversality) There are constants $c > 0$ and $b \geq 0$ such that, for all $y \in \mathcal{D}_h$ and $z \in \mathcal{E}$,
>
> $$\|y - z\|_{\mathbb{C}} \geq c\,(\|y\|_{\mathbb{C}} + \|z\|_{\mathbb{C}}) - b.$$

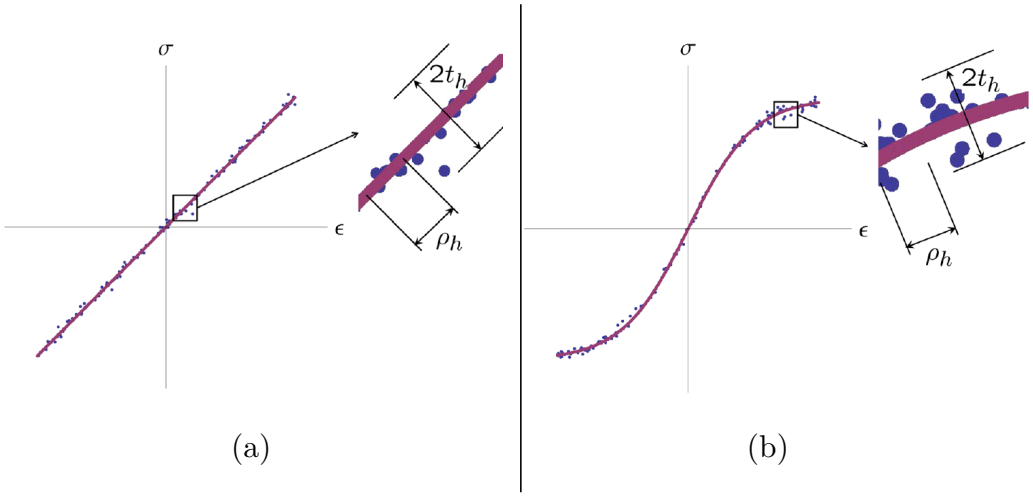


Figure II.3 – Schematic of convergent sequence of local material data sets $\mathcal{D}_{\text{loc},h}$. The parameter t_h controls the spread of the material data sets away from the limiting data set and the parameter ρ_h controls the density of material data point. (a) Linear elastic graph. (b) Nonlinear elastic graph. From Conti et al. (2018).

Then,

$$I_{\mathcal{E}}(\cdot) + d^2(\cdot, \mathcal{D}) = \Gamma - \lim_{h\to\infty} \left(I_{\mathcal{E}}(\cdot) + d^2\left(\cdot, \mathcal{D}_h\right)\right)$$

If (z_h) is a sequence of elements of $\mathcal{Z}$ with $\sup_h \; I_{\mathcal{E}}(z_h) + d^2(z_h, \mathcal{D}) < \infty$ then there is a subsequence converging weakly to some $z \in \mathcal{E}$.

In the above Theorem, the limit operator $\Gamma - \lim$ denotes the $\Gamma-$convergence of sequence of functionals in a topological space, defined in Conti et al. (2018, Definition 2.3). This Theorem states that the convergence of the material data set $\mathcal{D}_h$ to $\mathcal{D}$ and the equi-transversality condition are sufficient to ensure the convergence of the solution z_h to z of the corresponding data-driven $\mathcal{D}_h$-problem and $\mathcal{D}$-problem, respectively. A corollary is that any data-driven $\mathcal{D}$-problem in these conditions has solutions (see Conti et al. (2018, Corollary 2.9)), provided that the material data set $\mathcal{D}$ and the constraint set $\mathcal{E}$ are weakly sequentially closed[1]. This holds in particular for closed convex data sets, including linear subspaces of $\mathcal{Z}$, *i.e.* collection of linear elastic graphs. An analogous convergence theorem is derived in Conti et al. (2018, Section 3.1) for cases where the material set $\mathcal{D}$ fails to be weakly closed, which is the case for instance in nonlinear elasticity. It calls for the convergence of sequences $(\mathcal{D}_h)$ of material data sets and sequences $(y_h) \in \mathcal{E}$ of compatible and balanced local states simultaneously to ensure the convergence of the solution $z_h \in \mathcal{D}_h \cap \mathcal{E}$ to $z \in \mathcal{D} \cap \mathcal{E}$. Again, the convergence of the material data sets is understood in

[1] closed in weak topology: given any convergent sequence of points in the subset, every limit of the sequence lies inside the subset.

terms of fine (density) and uniform (spread away) approximations, as illustrated in Fig. II.3(b).

From an engineering point of view, the most important results to retain from the work of Conti et al. (2018), are, in our sense:

- the data-driven framework encompasses the classical linear elastic formulation, since the unique solution to the $\mathcal{D}_{\mathbb{C}}$-problem is the linear elastic solution Eq. (II.18);
- there is a solid mathematical ground proving that the better the discrete material data set $\mathcal{D}_h$ approximates the limiting (linear) elastic graph $\mathcal{D}$, the closer the data-driven solution to the classical (linear) elastic solution. In that sense, numerical data-driven schemes are expected to converge to the reference solution obtained with a constitutive model, as the material data set increasingly approximates the constitutive model.

1.2.3 Discrete formulation

As aforementioned, the data-driven approach was first introduced in a more computational manner by Kirchdoerfer and Ortiz (2016), through a finite dimensional[2] numerical formulation we briefly recall in this section.

Truss structures First, the data-driven problem is formulated for truss structures. Let us consider a truss structure of M bars and N degrees of freedom, subject to forces $\{f_i\}_{i=1}^{N}$. In this one-dimensional setting, only the longitudinal strain ε_e and stress σ_e are considered in each bar member e. The local phase space $\mathcal{Z}_{\text{loc}}^{(e)} = (\varepsilon_e, \sigma_e)$ is then conveniently reduced to a part of the $\mathbb{R}^2$ plane. The global phase space is now the finite product set

$$\mathcal{Z} = \mathcal{Z}_{\text{loc}}^{(1)} \times \cdots \times \mathcal{Z}_{\text{loc}}^{(M)}. \tag{II.22}$$

The global state of the truss is then represented by a point $(\varepsilon, \sigma) = \{(\varepsilon_e, \sigma_e)\}_{e=1}^{M}$ in $\mathcal{Z}$.

The governing equations for the structure write

$$\varepsilon_e = \sum_{e=1}^{M} B_{ei} u_i, \qquad \forall e \in [1\,..\,M], \tag{II.23a}$$

$$\sum_{e=1}^{M} w_e B_{ei} \sigma_e - f_i = 0, \qquad \forall i \in [1\,..\,N], \tag{II.23b}$$

where $\{u_i\}\, i = 1^N$ is the array of displacement degrees of freedom, the matrix $\boldsymbol{B}$ encodes the connectivity and geometry of the truss (*i.e.* the derivatives of the shape

[2] as opposed to the functional spaces of the continuum formulation which are of infinite dimension.

functions in each bar) and $w_e = A_e L_e$ is the volume of the bar with cross-sectional area A_e and length L_e. This governing equations re-define the constraint set as

$$\mathcal{E} = \{(\varepsilon, \sigma) \in \mathcal{Z} \mid \text{Eq. (II.23)}\}\,. \tag{II.24}$$

Let now assume that a local material data set $\mathcal{D}_{\text{loc}}^{(e)} = \{(\varepsilon'_i, \sigma'_i) \mid i \in [1\,..\,m_e]\}$ of m_e strain-stress pairs collects a sampling of the material response of each bar, as provided by uniaxial tensile tests performed *a priori* on the material for instance. The global material data set is now the collection of the M local data sets:

$$\mathcal{D} = \mathcal{D}_{\text{loc}}^{(1)} \times \cdots \times \mathcal{D}_{\text{loc}}^{(M)}. \tag{II.25}$$

The objective of the data-driven solver is to find the global state $(\varepsilon, \sigma) \in \mathcal{E}$ of the truss which verifies Eq. (II.23), while being as close as possible to the material data set. This is achieved by formulating a penalty function as

$$F = \sum_{e=1}^{M} w_e F_e(\varepsilon_e, \sigma_e), \tag{II.26}$$

where the local deviation to the database is defined in local phase space as

$$F_e = \min_{(\varepsilon'_e, \sigma'_e) \in \mathcal{D}_{\text{loc}}^{(e)}} \left(\frac{1}{2} C_e \left(\varepsilon_e - \varepsilon'_e\right)^2 + \frac{1}{2} C_e^{-1} \left(\sigma_e - \sigma'_e\right)^2 \right), \tag{II.27}$$

with C_e a positive constant. In the end, the data-driven formulation for truss structures consists in a constrained minimization problem of the form

Find $\mathcal{S} = \{(\varepsilon_e, \sigma_e)\}_{e=1}^{M}$ such that

$$\mathcal{S} = \arg \min_{(\varepsilon,\sigma) \in \mathcal{Z}} \sum_{e=1}^{M} w_e\, F_e(\varepsilon_e, \sigma_e), \tag{II.28a}$$

subject to

$$\varepsilon_e = \sum_{i=1}^{N} B_{ei} u_i; \quad \sum_{e=1}^{M} w_e B_{ei} \sigma_e - f_i = 0. \tag{II.28b}$$

As one can see, this data-driven formulation for truss structures is directly related to the continuous one recalled in Section 1.2.1: the compatibility and (the weak form of) equilibrium equations have merely been evaluated in a finite dimensional physical space (finite collection of bar elements). Likewise, the deviation function Eq. (II.26) results from a discrete integration of the square of the continuous norm Eq. (II.12), with constant strain and stress states in each bar element. Note that the positive scalar constant C_e plays the same role as the reference stiffness tensor $\mathbb{C}$ in Eq. (II.12).

Elastic solids Second, Kirchdoerfer and Ortiz (2016) formulated the approach for general geometrically linear elastic problems, where the strain-stress state is now evaluated at the integration points of a FE mesh. They considered a FE model of a nonlinear elastic solid in the linearized kinematics approximation. The mesh is composed of M integration points and N nodes. The local state $z_e = (\boldsymbol{\varepsilon}_e, \boldsymbol{\sigma}_e)$ of integration point e now belong to the 12-dimensional local phase space $\mathcal{Z}_{\text{loc}} = \mathbb{R}^{3\times3}_{\text{sym}} \times \mathbb{R}^{3\times3}_{\text{sym}}$. The material response of each integration point is then characterized by a material database $\mathcal{D}^{(e)}_{\text{loc}}$, consisting of a finite number of tensor pairs $z'_e = (\boldsymbol{\varepsilon}'_e, \boldsymbol{\sigma}'_e)$. The local penalty function writes

$$F_e = \min_{(\boldsymbol{\varepsilon}'_e, \boldsymbol{\sigma}'_e) \in \mathcal{D}^{(e)}_{\text{loc}}} \left(W_e \left(\boldsymbol{\varepsilon}_e - \boldsymbol{\varepsilon}'_e \right) + W^*_e \left(\boldsymbol{\sigma}_e - \boldsymbol{\sigma}'_e \right) \right), \tag{II.29a}$$

with

$$W_e(\boldsymbol{\varepsilon}_e) = \frac{1}{2} (\mathbb{C} : \boldsymbol{\varepsilon}_e) : \boldsymbol{\varepsilon}_e, \tag{II.29b}$$

$$W^*_e(\boldsymbol{\sigma}_e) = \frac{1}{2} (\mathbb{C}^{-1} : \boldsymbol{\sigma}_e) : \boldsymbol{\sigma}_e. \tag{II.29c}$$

two functionals that may regarded as reference strain and complementary energy densities. We define the local distance $|z_e - z'_e|_{\mathbb{C}}$ in $\mathcal{Z}_{\text{loc}} \times \mathcal{D}^{(e)}_{\text{loc}}$ as

$$|z_e - z^*_e|^2_{\mathbb{C}} = W_e \left(\boldsymbol{\varepsilon}_e - \boldsymbol{\varepsilon}'_e \right) + W^*_e \left(\boldsymbol{\sigma}_e - \boldsymbol{\sigma}'_e \right). \tag{II.30}$$

. Here, the reference stiffness tensor $\mathbb{C}$ is expressed as[3]

$$\mathbb{C} = \lambda \left(\boldsymbol{I} \otimes \boldsymbol{I} \right) + 2\mu \mathbb{I}^S, \tag{II.31}$$

where λ, μ are some numerical parameters and $\mathbb{I}^{\text{SYM}}$ (resp. $\boldsymbol{I} \otimes \boldsymbol{I}$) the symmetric (resp. spherical) fourth-order identity tensor. As for truss structures, the data-driven problem develops into a constrained minimization problem, which penalizes the deviation from the material data set while ensuring the respect of mechanical constraints. It is formulated as follows:

Find $\mathcal{S} = \{(\boldsymbol{\varepsilon}_e, \boldsymbol{\sigma}_e)\}_{e=1}^M$ such that

$$\mathcal{S} = \arg \min_{(\boldsymbol{\varepsilon}, \boldsymbol{\sigma}) \in \mathcal{Z}} \sum_{e=1}^{M} w_e \min_{(\boldsymbol{\varepsilon}'_e, \boldsymbol{\sigma}'_e) \in \mathcal{D}^{(e)}_{\text{loc}}} \left(W_e \left(\boldsymbol{\varepsilon}_e - \boldsymbol{\varepsilon}'_e \right) + W^*_e \left(\boldsymbol{\sigma}_e - \boldsymbol{\sigma}'_e \right) \right) \tag{II.32a}$$

subject to

$$\boldsymbol{\varepsilon}_e = \sum_{a=1}^{N} \boldsymbol{B}_{ea} \boldsymbol{u}_a, \qquad \forall e \in [1 \,..\, M], \tag{II.32b}$$

$$\sum_{e=1}^{M} w_e \boldsymbol{B}^T_{ea} \boldsymbol{\sigma}_e - \boldsymbol{f}_a = 0, \qquad \forall a \in [1 \,..\, N], \tag{II.32c}$$

[3] Note that in Kirchdoerfer and Ortiz (2016), the stiffness parameter is in fact $\frac{1}{2}\mathbb{C}$; the local reference energy densities hence do not comprise the $\frac{1}{2}$ factor. The formulation adopted here is obviously equivalent.

where $\boldsymbol{u} = \{\boldsymbol{u}_a\}_{a=1}^{N}$ denote the nodal displacement vectors, $\{\boldsymbol{f}_a\}_{a=1}^{N}$ are the nodal force vectors, the matrix $\boldsymbol{B}_{ea}$ encodes the connectivity and geometry of the FE mesh and w_e now denotes the integration weight. The solution of the minimization problem Eq. (II.32) is the collection $\{z_e\}_{e=1}^{M} \in \mathcal{E}$ of mechanical states which verify the compatibility and equilibrium equations Eqs. (II.32b) and (II.32c). Concurrently, the collection $\{z_e^*\}_{e=1}^{M} \in \mathcal{D}$ of the closest material data points are referred to as the material states. The material state of integration point e is thus defined as

$$z_e^* = (\boldsymbol{\varepsilon}_e^*, \boldsymbol{\sigma}_e^*) = \arg \min_{(\boldsymbol{\varepsilon}'_e, \boldsymbol{\sigma}'_e) \in \mathcal{D}_{\text{loc}}^{(e)}} \left| z_e - z'_e \right|_{\mathbb{C}}, \ \forall e \in [1 \,..\, M]. \tag{II.33}$$

To conclude this section, we summarize the above definitions and minimization problem as well as we illustrate the relationship between the original FE data-driven formulation of Kirchdoerfer and Ortiz (2016) and the mathematical framework of functional spaces derived in Conti et al. (2018) with the schematic in Fig. II.4.

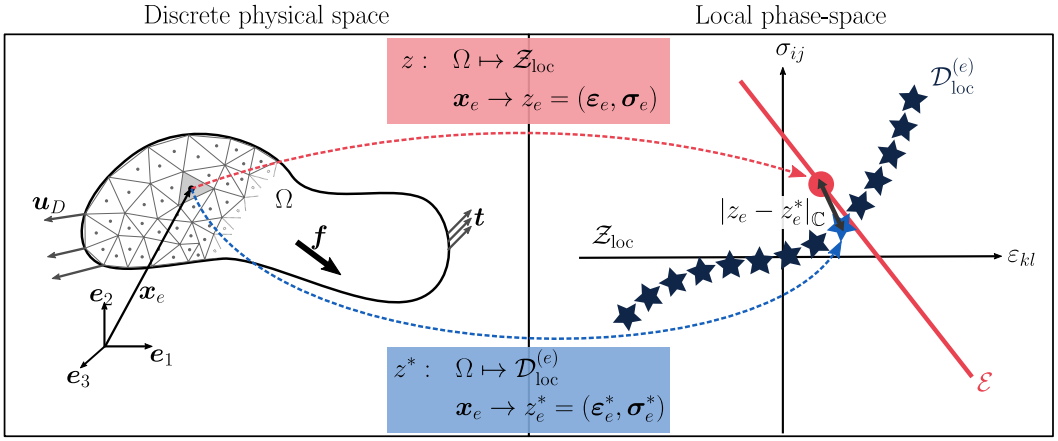


Figure II.4 – Functional spaces of the data-driven FE formulation. The mechanical state z_e (red circle) of the integration point can only move along the constraint set (red solid line). The material state z_e^* (light blue star) of the integration point is the closest material data point selected from the material database $\mathcal{D}_{\text{loc}}^{(e)}$ (dark blue stars), according to the local distance $|z_e - z_e^*|_{\mathbb{C}}$ defined Eq. (II.30).

1.3 Original algorithm

The data-driven problem Eq. (II.32) involves two nested minimization over a continuous-valued functional space $\mathcal{Z}$ and a discrete-valued functional space $\mathcal{D}$. It then mixes a conventional multi-variate constrained optimization problem with a combinatorial optimization. In that sense, it is a difficult problem to solve numerically. Kirchdoerfer and Ortiz (2016) originally proposed a simple heuristic consisting in an alternated minimization illustrated in Fig. II.5: first the material states are fixed and the mechanical states are determined solving the continuous constrained minimization problem; then the material states are updated from a search for nearest

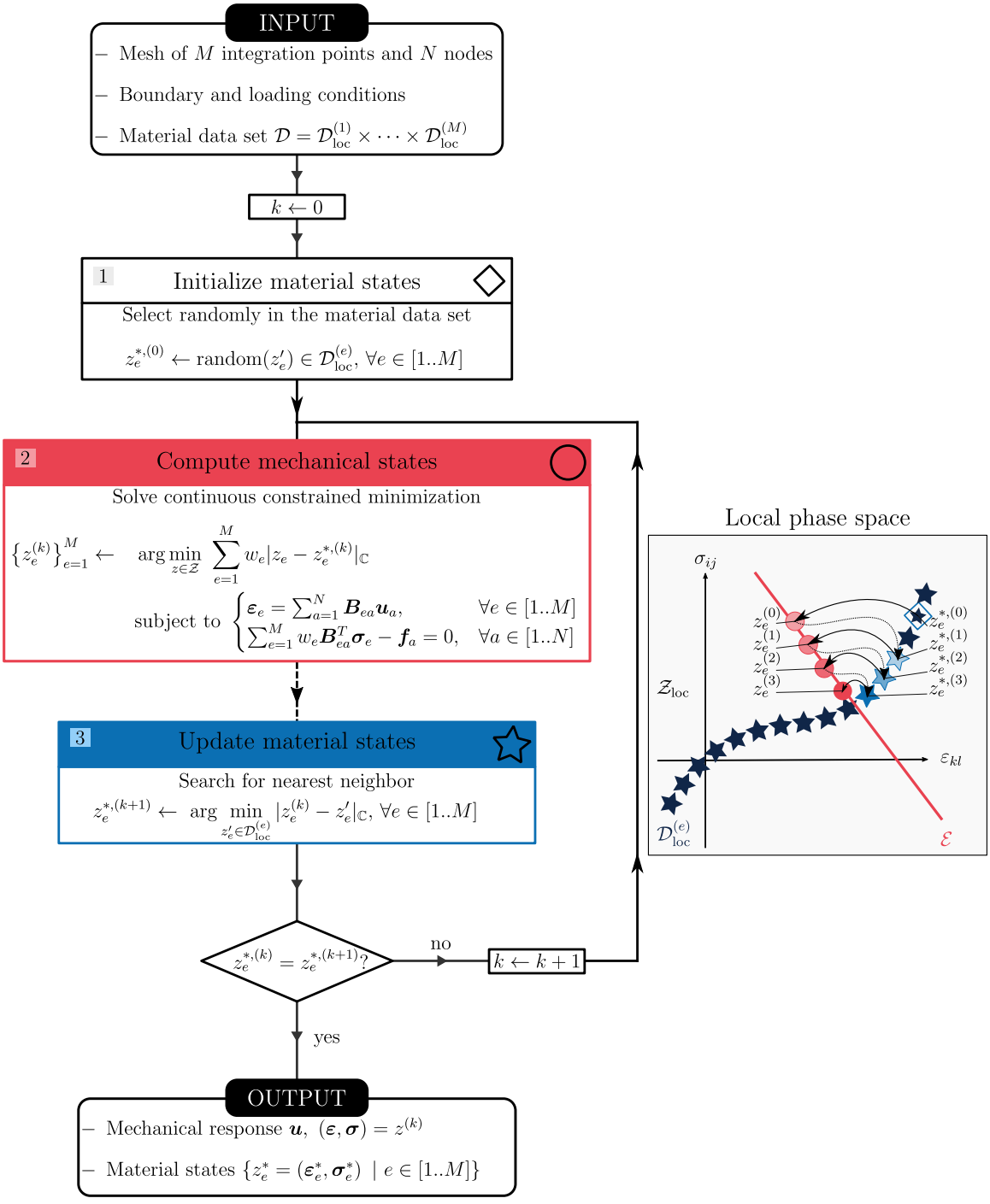


Figure II.5 – Original alternated minimization proposed by Kirchdoerfer and Ortiz (2016). We illustrate a typical local data assignment iteration in the local phase space. The mechanical states (red circles) always lie on the constraint set while the material states (light blue stars) are successively selected from the material data set, according to the local distance $|z_e - z_e^*|_{\mathbb{C}}$ defined Eq. (II.30).

neighbor in the material data set. The algorithm stops when the updated material states are unchanged: the objective function can then no longer improve.

With this approach, the double minimization problem Eq. (II.32) is split into two distinct problems, presented below.

- Let us first consider the stage where the mechanical state

  $$z = \{z_e = (\boldsymbol{\varepsilon}_e, \boldsymbol{\sigma}_e)\}_{e=1}^{M} \in \mathcal{E}$$

  is fixed. The intermediate material solution set $\mathcal{S}^{\text{mat}} \subset \mathcal{D}$ is then determined from Eqs. (II.32a) and (II.33). It results from M independent searches for nearest neighbors in local phase space:

  $$\mathcal{S}^{\text{mat}}(z) = \left\{ \arg \min_{z'_e \in \mathcal{D}^{(e)}_{\text{loc}}} \left| z_e - z'_e \right|_{\mathbb{C}} \;\middle|\; e \in [1 \,..\, M] \right\}. \tag{II.34}$$

- Let us now consider the stage where the material state

  $$z^* = \{z_e^* = (\boldsymbol{\varepsilon}_e^*, \boldsymbol{\sigma}_e^*)\}_{e=1}^{M} \in \mathcal{D}$$

  is determined. The double minimization problem Eq. (II.32) is reduced to a constrained simple minimization problem. The intermediate mechanical solution set $\mathcal{S}^{\text{mec}} \subset \mathcal{E}$ then results from

  $$\mathcal{S}^{\text{mec}}(z^*) = \arg \min_{z \in \mathcal{Z}} \sum_{e=1}^{M} w_e \left| z_e - z_e^* \right|_{\mathbb{C}} \tag{II.35a}$$

  subject to

  $$\boldsymbol{\varepsilon}_e = \sum_{a=1}^{N} \boldsymbol{B}_{ea} \boldsymbol{u}_a, \qquad \forall e \in [1 \,..\, M], \tag{II.35b}$$

  $$\sum_{e=1}^{M} w_e \boldsymbol{B}_{ea}^T \boldsymbol{\sigma}_e - \boldsymbol{f}_a = 0, \qquad \forall a \in [1 \,..\, N]. \tag{II.35c}$$

  The resolution method employed to solve this problem is detailed in the following.

To solve the continuous constrained minimization sub-problem Eq. (II.35), Kirchdoerfer and Ortiz (2016) adopted a Lagrange multipliers method to enforce the equilibrium constraint. The Dirichlet boundary conditions and the compatibility constraint are classically enforced in the FE model by fixing the prescribed nodal displacements and by directly expressing strains in terms of displacements, respectively. Consequently, the stationary problem writes

$$\delta \left[\sum_{e=1}^{M} w_e \left(W_e \left(\sum_{a=1}^{N} \boldsymbol{B}_{ea} \boldsymbol{u}_a - \boldsymbol{\varepsilon}_e^* \right) + W_e^* \left(\boldsymbol{\sigma}_e - \boldsymbol{\sigma}_e^* \right) \right) - \sum_{a=1}^{N} \boldsymbol{\eta}_a \cdot \left(\sum_{e=1}^{M} w_e \boldsymbol{B}_{ea}^T \boldsymbol{\sigma}_e - \boldsymbol{f}_a \right) \right] = 0, \tag{II.36}$$

with $\boldsymbol{\eta} = \{\boldsymbol{\eta}_a\}_{a=1}^N$ the nodal Lagrange multiplier vectors. All possible variations yield the following stationary equations

$$\delta \boldsymbol{u}_a \Rightarrow \sum_{e=1}^{M} w_e \boldsymbol{B}_{ea}^T \left(\mathbb{C} : \left(\sum_{b=1}^{N} \boldsymbol{B}_{eb} \boldsymbol{u}_b - \boldsymbol{\varepsilon}_e^* \right) \right) = \boldsymbol{0}, \qquad \forall a \in [1 \,..\, N], \tag{II.37a}$$

$$\delta \boldsymbol{\sigma}_e \Rightarrow \mathbb{C}^{-1} : (\boldsymbol{\sigma}_e - \boldsymbol{\sigma}_e^*) - \sum_{a=1}^{N} \boldsymbol{B}_{ea} \boldsymbol{\eta}_a = \boldsymbol{0}, \qquad \forall e \in [1 \,..\, M], \tag{II.37b}$$

$$\delta \boldsymbol{\eta}_a \Rightarrow \sum_{e=1}^{M} w_e \boldsymbol{B}_{ea}^T \boldsymbol{\sigma}_e - \boldsymbol{f}_a = 0, \qquad \forall a \in [1 \,..\, N], \tag{II.37c}$$

which are rendered into two standard linear elastic systems of pseudo-stiffness $\mathbb{C}$ of the form

$$\sum_{b=1}^{N} \left(\sum_{e=1}^{M} w_e \boldsymbol{B}_{ea}^T : \mathbb{C}_e : \boldsymbol{B}_{eb} \right) \boldsymbol{u}_b = \sum_{e=1}^{M} w_e \boldsymbol{B}_{ea}^T \left(\mathbb{C}_e : \boldsymbol{\varepsilon}_e^* \right), \qquad \forall a \in [1 \,..\, N], \tag{II.38a}$$

$$\sum_{b=1}^{N} \left(\sum_{e=1}^{M} w_e \boldsymbol{B}_{ea}^T : \mathbb{C}_e : \boldsymbol{B}_{eb} \right) \boldsymbol{\eta}_b = \boldsymbol{f}_a - \sum_{e=1}^{M} w_e \boldsymbol{B}_{ea}^T \boldsymbol{\sigma}_e^*, \qquad \forall a \in [1 \,..\, N]. \tag{II.38b}$$

As Kirchdoerfer and Ortiz (2016) point out, "the displacement problem Eq. (II.38a) is driven by the optimal local strains, whereas the Lagrange multipliers problem Eq. (II.38b) is driven by the out-of-balance forces attendant to the optimal local stresses". Moreover, the two systems share the same stiffness matrix of standard FE form, which can be computed, factorized and stored once and for all at the beginning of the simulation. In addition, from Eq. (II.37b), the mechanical stresses are obtained by a correction of the material stresses as

$$\boldsymbol{\sigma}_e = \boldsymbol{\sigma}_e^* + \sum_{a=1}^{N} \mathbb{C} : (\boldsymbol{B}_{ea} \boldsymbol{\eta}_a), \ \forall e = 1 \,..\, M. \tag{II.39}$$

Consequently, the Lagrange multipliers $\boldsymbol{\eta}$ can be regarded as virtual displacements which, associated to a stiffness $\mathbb{C}$, generate stresses that correct the out-of-balance gap between the material and mechanical stresses and the external forces. The computation of the mechanical states is then divided into two parts:

1. displacements $\boldsymbol{u}$ and Lagrange multipliers $\boldsymbol{\eta}$ result from Eq. (II.38),
2. mechanical strains $\boldsymbol{\varepsilon}$ consequently derive from the compatibility Eq. (II.32b) and mechanical stresses $\boldsymbol{\sigma}$ are obtained from the correction Eq. (II.39).

1.4 Numerical results

In their first contribution, Kirchdoerfer and Ortiz (2016) focused on numerically assessing the convergence of the data-driven solvers to the classical solution, as the

material data set approximates the strain-stress curve increasingly closely, for both a truss structure and a 3D tensile test. In this section, we choose to reproduce almost exhaustively their results, as they provide the specific method of analyzing data-driven results we will adopt in our work.

1.4.1 Truss structures

The analysis of truss structures is a convenient way of discussing the performances of data-driven solvers as it allows for a 2D representation of the material data set. Kirchdoerfer and Ortiz (2016) then first evaluated the data-driven solver for the truss formulation Eq. (II.28). They focused on: (i) the convergence to the classical solution with respect to the number of points for both noise-free and noisy material data sets, and (ii) the sample quality of the material data set, *i.e.* the adequacy between phase space coverage of the data sets and of the expected solution. For this purpose, they sampled a nonlinear elastic constitutive model and studied the truss structure both recalled in Fig. II.6.

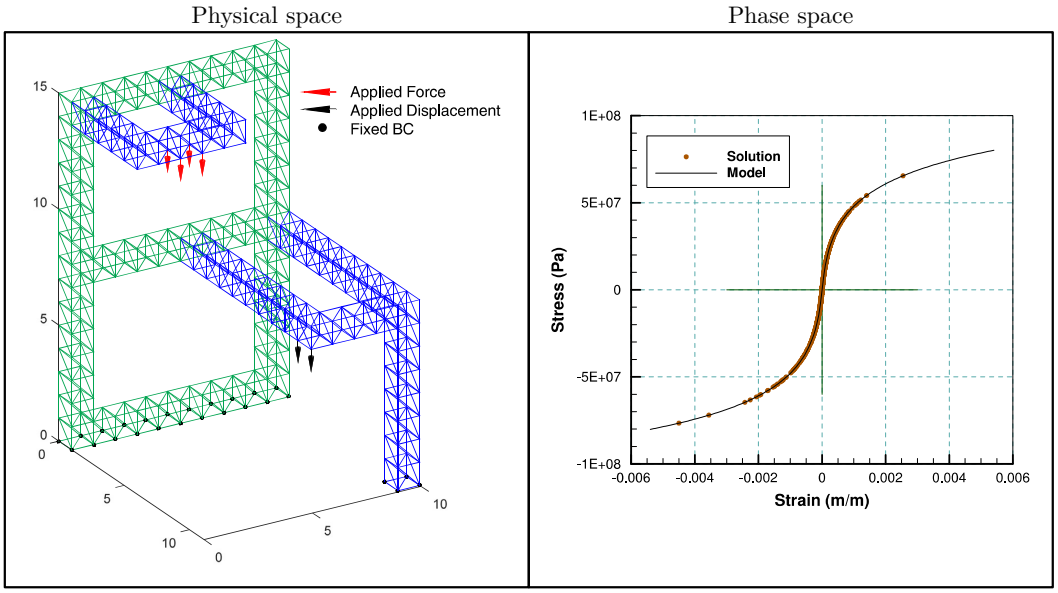
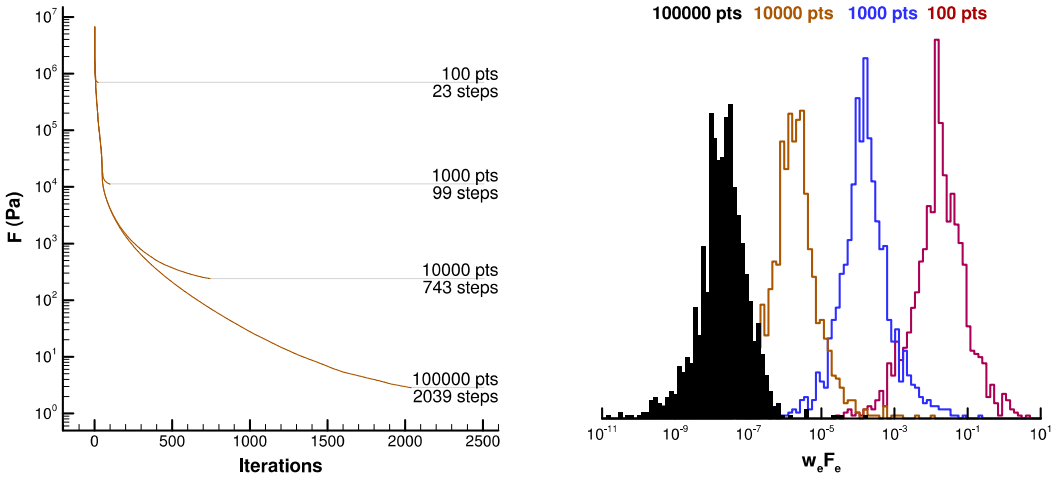


Figure II.6 – Physical space (left): geometry and loading of the truss containing 1048 degrees of freedom. Phase space (right): constitutive model (solid black line) all bars obey used to sample the material data set and reference solution (brown dots), to show the range of local states covered by the problem. From Kirchdoerfer and Ortiz (2016).

Convergence with respect to the number of data points First Kirchdoerfer and Ortiz (2016) monitored the decrease of the objective function throughout the data-driven iterations, for material database of different sizes, resulting from the sampling of the constitutive model, as shown in Fig. II.7(a). They emphasized



(a) (b)

Figure II.7 – (a) Convergence of the global penalty function Eq. (II.26) for noiseless material data sets of increasing size. (b) Distribution of values of the local penalty functions F_e Eq. (II.29) at convergence. From Kirchdoerfer and Ortiz (2016).

that the number of fixed point iterations required to converge is relatively small and increases with the number of material data points. The former is remarkable while the latter was expected, considering the combinatorial complexity of the problem. Second, they monitored the convergence to the reference solution by means of normalized percent root-mean-square (RMS) strain and stress errors defined as

$$\varepsilon_{(\%\mathrm{RMS})} = \frac{1}{\varepsilon^{\mathrm{ref}}_{\mathrm{max}}} \left(\frac{\sum_{e=1}^{M} w_e \left(\varepsilon_e - \varepsilon^{\mathrm{ref}}_e \right)^2}{M} \right)^{\frac{1}{2}} \tag{II.40a}$$

$$\sigma_{(\%\mathrm{RMS})} = \frac{1}{\sigma^{\mathrm{ref}}_{\mathrm{max}}} \left(\frac{\sum_{e=1}^{M} w_e \left(\sigma_e - \sigma^{\mathrm{ref}}_e \right)^2}{M} \right)^{\frac{1}{2}}, \tag{II.40b}$$

respectively, with $\left\{ \left(\varepsilon^{\mathrm{ref}}_e, \sigma^{\mathrm{ref}}_e, \right) \right\}_{e=1}^{M}$ the strain and stress states of the reference solution and $\left(\varepsilon^{\mathrm{ref}}_{\mathrm{max}}, \sigma^{\mathrm{ref}}_{\mathrm{max}}, \right)$ the corresponding maximum values. We summarize their results for both noise-free and noisy material databases of increasing size in Fig. II.8. From these figures, the authors highlighted the following key findings:

- For noise-free data sets:
  - the truss data-driven solver exhibits close to linear convergence to the reference solution, accordingly to the estimates derived in the last part of their paper (Kirchdoerfer and Ortiz, 2016);
  - the alternated minimization is insensitive to the random initial assignment of the material states (see the tightness of the histograms in bottom-left Fig. II.8);

- For noisy data sets:
  - the data-driven solver still converges to the reference solution, although the convergence rate of roughly 1/2 is lower than the one obtained with noise-free data;

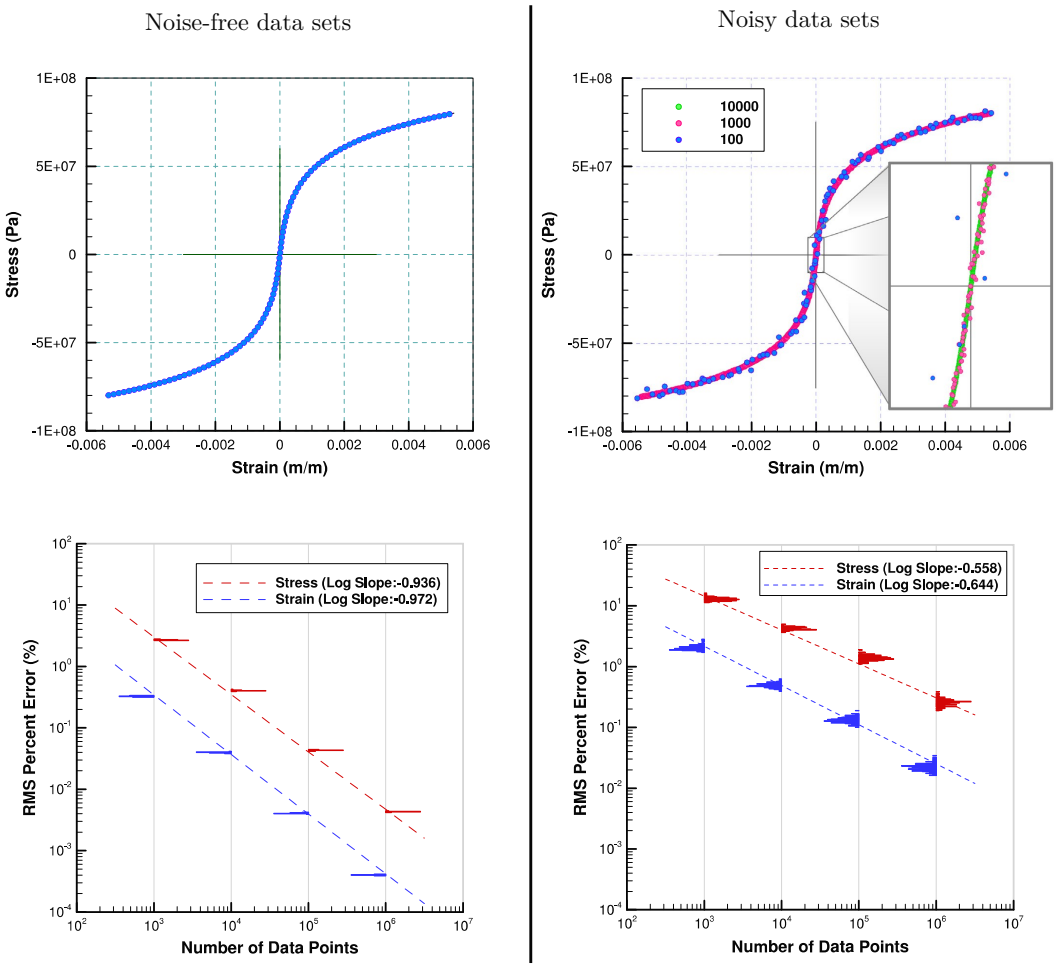


Figure II.8 – Top: material data sets, without noise (left) and with Gaussian random noise (right). Bottom: corresponding convergence of strain and stress RMS errors Eq. (II.40) with number of sampling points; histograms correspond to 30 initial random assignments (left) and to 100 different random data sets (right). From Kirchdoerfer and Ortiz (2016).

Sample quality of the material data set Another feature of the data-driven solver assesses the sample quality of the material data set. Indeed, the final value of the local penalty function Eq. (II.29) provides a measure of the distance between the mechanical and material states of each member of the truss. Histograms of these error measures for the data sets of different sizes are reported in Fig. II.7(b). The

average value decrease with the number of sampling points while a spread remains, "indicating that the states of certain truss members are better sampled by the data set than other". A high value of F_e indeed shows that no material data point exists near enough the solution state of the bar, and conversely. The authors argue that the analysis of this specific feature of the data-driven solver then provides a strategy to adaptively expand the data set, in order to improve phase space coverage in regions lacking data points.

1.4.2 Three-dimensional linear elasticity

The second contribution of Kirchdoerfer and Ortiz (2016) is the analysis of a three-dimensional linear elastic problem. In that case, the high-dimensionality (12) of the phase space, resulting from pairs of symmetric strain and stress tensors, is enough to raise questions on sampling and coverage. To lower phase-space sampling requirements, they restrict to a tensile test specimen, assuming plane stress state and isotropy of the material. The former conditions allows for the phase space to be reduced to a six-dimensional (or three-dimensional according to the authors) phase space: only a neighborhood of the subspace $\sigma_{13} = \sigma_{23} = \sigma_{33} = 0$ is needed, which is accomplished by a sampling of the $\sigma_{11}, \sigma_{22}, \sigma_{12}$ plane stress plane only, on a uniform cubid grid. The corresponding strain $\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{12}$ are generated with an isotropic linear elastic law. The sampling strategy is illustrated in Fig. II.9. The latter condition on isotropy also allows for reducing the amount of data points needed to represent the material behavior: isotropy means that if $(\boldsymbol{\varepsilon}_e, \boldsymbol{\sigma}_e)$ is a material data point, then so are $\left(\boldsymbol{R}_e^T \boldsymbol{\varepsilon}_e \boldsymbol{R}_e, \boldsymbol{R}_e^T \boldsymbol{\sigma}_e \boldsymbol{R}_e\right)$ for all rotations matrices $\boldsymbol{R}_e$. The search for nearest neighbors Eq. (II.33) is then modified to optimize on the rotation matrices as well:

$$\min_{(\boldsymbol{\varepsilon}'_e, \boldsymbol{\sigma}'_e) \in \mathcal{D}_{\text{loc}}^{(e)}} \min_{\boldsymbol{R} \in SO(3)} \left(W_e \left(\boldsymbol{\varepsilon}_e - \boldsymbol{R}_e^T \boldsymbol{\varepsilon}'_e \boldsymbol{R}_e \right) + W_e^* \left(\boldsymbol{\sigma}_e - \boldsymbol{R}_e^T \boldsymbol{\sigma}'_e \boldsymbol{R}_e \right) \right), \tag{II.41}$$

where $SO(3)$ is the group of proper orthogonal matrices in three dimensions. The corresponding optimality condition is handled with a Lagrange multipliers method resulting in a nonlinear system of equations that is solved using a Newton-Raphson scheme and a parametrization of $SO(3)$ (we refer to the original paper for more details).

The thin tensile test specimen is meshed with eight-node hexahedral elements containing eight quadrature or integration points each, with two different average element edge lengths of $h = 1\,\text{mm}$ and $h = 0.5\,\text{mm}$. The corresponding coarse and fine meshes contain one and two elements in the thickness respectively. The meshes together with the material data sampling strategy are illustrated in Fig. II.9.

The convergence of the data-driven solver is again monitored with the penalty function F, which decreases with the number of iterations and increasing size of the material databases, regardless of the mesh resolution, as shown in Fig. II.10(a). The convergence to the reference solution is linear, of rate 3, as shown in Fig. II.10(b),

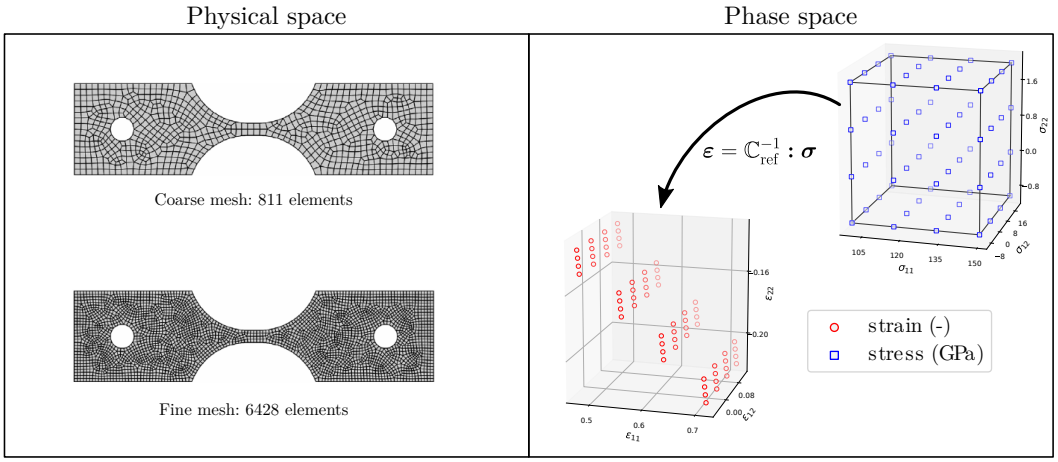


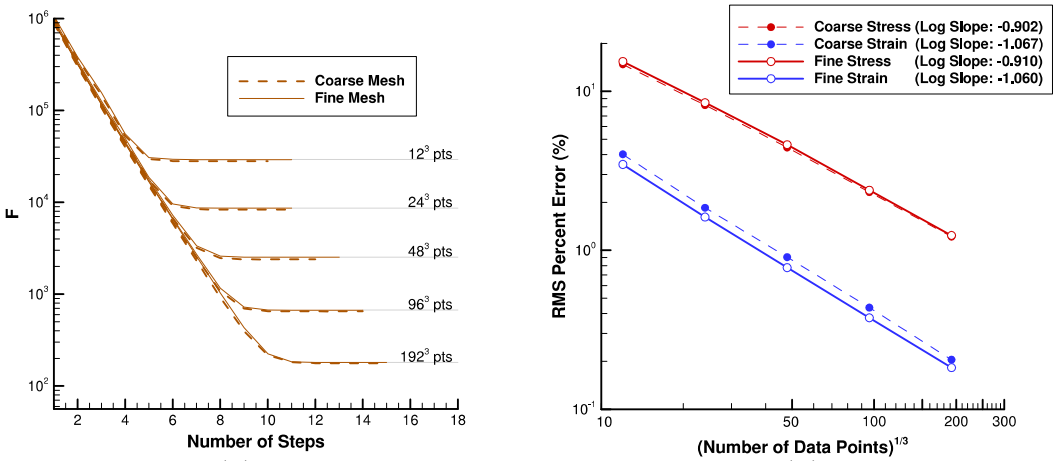
Figure II.9 – (left) Two-dimensional view of the three-dimensional meshes for the tensile test specimen, from Kirchdoerfer and Ortiz (2016). (right) Material data sampling strategy: first, the plane stress subspace is uniformly sampled on a cubic grid; second, the corresponding strain tensors are derived from a linear elastic law, with reference stiffness $\mathbb{C}_{\text{ref}}$.

where the RMS percent errors are redefined as

$$\varepsilon_{(\%\text{RMS})} = \left(\frac{\sum_{e=1}^{M} w_e W \left(\boldsymbol{\varepsilon}_e - \boldsymbol{\varepsilon}_e^{\text{ref}} \right)}{\sum_{e=1}^{M} w_e W \left(\boldsymbol{\varepsilon}_e^{\text{ref}} \right)} \right)^{\frac{1}{2}} \tag{II.42a}$$

$$\sigma_{(\%\text{RMS})} = \left(\frac{\sum_{e=1}^{M} w_e W^* \left(\boldsymbol{\sigma}_e - \boldsymbol{\sigma}_e^{\text{ref}} \right)}{\sum_{e=1}^{M} w_e W^* \left(\boldsymbol{\sigma}_e^{\text{ref}} \right)} \right)^{\frac{1}{2}}, \tag{II.42b}$$

with W and W^* the strain and complementary energy densities computed with the reference modulus $\mathbb{C}_{\text{ref}}$.



(a) (b)

Figure II.10 – (a) Convergence of the local material data assignment iteration. (b) Convergence to the reference solution with respect to sample size, measured by the RMS errors defined Eq. (II.42). From (Kirchdoerfer and Ortiz, 2016).

2 Improvements, inspirations and extensions

The data-driven computing method, as developed in Kirchdoerfer and Ortiz (2016) and Conti et al. (2018) is then well established in small strain elasticity. This seminal work has triggered renewed interest for research in the field of "data-driven computational mechanics", understood in the broad sense. Several authors have recently either contributed to improve the original DDCM, taken inspiration to develop new data acquisition techniques, or extended it to other classes of problems. In this section, we review, as exhaustively as possible, those contributions.

2.1 Dealing with local minima and/or noisy data sets

Kirchdoerfer and Ortiz (2016) quite rapidly evacuated the sensitivity of the proposed data-driven solvers to the initialization of the local data assignment, *i.e.* the choice of material states to begin the alternated minimization with. Yet, as can be seen from Fig. II.8(bottom-left), however narrow are the histograms bining the RMS percent errors obtained for 30 different initial random assignments, the choice of the initial material states inevitably influences the converged solution, for a material database of finite size. This means that the proposed scheme does not converge to the global minimum of the double constrained minimization problem but rather falls in local minima. This behavior is of no significant impact with rather noise-free data sets as the different converged solutions are very similar. However, with increasing noise in the data, possibly creating outliers data points, the alternated heuristic is very limited, as shown by Kanno (2018).

To overcome this issue, Kirchdoerfer and Ortiz (2017) proposed an entropy-maximizing (max-ent) solver that consists "assigning data points a variable relevance

depending on distance to the solution and through maximum-entropy estimation". Associated with a simulated annealing solver, this new approach efficiently penalizes the outliers in the database and is hence more robust to noisy material databases. In fact, pure data-driven approaches, and in particular the DDCM developed by Kirchdoerfer and Ortiz (2016), suffer from the well-known curse of dimensionality: the higher the dimensionality of the phase space, the harder for the material data set to cover the relevant regions needed for the solution to converge. As a reminder, the phase space of general 3D elastic problems is already of dimension 12. The recent contribution of Eggersmann et al. (2020) specifically addresses this issue. It consists in computing a tangent space to each point in the local material data set in a off-line stage, and additionally minimizing the distance to it. This enhancement allows for the interpolation in regions of sparse data sampling and showed higher performances and convergence rates than the original distance minimizing approach and the max-ent solver altogether. A different approach was chosen by Kanno (2019): the author showed that the data-driven distance-minimizing problem is in fact a well-posed Mixed-Integer Quadratic Programming (MIQP) problem, for which efficient branch-and-bounds solvers provide the global minimum of the problem, as opposed to the alternated minimization. Note that the solution is obtained however at high expense, since the number of unknowns is then the product of the number of material data points with the number of integration points in the mesh.

Another way of improving the robustness of data-driven solvers is to assume an existing, yet not parameterized, constitutive manifold in the material database. The optimal material state, for a given mechanical state, is obtained by a projection on the reconstructed manifold, rather than by the original simple nearest neighbor projection. This approach has been adopted in Kanno (2020), where the manifold is approximated using kernel regression, in He and Chen (2020) where a locally convex reconstruction is proposed, and in Gebhardt et al. (2020), where the authors modified the discrete-continuous minimization problem by adding the constraint that the material data points must lie on a smooth implicit manifold. These contributions all advocate for a coupling between a pure model-free data-driven solver and manifold learning techniques, in order to "smooth" the optimization problem and be less sensitive to noise in the data set and hence local minimizers.

When dealing with insufficient data, noise or uncertainty, another common approach is to use a statistical representation of the problem. In particular, Korzeniowski and Weinberg (2019) compared the DDCM approach to standard FE stochastic solvers. Moreover, Ayensa-Jiménez et al. (2018) proposed a stochastic formulation of the data-driven problem, where the material data points are considered to have random nature.

2.2 Data-driven identification

As we now understand, the key input in DDCM is the material database, *i.e.* the collection of stain-stress pairs representative of the material response. The first numerical experiments that were conducted by the aforementioned authors show

that data sampling and phase space coverage are of high importance for the convergence of the data-driven solution. But the simplest, yet not tackled, question is certainly the acquisition of such strain-stress *tensor* pairs. Indeed, the development of full field measurements techniques, such as Digital Image Correlation (DIC) or tomography, has given access to rich data sets of almost directly measured *strains*. But the measurement of *stresses* is usually subject to much more assumptions, in particular homogeneous loading conditions and, in the end, the experimental measure is reduced to a scalar. To provide a way of acquiring multi-dimensional stress field directly from experiments and without a constitutive relation, Leygue et al. (2018) developed an inverse formulation of the DDCM, so-called Data-Driven Identification (DDI). The material data set is no longer the input of the solver but the output, derived from the boundary and loading conditions and the mechanical, compatible, strain field obtained by DIC from the measured displacement field. At first, the DDI method was developed using synthetic FE-based data. Let us consider numerical tests for which the data is collected for different loading conditions, labeled X. Then, the available inputs, are:

- the nodal displacements $\boldsymbol{u}_a^X$,
- the mechanical strains $\boldsymbol{\varepsilon}_e^X = \sum_a \boldsymbol{B}_{ea}^X \boldsymbol{u}_a^X$ at each integration point, computed from the finite element geometry and connectivity, encoded in matrix $\boldsymbol{B}_{ea}^X$
- the applied nodal forces $\boldsymbol{f}_a^X$
- prescribed nodal displacements

The aim of the DDI method is then to compute from the available data:

- the mechanical stress state $\boldsymbol{\sigma}_e^X$ of each integration points e of the loading stage X,
- the database of material states $(\boldsymbol{\varepsilon}_i^*, \boldsymbol{\sigma}_i^*)$. The total number of material states is denoted N^* and fixed beforehand. They are determined according to the unchanged norm $|\cdot|_{\mathbb{C}}$ (see Eq. (II.29)) of the local phase space.

The discrete-continuous problem then writes:

$$\text{solution} = \arg \min_{\boldsymbol{\sigma}, (\boldsymbol{\varepsilon}', \boldsymbol{\sigma}') \in \mathcal{D}} \sum_{e,\, X} w_e^X \left| \left(\boldsymbol{\varepsilon}_e^X, \boldsymbol{\sigma}_e^X\right) - \left(\boldsymbol{\varepsilon}'_{eX}, \boldsymbol{\sigma}'_{eX}\right) \right|_{\mathbb{C}} \tag{II.43a}$$

$$\sum_{e=1}^{M} w_e^X \boldsymbol{B}_{ea}^{X^T} \boldsymbol{\sigma}_e^X - \boldsymbol{f}_a^X = 0, \, \forall a, \, X, \tag{II.43b}$$

where the material state $\left(\boldsymbol{\varepsilon}'_{eX}, \boldsymbol{\sigma}'_{eX}\right)$ of integration point e in loading stage X takes value in the (unknown) material database $\mathcal{D}_{\text{loc}} = \{(\boldsymbol{\varepsilon}_i^*, \boldsymbol{\sigma}_i^*)\}_{i=1}^{N^*}$. In comparison with the original DDCM formulation, the minimization variables have changed, as well as the compatibility equation is no longer needed, since the mechanical strains are known. The approach was successfully implemented in small strain nonlinear

elasticity for synthetic problems in Leygue et al. (2018). It is of particular interest for supplying DDCM simulations with "importance sampled" material data sets, as shown in Stainier et al. (2019). In addition, it was successfully extended to large strains and tested with raw experimental data in the PhD thesis of Dalémat (2019), which provided a new approach for "measuring stress field without constitutive equation" (Dalémat et al., 2019). This type of inverse methods was also developed for elasto-plasticity by Réthoré et al. (2018) and further explored in nonlinear elasticity, plasticity and dynamics in Leygue et al. (2019). This appealing direction of research is out of the scope of the present thesis: we will develop data-driven solvers and numerically tests them with synthetic data, assuming that, in a near future, some experimentally acquired data turn available thanks to these techniques.

2.3 Extension to other classes of problems

In the recent years, the DDCM has been extended to other classes of problems. First, by the authors themselves, who extended the method to dynamics in Kirchdoerfer and Ortiz (2018). We can also mention the work of Nguyen et al. (2020) who proposed a variational formulation suitable for solving Poisson equation, diffusion problems and which applies to elasticity. The authors also elucidated a proper transition from the continuous formulation to the discrete FE formulation, with special attention to the boundary conditions needed for the Lagrange multipliers. Recently, Carrara et al. (2020) have also proposed an extension to fracture mechanics.

In fact, changing the class of problem tackled by the data-driven approach means adapting the phase-space to the quantities which are relevant to describe the problem. For instance, Eggersmann et al. (2019) provided an application of the DDCM to history-dependent behavior such as visco-elasticity. Notably, the authors discussed the possible manner of taking into account the past history of deformation. Finally, Nguyen and Keip (2018) proposed an extension to large strain elasticity, adopting a Lagrangian formulation: the phase-space is now constituted with all possible Green-Lagrange strain–second Piola-Kirchhoff stress pairs $(\boldsymbol{E}, \boldsymbol{S})$. The structure of the minimization problem is unchanged, but the mechanical constraints (compatibility and equilibrium) are now nonlinear. Another formulation is then chosen by Conti et al. (2020), which we refer to as the nominal formulation: the minimization is conducted over all possible deformation gradient–first Piola Kirchhoff stress pairs $(\boldsymbol{F}, \boldsymbol{P})$. We shall explore these alternatives in the present thesis.

3 Summary

The DDCM approach first introduced by Kirchdoerfer and Ortiz (2016) not only provides a new FE linear elastic solvers, which can not yet compete with the mature FE method obviously, but mostly a new computing paradigm, which has triggered many applications and extensions in the past few years and months. It invites to a radical change of perspective: the material behavior is no longer hard-coded in the simulation with a constitutive model but rather represented with *raw* strain-

stress data, which can, if needed, be locally interpreted in terms of tangent spaces or manifolds, without any prior assumptions. It also changes the structure of the problems to be solved: the objective is now to minimize some distance to a material data set, while ensuring mechanical admissibility constraints. Hence, data-driven computing is more consistent than the classical approach with the reliability of each type of equations. The difference between the standard and the data-driven FE formulations are highlighted in Fig. II.11.

The DDCM approach was developed and can now be considered mature for (non-linear) elastic problems, under the linearized kinematics assumption, for homogeneous materials. Several directions of research then offered for this PhD thesis from the starting point of Kirchdoerfer and Ortiz (2016). We chose to investigate two of them: (i) the extension to other mechanical behavior for homogeneous materials, (ii) the application of the method to the computational homogenization of heterogeneous materials. The rest of manuscript is then divided into two parts:

- Adaptation to large strain elasticity.
- Towards data-driven multiscale numerical schemes.

In Part B, we first address the extension of DDCM to large elastic deformation. As aforementioned, this requires to formulate the distance-minimizing problem in a different phase space. The specific case of large strain elasticity was favored over the extension to anelastic behaviors because of its particular interest for rubber-like materials, which bring about a large part of research on constitutive modeling, as pointed out in Chapter I. Besides, once finite strain is elucidated for elastic materials, it should allow for the extension of the approach to anelastic materials exhibiting large deformation, based on the propositions by Eggersmann et al. (2019). To the best of our knowledge, only the two above-mentioned contributions provided a large strain extension to the DDCM: the Lagrangian formulation of Nguyen and Keip (2018) and the nominal formulation of Conti et al. (2020). However, as first propositions, they both present some limitations and are hence worth exploring further.

- The contribution of Nguyen and Keip (2018) lacks a proper generic FE formulation which would compare with the well-established standard finite strain FE method we recalled from Bonet and Wood (2008) in Chapter I. In addition, little analysis on the richness and the sampling of the material database has been conducted. The mathematical convergence of the Lagrangian formulation has also yet to be assessed, due to the nonlinearity of the kinematic and equilibrium constraints.

- With their proposition, Conti et al. (2020) overcome the latter issue. However, the proposed continuum mechanics formulation does not provide a straightforward FE implementation and the performance of the attendant solver has yet to be assessed.

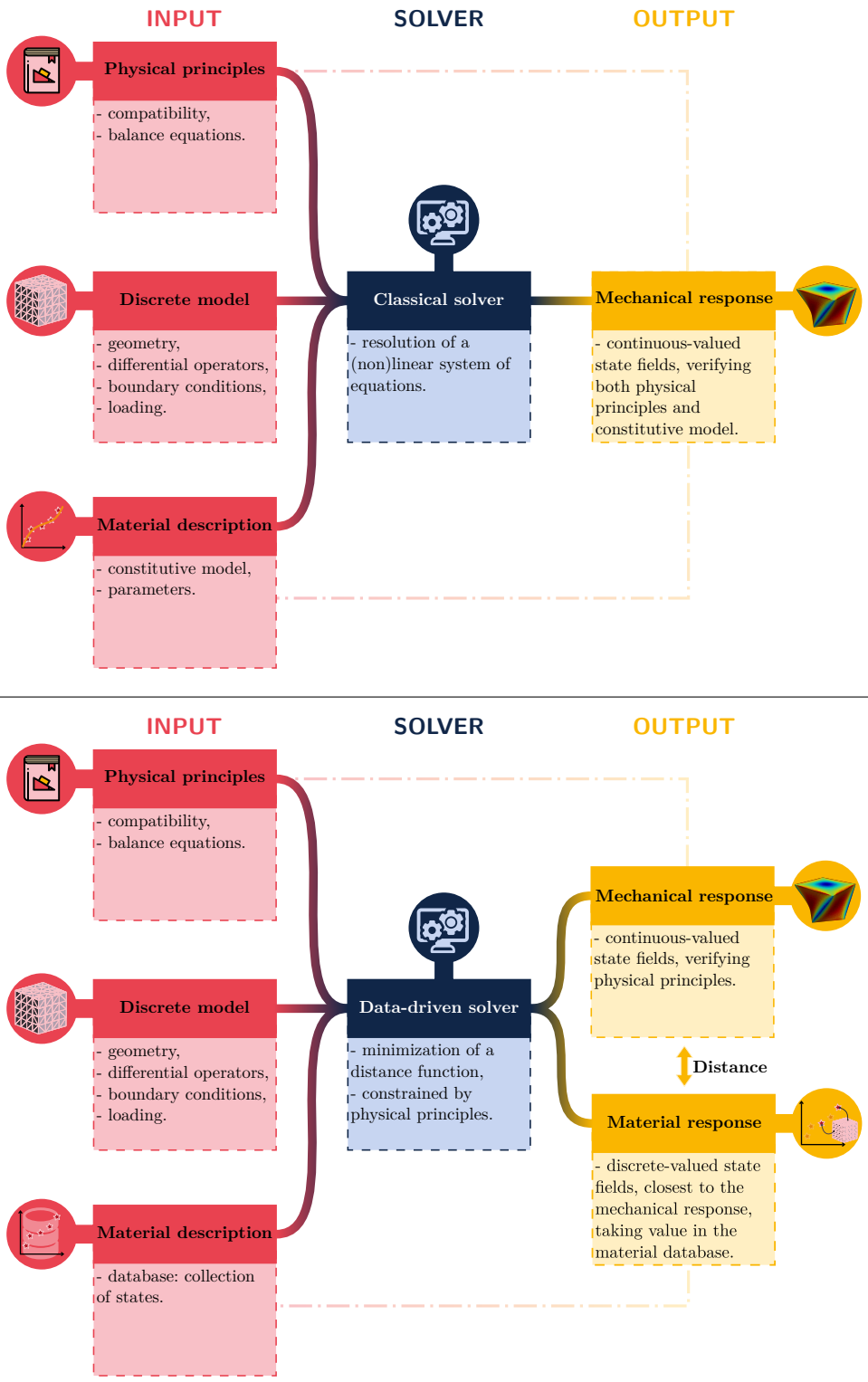


Figure II.11 – Input-output diagrams of a standard FE solver (top) and the DDCM solver (bottom).

In the present thesis, we then chose to investigate and compare both formulations, so as to address the remaining gaps towards a standardization analogous to the original DDCM approach. In short, the extension of DDCM to finite strain elasticity requires to modify several blocks of the representation in Fig. II.11:

1. the core of the data-driven solver (blue central block) must be modified, as the distance function and the constraints differ from one formulation to the other,

2. the material description (bottom left red block) should now involve a database of Lagrangian or nominal states,

3. the mechanical response (top right yellow block) is subsequently modified and must be evaluated with renewed relevant metrics.

The manuscript is accordingly organized as follows.

Part B. Both the Lagrangian and nominal formulations of a finite strain FE data-driven solver are derived in Chapter III. Chapter IV is dedicated to the generation and analysis of different material databases suitable for large strain simulation. As pointed out by Kirchdoerfer and Ortiz (2016), the distance-minimizing formulation provides error measures which "highlight data regions that require additional resolution" and penalize "attempts to simulate beyond the data regime", while the constitutive models are often used outside of their validity domain. A thorough examination of the material data sets is equally required in finite strain and has not been addressed yet, to the best of our knowledge. Finally, we investigate the performance of the thus-elaborated data-driven solvers in Chapter V, with the aid of a particular example: the static FE analysis of a three-dimensional nonlinear elastic solid, subject to large deformation.

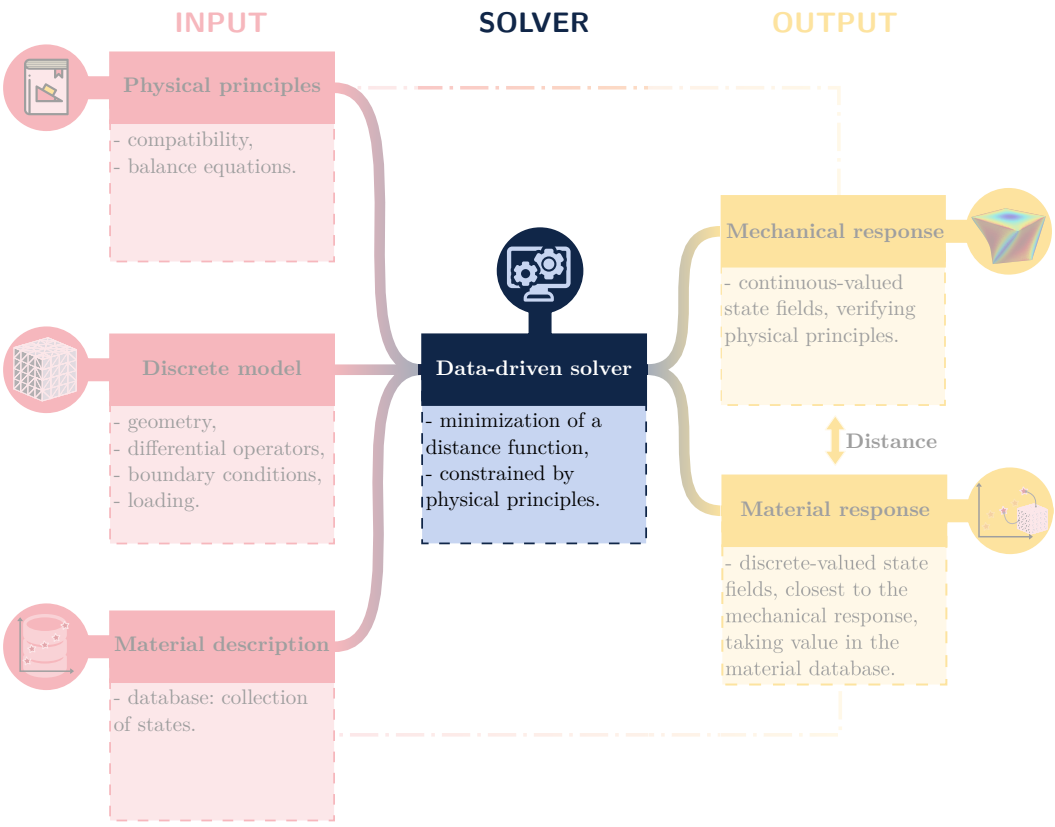
Part C. The second part of our work is devoted to the development of data-driven multiscale simulations. We introduce in Chapter VI the preliminary developments and attendant results for a novel data-driven scheme applied to heterogeneous materials. The material database then consists of macroscopic strain-stress pairs and is successively enriched, in a multiscale iterative process, from FE computational homogenization of the microstructure.

Part B

Adaptation to large strain elasticity

Chapter III

Data-driven solvers for large strain elasticity



Contents

1 Introduction

From Chapter II, we can say that DDCM roughly consists in minimizing the distance between two types of strain-stress pair fields: one, representing the material response, is selected in a material database and the other, verifying both equilibrium and compatibility equations, represents the mechanical state of the body. To extend the approach to large elastic deformations, we now need to determine the appropriate constitutive variables to work with.

In the case of materials exhibiting large deformations independent of the history of deformation, the strain and stress fields still suffice to fully describe the mechanical state of a material point in the body. From Chapter I, we know that several work/power conjugate strain-stress pairs are available in finite strain (cf. Table I.1):

Lagrangian: the Green-Lagrange strain tensor and the second Piola-Kirchhoff stress tensor $(\boldsymbol{E}, \boldsymbol{S})$;

Nominal: the deformation gradient tensor and the first Piola-Kirchhoff stress tensor $(\boldsymbol{F}, \boldsymbol{P})$;

Eulerian: the rate of deformation tensor and the Cauchy stress tensor $(\boldsymbol{d}, \boldsymbol{\sigma})$;

Note that defining an Eulerian strain measure that is work conjugate to the Cauchy stress is not as immediate as for the Lagrangian or nominal formulations. That is why we did not consider the Eulerian framework in the present thesis. The extension of the DDCM to large strain elasticity has been tackled in two recent publications, by Nguyen and Keip (2018) and Conti et al. (2020). The former proposed a Lagrangian formulation of the problem while the latter provided a nominal formulation, enabling mathematical proofs of convergence, inspired from their previous work in small strain (Conti et al., 2018). These first contributions have yet to be explored and compared, and both lack a proper FE formulation. In this chapter, we then provide FE formulations and solvers for both approaches and discuss their implementation and robustness, thanks to simple examples.

2 Finite strain data-driven computational mechanics

Both Lagrangian and nominal formulations presented here adopt the same heuristic as in small strain, *i.e.* an alternated minimization between the material states and the mechanical states, which is represented in Fig. III.1. Both approaches require several key ingredients of the DDCM to be adapted to finite strain elasticity:

- the strain-stress phase space is now constituted either with Lagrangian conjugate tensors $(\boldsymbol{E}, \boldsymbol{S})$ or with nominal conjugate pairs $(\boldsymbol{F}, \boldsymbol{P})$;
- the phase space is equipped with a new metric measuring distances between local states, thus modifying the objective function;

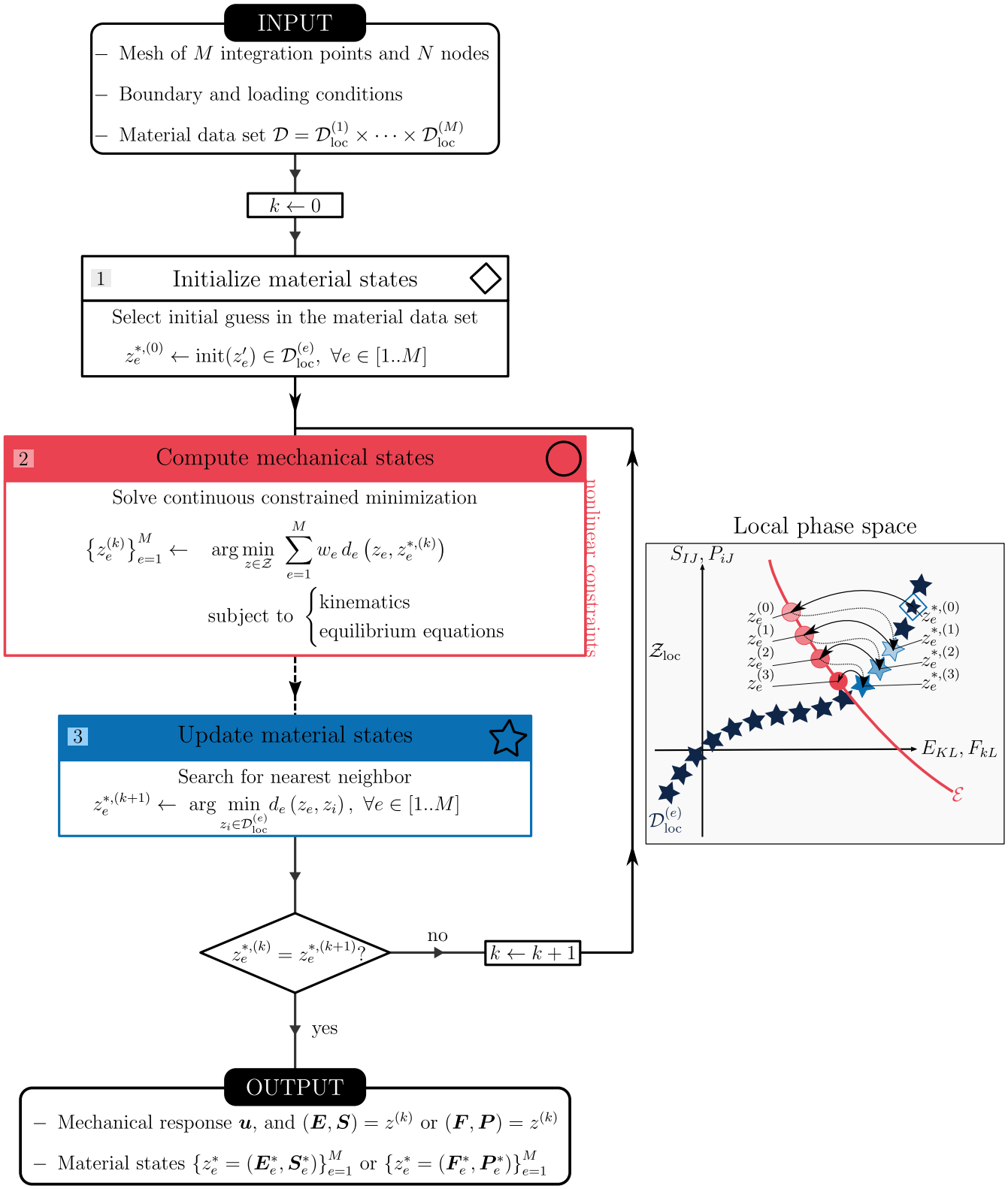


Figure III.1 – Alternated minimization used for large strain data-driven solvers, adapted from the original heuristic proposed by Kirchdoerfer and Ortiz (2016).

- the kinematics and equilibrium equations are reformulated using the classical Lagrangian or nominal formulation recalled in Chapter I;
- the nonlinearity of the above-mentioned minimization constraints complicates the computation of the mechanical states (step 2 of the algorithm), for which specific numerical schemes must be employed.

This section is then organized as follows: for each formulation, we present the continuum mechanics of the large strain data-driven BVP before its FE discretization. Then, we provide the specific numerical strategies employed to adapt the data-driven solvers to the nonlinearity of the resulting equations. We follow, when possible, the same rationale as the one adopted in Chapter II to present small strain DDCM.

2.1 Lagrangian approach

As mentioned in Nguyen and Keip (2018), a reasonable choice for the strain-stress pair field in finite strain elasticity is $(\boldsymbol{E}, \boldsymbol{S})$ as it ensures both objectivity and a straightforward linearization to the small strain elastic case. Moreover, it involves symmetric tensors which limits the dimensionality of the phase space to at most 12.

In this section, we provide the Lagrangian formulation for continuum and discretized data-driven mechanics, inspired by the work of Nguyen and Keip (2018) but in a more general manner. Then, we present the corresponding FE data-driven solver.

2.1.1 Continuum mechanics formulation

The large strain continuum mechanics formulation is derived as in the small strain case (see Section 1.2.1 of Chapter I), *i.e.* defining local state functions mapping from the physical space to the phase space.

Let us consider an elastic body occupying a domain $\Omega_0 \in \mathbb{R}^n$ in the reference configuration and $\Omega \in \mathbb{R}^n$ is the deformed one. It is subject to quasi-static loading. For now, the general three-dimensional case $n = 3$ is considered only; specific assumptions allowing for the reduction to two-dimensional mechanics will be discussed next. A material point in the body moves from its initial coordinate $\boldsymbol{X}$ to its current coordinate $\boldsymbol{x}$. Since we adopt a Lagrangian formulation of the BVP, the local state function z now maps every material point $\boldsymbol{X}$ of Ω_0 to its Lagrangian local state:

$$\begin{aligned} z: \quad & \Omega_0 \mapsto \mathcal{Z}_{\text{loc}} \\ & \boldsymbol{X} \to z(\boldsymbol{X}) = (\boldsymbol{E}(\boldsymbol{X}), \boldsymbol{S}(\boldsymbol{X})) \end{aligned} \tag{III.1}$$

where the local phase space $\mathcal{Z}_{\text{loc}}$ is $\mathbb{R}^{n\times n}_{\text{sym}} \times \mathbb{R}^{n\times n}_{\text{sym}}$. The global phase space $\mathcal{Z}$ is again the collection of local state functions $z: \quad \Omega_0 \to \mathcal{Z}_{\text{loc}}$. Similarly to the small strain case, two different subsets of $\mathcal{Z}$ divide mechanically admissible and material quantities.

First, we gather in the constraint set $\mathcal{E}$ all local state functions mapping to mechanically admissible strain-stress fields, *i.e.*, verifying both compatibility and

equilibrium equations. Using the Lagrangian continuum mechanics setting and notations recalled in Chapter I, these write:

$$\boldsymbol{F}(\boldsymbol{X}) = \boldsymbol{\nabla}_0 \boldsymbol{u} + \boldsymbol{I} \text{ and } \boldsymbol{E}(\boldsymbol{X}) = \frac{1}{2}\left(\boldsymbol{F}^T(\boldsymbol{X})\boldsymbol{F}(\boldsymbol{X}) + \boldsymbol{I}\right) \qquad \text{in } \Omega_0, \tag{III.2a}$$

$$\boldsymbol{u}(\boldsymbol{X}) = \boldsymbol{u}_D(\boldsymbol{X}) \qquad \text{on } \Gamma_{0_D}, \tag{III.2b}$$

and

$$\delta W = \int_{\Omega_0} \boldsymbol{S} : \delta \dot{\boldsymbol{E}} \, \mathrm{d}V - \int_{\Omega_0} \boldsymbol{f}_0 \cdot \delta \boldsymbol{v} \, \mathrm{d}V - \int_{\Omega_0} \boldsymbol{t}_0 \cdot \delta \boldsymbol{v} \, \mathrm{d}A = 0, \, \forall \delta \boldsymbol{v} \tag{III.3}$$

where $\delta \dot{\boldsymbol{E}} = 1/2 \left(\delta \dot{\boldsymbol{F}}^T \boldsymbol{F} + \boldsymbol{F}^T \delta \dot{\boldsymbol{F}}\right)$, with $\delta \dot{\boldsymbol{F}} = \partial \delta \boldsymbol{v} / \partial \boldsymbol{X} = \boldsymbol{\nabla}_0 \delta \boldsymbol{v}$. The material-independent constraint set $\mathcal{E} \subset \mathcal{Z}$ is again defined as

$$\mathcal{E} = \{z \in \mathcal{Z} \mid \text{Eqs. (III.2) and (III.3)}\}. \tag{III.4}$$

Second, the material data set $\mathcal{D}$ is defined by the collection of local state functions mapping to Lagrangian strain-stress pairs, stored in the local material data set or material database $\mathcal{D}_{\text{loc}}$:

$$\mathcal{D} = \{z \in \mathcal{Z} \mid z(\boldsymbol{X}) \in \mathcal{D}_{\text{loc}}\}, \tag{III.5}$$

with

$$\mathcal{D}_{\text{loc}} = \{(\boldsymbol{E}_i, \boldsymbol{S}_i) \mid i \in [1 \,..\, m]\} \subset \mathcal{Z}_{\text{loc}}, \tag{III.6}$$

where m is the number of material data points.

The solution set $\mathcal{S}$ of the data-driven BVP can again no longer lie at the intersection $\mathcal{D} \cap \mathcal{E}$ since $\mathcal{D}$ is a discrete-valued set. It is defined as the *mechanical state* function $z \in \mathcal{E}$ closest to the material data set $\mathcal{D}$:

$$\mathcal{S} = \arg\min_{z \in \mathcal{E}} d(z, \mathcal{D}). \tag{III.7}$$

where the distance d is defined very similarly to the small strain case as

$$d(z, \mathcal{Y}) = \min_{y \in \mathcal{Y}} \|z - y\|_{\mathbb{C}}, \; \forall \mathcal{Y} \subset \mathcal{Z}, \tag{III.8a}$$

with

$$\|z\|_{\mathbb{C}} = \left(\int_{\Omega_0} \left(W(\boldsymbol{E}) + W^*(\boldsymbol{S})\right) \mathrm{d}V\right)^{\frac{1}{2}}, \; \forall z \in \mathcal{Z}, \tag{III.8b}$$

where the deviation functions W and W^* are defined as

$$W(\boldsymbol{E}) = \frac{1}{2}(\mathbb{C} : \boldsymbol{E}) : \boldsymbol{E}, \tag{III.9a}$$

$$W(\boldsymbol{S}) = \frac{1}{2}(\mathbb{C}^{-1} : \boldsymbol{S}) : \boldsymbol{S}, \tag{III.9b}$$

with $\mathbb{C}$ a constant symmetric fourth-order positive definite tensor.

For later purposes, let us remark that the minimization Eq. (III.7) can equivalently be conducted with the square of the norm Eq. (III.8b), since the distance set $\{\|z - z^*\|_{\mathbb{C}} \mid z^* \in \mathcal{D}\}$ only contains real positive numbers (Zakon, 2017):

$$\begin{aligned}\mathcal{S} = \arg\min_{z\in\mathcal{E}} \; d(z,\mathcal{D}) = \arg\min_{z\in\mathcal{E}} \; d^2(z,\mathcal{D}) \\ = \arg\min_{z\in\mathcal{E}} \left(\min_{y\in\mathcal{D}} \|z-y\|_{\mathbb{C}}\right)^2 = \arg\min_{z\in\mathcal{E}} \; \min_{y\in\mathcal{D}} \|z-y\|^2_{\mathbb{C}}.\end{aligned} \tag{III.10}$$

2.1.2 Reduction to two-dimensional problems

Before moving to the finite element discretization of the finite strain data-driven setting presented above, we discuss the reduction to two-dimensional problems.

Two-dimensional problems typically arise under plane strain or plane stress conditions. We show in this section that the local phase space $\mathcal{Z}_{\text{loc}}$ is safely reduced to $\mathbb{R}^{2\times2} \times \mathbb{R}^{2\times2}$ in both cases.

On the one hand, let us consider a BVP where a plane stress condition is verified along $\boldsymbol{e}_3$ and the material is incompressible. The former constraint enables the stress tensor $\boldsymbol{S} \in \mathbb{R}^{3\times3}_{\text{sym}}$ to be reduced to its in-plane components $\boldsymbol{S}_{(2)} \in \mathbb{R}^{2\times2}_{\text{sym}}$ only, as

$$\boldsymbol{S} = \begin{pmatrix} \boldsymbol{S}_{(2)} & \boldsymbol{0} \\ \boldsymbol{0} & S_{33} \end{pmatrix}_{(\boldsymbol{e}_1,\boldsymbol{e}_2,\boldsymbol{e}_3)}, \tag{III.11}$$

with $S_{33} = 0$. Together with the plane stress condition $\boldsymbol{S}\boldsymbol{e}_3 = 0$, the latter constraint enables to also only retain the in-plane components $\boldsymbol{E}_{(2)}$ of the strain tensor $\boldsymbol{E} \in \mathbb{R}^{3\times3}_{\text{sym}}$, as $J = \det \boldsymbol{F} = 1$ and

$$\boldsymbol{E} = \begin{pmatrix} \boldsymbol{E}_{(2)} & \boldsymbol{0} \\ \boldsymbol{0} & E_{33} \end{pmatrix}_{(\boldsymbol{e}_1,\boldsymbol{e}_2,\boldsymbol{e}_3)}, \tag{III.12}$$

where the out-of-plane strain $E_{33} = (j^{-2} - 1)/2$ can be determined *a posteriori* from $j = \det \boldsymbol{F}_{(2)}$, with $\boldsymbol{F}_{(2)} \in \mathbb{R}^{2\times2}$ the in-plane components of the deformation gradient tensor $\boldsymbol{F}$.

On the other hand, when plane strain conditions are assumed, $F_{33} = 1$. Hence $\boldsymbol{F}_{(2)}$ and $\boldsymbol{E}_{(2)}$ suffice to characterize the deformation state. However, the out-of-plane stress S_{33} can not be determined in the data-driven approach, as no constitutive equation relates the two three-dimensional tensors. The data-driven results of a plane strain problem will then be purely two-dimensional.

In summary, provided that the material data set is constituted with two dimensional $(\boldsymbol{E}_{(2)}, \boldsymbol{S}_{(2)})$ tensors, the data-driven approach derived above is unchanged whether $n = 2$ or $n = 3$. Note that a similar rationale allows for the definition of data-driven one-dimensional settings, with applications to other types of mechanical assumption (*e.g.* beam theory).

2.1.3 Finite element formulation

The discretization of the previous data-driven BVP is presented into two steps: (i) the global phase space $\mathcal{Z}$ and metrics are derived from the finite element discretization of the physical space Ω_0 ; (ii) the material data set is then defined as the corresponding subset while the kinematics and equilibrium constraints Eqs. (III.2) and (III.3) are discretized in order to formulate the constraint set.

Global phase space and metrics. The data-driven problem Eq. (III.10) requires to evaluate the following integrals:

$$\begin{aligned}\|z\|_{\mathbb{C}}^2 &= \int_{\Omega_0} \left(\frac{1}{2}(\mathbb{C} : \boldsymbol{E}) : \boldsymbol{E} + \frac{1}{2}(\mathbb{C}^{-1} : \boldsymbol{S}) : \boldsymbol{S}\right) \mathrm{d}V \\ &= \int_{\Omega_0} \left(\frac{1}{2}(\mathbb{C} : \boldsymbol{E}) : \boldsymbol{E}\right) \mathrm{d}V + \int_{\Omega_0} \left(\frac{1}{2}(\mathbb{C}^{-1} : \boldsymbol{S}) : \boldsymbol{S}\right) \mathrm{d}V, \end{aligned} \tag{III.13}$$

where $\boldsymbol{E}$ and $\boldsymbol{S}$ verify equations Eqs. (III.2a) and (III.3). Let us consider a finite element discretization of the domain Ω_0 into N nodes, labeled $a \in [1\,..\,N]$. Let $\alpha \in [1\,..\,N \times n]$ label the degrees of freedom; note that here[1]

$$\alpha = \alpha(a, i) = (a-1)n + i \tag{III.14}$$

with $i \in [1\,..\,n]$ the direction. Moreover, the elements are equipped with a Gauss quadrature rule such that the mesh comprises M integration points, labeled $e \in [1\,..\,M]$. As mentioned in Section 1.2.3 of Chapter I, the integrals are evaluated as weighted sums of their integrands evaluated at the integration points. The integral of a functional $\mathcal{F}$ over the domain Ω_0 then results from

$$\int_{\Omega_0} \mathcal{F}(\boldsymbol{X})\,\mathrm{d}V = \sum_{e=1}^{M} \mathcal{F}(\boldsymbol{X}_e), \tag{III.15}$$

with w_e is the weight associated to the integration point e of coordinate $\boldsymbol{X}_e$ in the reference configuration Ω_0. In the following, we explicitly derive the evaluation of the strain $\boldsymbol{E}$ and stress $\boldsymbol{S}$ fields at the integration points, in the data-driven setting.

The first term in Eq. (III.13) is evaluated by simply replacing the strain field $\boldsymbol{E}$ with its relationship to the displacement field Eq. (III.2a). As a reminder, the displacement field is interpolated in terms of nodal values $\{\boldsymbol{u}_a\}_{a=1}^N$ and nodal shape functions $\{\mathcal{N}_a\}_{a=1}^N$ as

$$\boldsymbol{u}(\boldsymbol{X}) = \sum_{a=1}^{N} \mathcal{N}_a(\boldsymbol{X})\boldsymbol{u}_a,\ \forall \boldsymbol{X} \in \Omega_0. \tag{III.16}$$

From Eq. (III.2a) the strain field is then approximated by

$$\boldsymbol{E}(\boldsymbol{X}) = \frac{1}{2}\left((\boldsymbol{\nabla}_0\boldsymbol{u})^T + \boldsymbol{\nabla}_0\boldsymbol{u} + (\boldsymbol{\nabla}_0\boldsymbol{u})^T\,\boldsymbol{\nabla}_0\boldsymbol{u}\right), \tag{III.17a}$$

[1] as one possibility among others.

with

$$\boldsymbol{\nabla}_0 \boldsymbol{u}\,(\boldsymbol{X}) = \sum_{a=1}^{N} (\boldsymbol{u}_a \otimes \boldsymbol{\nabla}_0 \mathcal{N}_a(\boldsymbol{X}_e))\,. \tag{III.17b}$$

In the classical approach, the approximation of the stress field readily results from the approximation of the displacement field, *via* the compatibility and the constitutive equations. In the data-driven approach however, the mechanical stress field $\boldsymbol{S}$ verifying equilibrium Eq. (III.3) is no longer directly related to the displacement field. The definition of the functional space for the FE stress field in the data-driven setting is out of the scope of the present work. Instead, the stresses are merely evaluated at the integration points, using Eq. (III.15), as

$$\int_{\Omega_0} \boldsymbol{S}(\boldsymbol{X})\,\mathrm{d}\boldsymbol{X} = \sum_{e=1}^{M} w_e \boldsymbol{S}_e \tag{III.18}$$

where $\boldsymbol{S}_e = \boldsymbol{S}(\boldsymbol{X}_e)$.

In the finite element formulation, the local state functions are then defined as the mapping from the integration points to the corresponding strain-stress pairs as

$$\begin{array}{rccl} z_e: & \Omega_0 & \to \mathcal{Z}_{\mathrm{loc}}^{(e)} = \mathbb{R}_{\mathrm{sym}}^{n\times n} \times \mathbb{R}_{\mathrm{sym}}^{n\times n} \\ & \boldsymbol{X}_e & \mapsto (\boldsymbol{E}_e, \boldsymbol{S}_e) = (\boldsymbol{E}(\boldsymbol{X}_e), \boldsymbol{S}(\boldsymbol{X}_e))\,. \end{array} \tag{III.19}$$

The global phase space being the product set $\mathcal{Z} = \mathcal{Z}_{\mathrm{loc}}^1 \times \cdots \times \mathcal{Z}_{\mathrm{loc}}^M$, a global point z in $\mathcal{Z}$ is the collection of M local state functions: $z = \{z_e\}_{e=1}^M$. The discretization of the metric Eq. (III.8b) then results from the evaluations of the strain and stress fields Eqs. (III.17) and (III.18) as

$$d(z, \mathcal{Y}) = \min_{y\in\mathcal{Y}} \|z - y\|_{\mathbb{C}},\ \forall \mathcal{Y} \subset \mathcal{Z}, \tag{III.20a}$$

with

$$\|z\|_{\mathbb{C}}^2 = \sum_{e=1}^{M} w_e \left(W(\boldsymbol{E}_e) + W^*(\boldsymbol{S}_e)\right),\ \forall z = \{z_e = (\boldsymbol{E}_e, \boldsymbol{S}_e)\}_{e=1}^M \in \mathcal{Z}, \tag{III.20b}$$

with $\boldsymbol{E}_e = \boldsymbol{E}(\boldsymbol{X}_e)$ from Eq. (III.17).

Constraint set and material data set. On the one hand, the set of mechanically admissible local state functions $\mathcal{E} \subset \mathcal{Z}$ in the finite element formulation is obtained by injecting the approximations of the displacement and stress fields Eqs. (III.16) and (III.18) into equations Eqs. (III.2) and (III.3).

- The Dirichlet boundary conditions Eq. (III.2b) are encoded in an array of n_{D} scalar prescribed displacements, denoted $\left\{u_\alpha^{\mathrm{D}}\right\}_{\alpha\in D}$ with $D = \{\alpha_1, \cdots, \alpha_{n_{\mathrm{D}}}\}$,

such that $\alpha_k \in [1\,..\,Nn]$, $\forall k \in [1\,..\,n_{\mathrm{D}}]$. From Eq. (III.17), the kinematics constraints Eq. (III.2) then write

$$\boldsymbol{E}_e = \frac{1}{2}\left((\boldsymbol{\nabla}_0^e \boldsymbol{u})^T + \boldsymbol{\nabla}_0^e \boldsymbol{u} + (\boldsymbol{\nabla}_0^e \boldsymbol{u})^T \boldsymbol{\nabla}_0^e \boldsymbol{u}\right), \quad \forall e \in [1\,..\,M], \tag{III.21a}$$

$$\boldsymbol{u}_a \cdot \boldsymbol{e}_i = u_\alpha^{\mathrm{D}}, \quad \forall (a,i) : \alpha(a,i) \in D, \tag{III.21b}$$

with $\boldsymbol{\nabla}_0^e \boldsymbol{u} = \boldsymbol{\nabla}_0 \boldsymbol{u}(\boldsymbol{X}_e)$ from Eq. (III.17b).

- The discretization of the material principle of virtual work Eq. (III.3) is obtained by using the same discretization for the virtual velocity $\delta \boldsymbol{e}$ as in Eq. (III.16) and the symmetry of $\boldsymbol{S}$. We then have successively

$$\begin{aligned} \boldsymbol{S} : \delta\dot{\boldsymbol{E}} = \boldsymbol{S} : \left(\boldsymbol{F}^T \delta\dot{\boldsymbol{F}}\right) &= (\boldsymbol{F}\boldsymbol{S}) : \delta\dot{\boldsymbol{F}} \\ &= (\boldsymbol{F}\boldsymbol{S}) : \left(\sum_{a=1}^{N} \delta\boldsymbol{v}_a \otimes \boldsymbol{\nabla}_0 \mathcal{N}_a\right) = \sum_{a=1}^{N} (\boldsymbol{F}\boldsymbol{S}\,\boldsymbol{\nabla}_0 \mathcal{N}_a)\,\delta\boldsymbol{v}_a, \end{aligned} \tag{III.22}$$

with $\{\delta\boldsymbol{v}_a\}_{a=1}^N$ the nodal virtual velocities. Injecting Eq. (III.22) into Eq. (III.3) and recalling that Eq. (III.3) holds for any combination of nodal virtual velocities, the discrete form of the principle of virtual work is then defined as the nullity of the discretized residual force vector $\boldsymbol{R} = \{\boldsymbol{R}_a\}_{a=1}^N$ on the degrees of freedom $\alpha \notin D$:

$$\boldsymbol{R}_a \cdot \boldsymbol{e}_i = 0, \; \forall (a,i) : \alpha(a,i) \in D \tag{III.23a}$$

with

$$\begin{aligned} \boldsymbol{R}_a = &\underbrace{\sum_{e=1}^{M} w_e \boldsymbol{F}_e \boldsymbol{S}_e \boldsymbol{\nabla}_0 \mathcal{N}_a(\boldsymbol{X}_e)}_{\boldsymbol{T}_a^{\mathrm{int}}} \\ &\underbrace{- \left(\sum_{e=1}^{M} w_e\, \mathcal{N}_a(\boldsymbol{X}_e) \boldsymbol{f}_0(\boldsymbol{X}_e) + \int_{\Gamma_{0_N}} \mathcal{N}_a(\boldsymbol{X}) \boldsymbol{t}_0(\boldsymbol{X}) \mathrm{d}A\right)}_{\boldsymbol{T}_a^{\mathrm{ext}}} = \boldsymbol{0}, \end{aligned} \tag{III.23b}$$

where $\boldsymbol{T}_a^{\mathrm{int}}$ (resp. $\boldsymbol{T}_a^{\mathrm{ext}}$) denotes the internal (resp. external) force vector[2]. The nullity of the residual force vector Eq. (III.23) together with the discrete form of the kinematic constraints Eq. (III.21) redefine the mechanical admissibility of the local strain and stress states as

$$\mathcal{E} = \{z \in \mathcal{Z} : \text{Eqs. (III.21) and (III.23)}\}. \tag{III.24}$$

[2]Note that $\mathcal{N}_a(\boldsymbol{X}_e)$ and $\boldsymbol{\nabla}_0 \mathcal{N}_a(\boldsymbol{X}_e)$ vanish whenever node a and integration point e are not connected (*i.e.* do not belong to the same element).

On the other hand, the material data set $\mathcal{D} \subset \mathcal{Z}$ for the finite element problem is easily derived as follows. For each integration point e, a local material data set comprises m_e strain-stress pairs:

$$\mathcal{D}_{\text{loc}}^{(e)} = \{(\boldsymbol{E}_i, \boldsymbol{S}_i)\,, i \in [1\,..\,m_e]\} \subset \mathcal{Z}_{\text{loc}}^{(e)}. \tag{III.25}$$

The global material data set is the collection of local state functions mapping every integration point to a sampling of the material behavior:

$$\mathcal{D} = \left\{ z \in \mathcal{Z} \mid \forall e = 1\,..\,M,\; z_e(\boldsymbol{X}_e) \in \mathcal{D}_{\text{loc}}^{(e)} \right\}. \tag{III.26}$$

This formulation allows for taking into account various materials in the structure. In the rest of the text, the terms *material database* are used to refer to local material data sets $\mathcal{D}_{\text{loc}}^{(e)}$, whereas the material data set refers to the functional set $\mathcal{D}$.

All of the above definitions and notations are gathered in the schematic representation of the physical and phase spaces given in Fig. III.2.

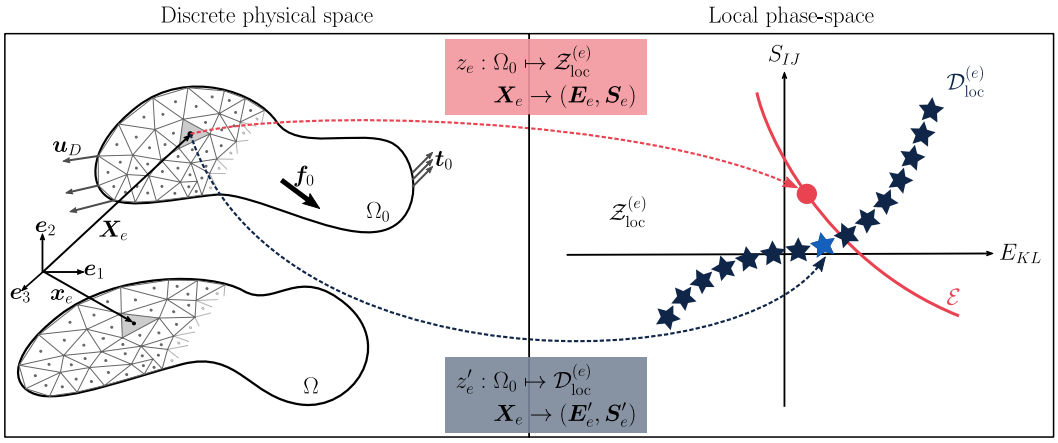


Figure III.2 – Functional spaces of the Lagrangian finite strain formulation. The mechanical state function z_e (in red) is a strain-stress field of the nonlinear constraint set $\mathcal{E}$ (verifying kinematics and equilibrium constraints) and the material state function z'_e (in light blue) maps to a strain-stress pair (light blue star), selected from the material database $\mathcal{D}_{\text{loc}}^{(e)}$ (dark blue stars).

Double minimization problem. The solution set in the finite element formulation can now be obtained from injecting the discrete evaluation of the metrics Eq. (III.20b) and the discrete definitions of the constraint and material data sets Eqs. (III.24) and (III.26) into the double minimization problem Eq. (III.10):

$$\mathcal{S} = \arg\min_{z \in \mathcal{E}} \; \min_{z' \in \mathcal{D}} \sum_{e=1}^{M} w_e \, d_e^2 \left(z_e, z'_e\right) \tag{III.27}$$

where d_e denotes the local deviation function in $\mathcal{Z}_{\text{loc}}^{(e)} \times \mathcal{D}_{\text{loc}}^{(e)}$, defined as

$$d_e\left(z_e, z_i\right) = W\left(\boldsymbol{E}_e - \boldsymbol{E}_i\right) + W^*\left(\boldsymbol{S}_e - \boldsymbol{S}_i\right), \quad \forall e \in [1\,..\,M],\ \forall z_e = (\boldsymbol{E}_e, \boldsymbol{S}_e) \in \mathcal{Z}_{\text{loc}}^{(e)},\ \forall z_i = (\boldsymbol{E}_i, \boldsymbol{S}_i) \in \mathcal{D}_{\text{loc}}^{(e)}, \tag{III.28}$$

with W and W^* as in Eq. (III.9).

It is worth mentioning here that the minimization Eq. (III.27) involves both continuous-valued functions $z \in \mathcal{Z}$ and discrete-valued functions $z' \in \mathcal{D}$, exactly like in the small strain case. However, the mechanical admissibility equations are now nonlinear:

- the compatibility Eq. (III.21a) is a quadratic function $\boldsymbol{E}_e(\boldsymbol{u})$ of the discrete displacement field $\boldsymbol{u} = \{\boldsymbol{u}_a\}_{a=1}^N$;
- the equilibrium Eq. (III.23) is a bilinear constraint on the discrete displacement and stress fields $\boldsymbol{u}$ and $\boldsymbol{S} = \{\boldsymbol{S}_e\}_{e=1}^M$.

To address the discrete/continuous coupling and the nonlinear constraints, specific numerical schemes must be employed, as detailed next.

2.1.4 Lagrangian data-driven solver

Alternated minimization. As in small strain, the data-driven finite strain formulation yields a minimization problem coupled with a combinatorial optimization. As aforementioned, Kanno (2019) showed that such classes of problems can fit in a MIQP formulation in the case of small strain. Here, however, the nonlinearity of the constraints prevents the use of such solvers in finite strain, to the best of our knowledge. In like manner, the max-ent solver developed in Kirchdoerfer and Ortiz (2017) relies on linear optimization constraints; adapting it to finite strain is out of the scope of the present work. This explains why we have chosen to use the original alternated minimization scheme only (see Fig. III.1).

The double minimization Eq. (III.34) is split into two distinct simple minimization problems, successively addressed in an iterative scheme:

- Let us first consider the stage where the *mechanical state*

$$z = \{z_e = (\boldsymbol{F}_e, \boldsymbol{P}_e)\}_{e=1}^M \in \mathcal{E} \tag{III.29}$$

is fixed (see stage 3 of the algorithm in Fig. III.1). The material solution set $\mathcal{S}^{\text{mat}}$ is then determined from Eq. (III.34) as

$$\mathcal{S}^{\text{mat}} = \arg\min_{z' \in \mathcal{D}} \sum_{e=1}^{M} w_e\, d_e^2\left(z_e, z'_e\right). \tag{III.30}$$

Note that $\mathcal{D} = \mathcal{D}_{\text{loc}}^{(1)} \times \cdots \times \mathcal{D}_{\text{loc}}^{(M)}$ is a product set of very high dimension $(\sum_{e=1}^M m_e)$ which makes the minimization problem Eq. (III.30) *a priori* intractable. However, as all terms in the summation are independent from one

another, the minimization can be conducted in each material database $\mathcal{D}^{(e)}_{\text{loc}}$ separately. The material solution set $\mathcal{S}^{\text{mat}}$ then results from M searches for nearest neighbors in local phase space:

$$\mathcal{S}^{\text{mat}}(z) = \left\{ \arg \min_{z_i \in \mathcal{D}^{(e)}_{\text{loc}}} d_e^2 \left(z_e, z_i\right) \;\middle|\; \forall e \in [1 \,..\, M] \right\}. \tag{III.31}$$

We refer to $z^*_{\text{ie}} = (\boldsymbol{E}^*_{\text{ie}}, \boldsymbol{S}^*_{\text{ie}})$ as the *material state* associated to e, defined as the material data point closest to the current local mechanical state $z_e = (\boldsymbol{F}_e, \boldsymbol{P}_e)$:

$$d_e\left(z_e, z^*_{\text{ie}}\right) \leq d_e\left(z_e, z_i\right), \forall e \in [1 \,..\, M], \forall z_i = (\boldsymbol{E}_i, \boldsymbol{S}_i) \in \mathcal{D}^{(e)}_{\text{loc}}, \tag{III.32}$$

with d_e as in Eq. (III.28). The integer index ie is then the actual optimization variable. It maps the e^{th} integration point to the optimal i^{th} state pair in $\mathcal{D}^{(e)}_{\text{loc}}$. This first minimization can also be considered as a projection of the mechanical states on the discrete material data set, with direction $\mathbb{C}$: $z^* = P_{\mathbb{C}}(\mathcal{E})$.

- Let us now consider the stage where the material state

$$z^* = \{z^*_{\text{ie}} = (\boldsymbol{F}^*_{\text{ie}}, \boldsymbol{P}^*_{\text{ie}})\}_{e=1}^{M} \in \mathcal{D} \tag{III.33}$$

is determined (stage 2 of the algorithm in Fig. III.1). Eq. (III.34) is then reduced to a *constrained* simple minimization problem. The solution set $\mathcal{S}^{\text{mec}}$ of mechanical states is then given as

$$\mathcal{S}^{\text{mec}}(z^*) = \arg\min_{z \in \mathcal{Z}} \sum_{e=1}^{M} w_e \, d_e^2 \left(z_e, z^*_{\text{ie}}\right) \tag{III.34a}$$

subject to $z \in \mathcal{E}$, *i.e.*

$$\boldsymbol{E}_e = \frac{1}{2}\left(\left(\boldsymbol{\nabla}^e_0 \boldsymbol{u}\right)^T + \boldsymbol{\nabla}^e_0 \boldsymbol{u} + \left(\boldsymbol{\nabla}^e_0 \boldsymbol{u}\right)^T \boldsymbol{\nabla}^e_0 \boldsymbol{u} \right), \qquad \forall e \in [1 \,..\, M], \tag{III.34b}$$

$$\boldsymbol{u}_a \cdot \boldsymbol{e}_i = u^{\text{D}}_\alpha, \qquad \forall (a,i) : \alpha(a,i) \in D, \tag{III.34c}$$

$$\boldsymbol{R}_a \cdot \boldsymbol{e}_i = \sum_{e=1}^{M} w_e F^e_{iJ} S^e_{JK} \frac{\partial \mathcal{N}_a}{\partial X_K}(\boldsymbol{X}_e) - \boldsymbol{T}^{\text{ext}}_a \cdot \boldsymbol{e}_i = 0, \qquad \forall (a,i) : \alpha(a,i) \notin D. \tag{III.34d}$$

Before addressing the computation of Eq. (III.34) in details, let us draw attention on a few implementation remarks.

1. The presented data-driven solver alternates between (i) solving the constrained minimization problem Eq. (III.34) and (ii) finding the optimal mapping i*e* according to the nearest-neighbor search problem Eq. (III.31). Before entering the iterative loop, all material states must be initialized (stage 1 in Fig. III.1). In the literature, authors have used a random initialization (Kirchdoerfer and Ortiz, 2016; Nguyen and Keip, 2018) but, in our case, the data-driven solver is embedded in an incremental loading loop. Other choices can thus be made and will be discussed in the numerical applications presented later on, in Section 3.

2. The alternated minimization scheme ends when the material states remain unchanged from one iteration (k) to the next $(k+1)$. In practice, we control this numerically by checking whether the integer mapping i*e* changes, which prevents from having to use a tolerance for the test $z^{*,(k+1)} = z^{*,(k)}$.

3. As aforementioned, stage 3 merely consists in a search for nearest neighbors in the local material data set. To perform this task, we use a tree-based nearest neighbor search algorithm, which drastically diminishes the computational cost of this part of the solver. Indeed, the local phase space $\mathbb{R}^{n\times n}_{\text{sym}} \times \mathbb{R}^{n\times n}_{\text{sym}}$ consisting of pairs of symmetric tensors in dimension n, it can be recast into the standard Euclidean space $\mathbb{R}^{2n(n-1)}$, using the Mandel vector representation of the strain and stress tensors (Brannon, 2018). In that case, the search for nearest neighbor is to be performed M times (for every integration point) in a database of m samples of dimension $2n(n-1)$. Using a brute-force search algorithm, this task grows as $O[Mm]$. With a tree-based data structure, the computational cost can be reduced to $O[M \log m]$ (Bentley, 1975). In practical DDCM applications, $m \gg M$ and the same material database is used for all integration points in the mesh (when a single material is considered); the tree-based data structure can be initialized once and for all at the beginning of the simulation. This approach thus provides a substantial gain in computation time. The interface that we use herein[3], requires the data to be transformed so that the data-driven distance d_e in local phase space $\mathbb{R}^{n\times n}_{\text{sym}} \times \mathbb{R}^{n\times n}_{\text{sym}}$ is equivalent to the Euclidean distance in $\mathbb{R}^{2n(n-1)}$. We refer to this transformation as the Euclidean mapping, which is detailed in Appendix B.

4. The core of the Lagrangian DDCM solver is then stage 2: computing the mechanical states. In finite strain, the resolution of system Eq. (III.34) is more complex than in small strain and is therefore detailed in the next paragraphs.

Formulation of a stationary problem. To solve the constrained minimization problem Eq. (III.34), we adopt a numerical method which enforces the constraints as strongly as possible, so as to always remain in the constraint set. On the one hand, the compatibility constraint Eq. (III.34b) is enforced by expressing strains in terms of displacements in the objective function Eq. (III.34a). In addition, the

[3] We use the `KDtree` class of the Python scikit-learn library, developed by Pedregosa et al. (2011).

Dirichlet boundary conditions Eq. (III.34c) are directly encoded in the definition of the displacement array. Kinematics constraints are then treated as classically. On the other hand, the equilibrium constraint Eq. (III.34d) is enforced by means of nodal Lagrange multipliers $\boldsymbol{\eta} = \{\boldsymbol{\eta}_a\}_{a=1}^N$, leading to the following stationary problem:

$$\delta \left(\sum_{e=1}^{M} w_e d_e^2 \left(z_e, z_{\mathrm{ie}}^* \right) - \sum_{a=1}^{N} \boldsymbol{\eta}_a \cdot \boldsymbol{R}_a \right) = 0, \tag{III.35}$$

where, as recommended by Nguyen et al. (2020), the nodal Lagrangian multipliers must vanish on the Dirichlet boundary:

$$\boldsymbol{\eta}_a \cdot \boldsymbol{e}_i = 0, \, \forall (a,i) : \alpha(a,i) \in D. \tag{III.36}$$

Consistently, the equilibrium constraint is not enforced on the Dirichlet boundary, or more precisely, it is verified *a posteriori* by adjusting the nodal reactions. The stationary equations with respect to each unknown $(\boldsymbol{u}_a, \boldsymbol{\eta}_a, \boldsymbol{S}_e)$ yield:

$$\delta \boldsymbol{S}_e \Rightarrow \mathbb{C}^{-1} \left(\boldsymbol{S}_e - \boldsymbol{S}_{\mathrm{ie}}^* \right) - \boldsymbol{F}_e^T \boldsymbol{\nabla}_0^e \boldsymbol{\eta} = \boldsymbol{0}, \, \forall e, \tag{III.37a}$$

$$\delta \boldsymbol{\eta}_a \Rightarrow \sum_e w_e \, \boldsymbol{F}_e \boldsymbol{S}_e \boldsymbol{B}_{ea} - \boldsymbol{T}_a^{\mathrm{ext}} = \boldsymbol{0}, \, \forall a, \tag{III.37b}$$

$$\begin{aligned} \delta \boldsymbol{u}_a \Rightarrow & \sum_e w_e \, \boldsymbol{F}_e \left(\mathbb{C} : \left(\boldsymbol{E}_e - \boldsymbol{E}_{\mathrm{ie}}^* \right) \right) \boldsymbol{B}_{ea} \\ & - \sum_e w_e \, \boldsymbol{\nabla}_0^e \boldsymbol{\eta} \, \boldsymbol{S}_e \boldsymbol{B}_{ea} - \sum_b \boldsymbol{\eta}_b \cdot \frac{\partial \boldsymbol{T}_b^{\mathrm{ext}}}{\partial \boldsymbol{u}_a} = \boldsymbol{0}, \, \forall a, \end{aligned} \tag{III.37c}$$

where the vector $\boldsymbol{B}_{ea} \in \mathbb{R}^n$ stands for $\boldsymbol{\nabla}_0 \mathcal{N}_a(\boldsymbol{X}_e)$, and $\boldsymbol{\nabla}_0^e \boldsymbol{\eta} = \sum_{a=1}^N \left(\boldsymbol{\eta}_a \otimes \boldsymbol{B}_{ea} \right)$ denotes the equivalent gradient of the Lagrangian multiplier vector, although $\boldsymbol{\eta}$ is defined by nodal quantities only. In the following, we restrict ourselves to problems where the external loads do not depend on the motion (typically in presence of body forces only or in absence of surface tractions). Consequently, the summation term $\sum_{b=1}^N \boldsymbol{\eta}_b \cdot \partial \boldsymbol{T}_b^{\mathrm{ext}} / \partial \boldsymbol{u}_a$ in Eq. (III.37c) vanishes. Further manipulations of equations Eq. (III.37) yield two nonlinear and coupled pseudo-elastic systems in $(\boldsymbol{u}, \boldsymbol{\eta})$, written as

$$\boldsymbol{R}_a^{\eta} \left(\boldsymbol{u}, \boldsymbol{\eta} \right) \cdot \boldsymbol{e}_i = 0, \qquad \forall (a,i) : \alpha(a,i) \notin D \tag{III.38a}$$

$$\boldsymbol{R}_a^{u} \left(\boldsymbol{u}, \boldsymbol{\eta} \right) \cdot \boldsymbol{e}_i = 0, \qquad \forall (a,i) : \alpha(a,i) \notin D \tag{III.38b}$$

where

$$\begin{aligned} \boldsymbol{R}_a^{\eta} \left(\boldsymbol{u}, \boldsymbol{\eta} \right) = & \sum_e w_e \, \boldsymbol{F}_e \left(\mathbb{C} : \left(\boldsymbol{F}_e^T \boldsymbol{\nabla}_0^e \boldsymbol{\eta} \right) \right) \boldsymbol{B}_{ea} \\ & + \sum_e w_e \, \boldsymbol{F}_e \boldsymbol{S}_{\mathrm{ie}}^* \boldsymbol{B}_{ea} - \boldsymbol{T}_a^{\mathrm{ext}}, \end{aligned} \tag{III.39a}$$

$$\boldsymbol{R}_a^u(\boldsymbol{u},\boldsymbol{\eta}) = \sum_e w_e \boldsymbol{F}_e \left(\mathbb{C} : (\boldsymbol{E}_e - \boldsymbol{E}_{\mathrm{ie}}^*)\right) \boldsymbol{B}_{ea} - \sum_e w_e \boldsymbol{\nabla}_0^e \boldsymbol{\eta} \left(\mathbb{C} : \left(\boldsymbol{F}_e^T \boldsymbol{\nabla}_0^e \boldsymbol{\eta}\right)\right) \boldsymbol{B}_{ea} - \sum_e w_e \boldsymbol{\nabla}_0^e \boldsymbol{\eta}\, \boldsymbol{S}_{\mathrm{ie}}^* \boldsymbol{B}_{ea} \quad \text{(III.39b)}$$

in which the stress $\boldsymbol{S}_e$ has been replaced by

$$\boldsymbol{S}_e(\boldsymbol{u},\boldsymbol{\eta}) = \mathbb{C} : \left(\boldsymbol{F}_e^T \boldsymbol{\nabla}_0^e \boldsymbol{\eta}\right) + \boldsymbol{S}_{\mathrm{ie}}^*, \ \forall e = 1 \,..\, M. \quad \text{(III.40)}$$

Several comments on system Eq. (III.38) are to be made:

- First, the two pseudo-elastic linear systems together with the stress correction derived in Kirchdoerfer and Ortiz (2016) are easily recovered with linearized kinematics:

$$\sum_e w_e \ (\mathbb{C} : \boldsymbol{\nabla}\boldsymbol{\eta})\, \boldsymbol{B}_{ea} = \boldsymbol{T}_a^{\mathrm{ext}} - \sum_e w_e \ \boldsymbol{\sigma}_{\mathrm{ie}}^* \boldsymbol{B}_{ea}, \qquad \forall a, \quad \text{(III.41a)}$$

$$\sum_e w_e \ (\mathbb{C} : \boldsymbol{\nabla}\boldsymbol{u})\, \boldsymbol{B}_{ea} = \sum_e w_e \ (\mathbb{C} : \boldsymbol{\varepsilon}_{\mathrm{ie}}^*)\, \boldsymbol{B}_{ea}, \qquad \forall a, \quad \text{(III.41b)}$$

$$\boldsymbol{\sigma}_e(\boldsymbol{\eta}) = \mathbb{C} : \boldsymbol{\nabla}\boldsymbol{\eta} + \boldsymbol{\sigma}_{\mathrm{ie}}^*, \qquad \forall e. \quad \text{(III.41c)}$$

- Second, the mechanical stress $\boldsymbol{S}_e$ now results from a correction of the material stress $\boldsymbol{S}_{\mathrm{ie}}^*$ involving both the displacements $\boldsymbol{u}$ and the Lagrange multipliers $\boldsymbol{\eta}$ (see Eq. (III.40)). This double dependency emerges from the geometrical nonlinearity and yields a strong coupling of both residuals $\boldsymbol{R}^\eta$ and $\boldsymbol{R}^u$.
- Third, despite the data-driven problem Eq. (III.34) being formulated in terms of strain and stress, the discretization yields a resolution in terms of displacement and Lagrange multipliers.

Monolithic resolution of the stationary problem. In Nguyen and Keip (2018), the system Eq. (III.38) is addressed with a Newton-Raphson scheme. It requires to build and solve a new $(2n_{\mathrm{dof}} \times 2n_{\mathrm{dof}})$ linear system at every iteration of the Newton-Raphson scheme, at every iteration of the data-driven solver, with n_{dof} the total number of degrees of freedom. A staggered approach was proposed in Platzer et al. (2019) but it appeared that this approach could ensure the decrease of the objective function at every iteration. We then adopt the monolithic resolution scheme, which ensures that the constraints are always satisfied. A typical Newton-Raphson iteration writes

$$\boldsymbol{q}^{(j+1)} = \boldsymbol{q}^{(j)} + \Delta\boldsymbol{q} \quad \text{(III.42a)}$$

with

$$\boldsymbol{R}_a(\boldsymbol{q}^{(j)}) + \sum_{b=1}^{N} \boldsymbol{K}_{ab}(\boldsymbol{q}^{(j)}) \Delta\boldsymbol{q}_b = \boldsymbol{0}, \forall a, \quad \text{(III.42b)}$$

with $\boldsymbol{q} = \left[\boldsymbol{u}_1^T, \cdots \boldsymbol{u}_N^T, \boldsymbol{\eta}_1^T, \cdots \boldsymbol{\eta}_N^T\right]^T$, and the residual force vector $\boldsymbol{R}_a$ and tangent matrix $\boldsymbol{K}_{ab}$ relating node a and node b of shape $2n \times 2n$, respectively defined as

$$\boldsymbol{R}_a = \begin{pmatrix} \boldsymbol{R}_a^u \\ \\ \boldsymbol{R}_a^\eta \end{pmatrix} \quad \text{and} \quad \boldsymbol{K}_{ab} = \begin{pmatrix} \dfrac{\partial \boldsymbol{R}_a^u}{\partial \boldsymbol{u}_b} & \dfrac{\partial \boldsymbol{R}_a^u}{\partial \boldsymbol{\eta}_b} \\ \dfrac{\partial \boldsymbol{R}_a^\eta}{\partial \boldsymbol{u}_b} & \dfrac{\partial \boldsymbol{R}_a^\eta}{\partial \boldsymbol{\eta}_b} \end{pmatrix}. \tag{III.43}$$

Evidently, $\Delta \boldsymbol{q}^{(j)}$ is determined by a reduction of the system Eq. (III.42b) to the degrees of freedom $\alpha(a,i) \times \beta(b,j) \notin D \times D$ only. The expression of every block is detailed in the following, assuming that $\mathbb{C}$ is a fully symmetric tensor ($\mathbb{C}_{IJKL} = \mathbb{C}_{IJLK}$ in particular). First,

$$\frac{\partial \boldsymbol{R}_a^u}{\partial \boldsymbol{u}_b} = \boldsymbol{K}_{ab}^{\mathbb{C}} + \boldsymbol{K}_{ab}^{\Delta E} - \boldsymbol{K}_{ab}^{\eta}, \tag{III.44a}$$

where

$$\left[\boldsymbol{K}_{ab}^{\mathbb{C}}\right]_{ij} = \sum_e w_e \, F_{iI}^e \, B_J^{ea} \, \mathbb{C}_{IJKL} \, B_K^{eb} \, F_{jL}^e, \tag{III.44b}$$

$$\boldsymbol{K}_{ab}^{\Delta E} = \sum_e w_e \, \left(\boldsymbol{B}_{eb} \cdot (\mathbb{C} : \Delta \boldsymbol{E}_e) \, \boldsymbol{B}_{ea}\right) \boldsymbol{I}, \tag{III.44c}$$

$$\left[\boldsymbol{K}_{ab}^{\eta}\right]_{ij} = \sum_e w_e \sum_{IJKL} \left[\boldsymbol{\nabla}_0^e \boldsymbol{\eta}\right]_{iI} \, B_J^{ea} \, \mathbb{C}_{IJKL} \, B_K^{eb} \, \left[\boldsymbol{\nabla}_0^e \boldsymbol{\eta}\right]_{jL}. \tag{III.44d}$$

The term $\boldsymbol{K}_{ab}^{\mathbb{C}}$ defined in Eq. (III.44b) retrieves the constitutive component of the standard tangent matrix emerging from the Lagrangian FE formulation of classical hyperelasticity, where $\mathbb{C}$ plays the role of the material elasticity tensor (see Eq. (I.88) in Table I.4). The term $\boldsymbol{K}_{ab}^{\Delta E}$ defined in Eq. (III.44c) resembles the initial stress component of the standard tangent matrix where $\mathbb{C} : \Delta \boldsymbol{E}_e = \mathbb{C} : (\boldsymbol{E}_e - \boldsymbol{E}_{\text{ie}}^*)$ can be seen as a reference stress from which derives the reference stiffness $\mathbb{C}$ (see Eq. (I.89) in Table I.4). Furthermore, the anti-diagonal blocks are defined as

$$\left[\frac{\partial \boldsymbol{R}_a^u}{\partial \boldsymbol{\eta}_b}\right]_{ij} = -\left[\boldsymbol{K}_{ab}^S\right]_{ij} - \sum_e w_e \, \left[\boldsymbol{\nabla}_0^e \boldsymbol{\eta}\right]_{iI} \left(B_J^{ea} \, \mathbb{C}_{IJKL} \, B_K^{eb}\right) F_{jL}^e \tag{III.45a}$$

where

$$\boldsymbol{K}_{ab}^S = \sum_e w_e \, \left(\boldsymbol{B}_{eb} \cdot \boldsymbol{S}_e \boldsymbol{B}_{ea}\right) \boldsymbol{I}, \tag{III.45b}$$

with

$$\boldsymbol{S}_e = \mathbb{C} : \left(\boldsymbol{F}_e^T \boldsymbol{\nabla}_0^e \boldsymbol{\eta}\right) + \boldsymbol{S}_{\text{ie}}^*,$$

and

$$\left[\frac{\partial \boldsymbol{R}_a^\eta}{\partial \boldsymbol{u}_b}\right]_{ij} = \left[\boldsymbol{K}_{ab}^S\right]_{ij} + \sum_e w_e \, F_{iI}^e \left(B_J^{ea} \, \mathbb{C}_{IJKL} \, B_K^{eb}\right) \left[\boldsymbol{\nabla}_0^e \boldsymbol{\eta}\right]_{jL}. \tag{III.46}$$

Here, the tangent matrix $\boldsymbol{K}^S_{ab}$ defined in Eq. (III.45b) exactly corresponds to the initial stress component derived in the classical FE approach, where the stress is no longer obtained from a constitutive model but from the correction of the corresponding material stress $\boldsymbol{S}^*_{ie}$ Eq. (III.40). Finally,

$$\frac{\partial \boldsymbol{R}^\eta_a}{\partial \boldsymbol{\eta}_b} = \boldsymbol{K}^{\mathbb{C}}_{ab}. \tag{III.47}$$

As a remark, the complete tangent matrix $\boldsymbol{K}_{ab}$ needs to be updated at every iteration of the Newton-Raphson scheme. However, all block matrices defined in Eqs. (III.44) to (III.47) contain one constant term which can be computed once and for all at the integration point level:

$$\left[\boldsymbol{c}^{(e)}_{ab}\right]_{IL} = B^{ea}_J \, \mathbb{C}_{IJKL} \, B^{ea}_K. \tag{III.48}$$

Summary. The finite strain data-driven solver resulting from the Lagrangian formulation derived above is detailed in Algorithm III.1. The key ingredients are:

- the computation of the mechanical states in two steps, namely
  - the resolution of the nonlinear coupled systems Eqs. (III.39a) and (III.39b) with a Newton-Raphson scheme, as proposed by Nguyen and Keip (2018),
  - the computation of local strain and stress states with (i) the compatibility Eq. (III.21a), (ii) the correction of the stress Eq. (III.40);
- the nearest-neighbor search for the optimal states in the material database from Eq. (III.31), using a tree-based search algorithm.

2.2 Nominal approach

The nominal formulation has first been introduced by Conti et al. (2020) to extend the data-driven framework defined in (Conti et al., 2018) to finite strain elasticity. The phase space now consists of deformation gradient–first Piola-Kirchhoff stress tensor fields $(\boldsymbol{F}, \boldsymbol{P})$. Hence, the local phase space can no longer be reduced to $\mathbb{R}^{n\times n}_{\text{sym}} \times \mathbb{R}^{n\times n}_{\text{sym}}$, which enabled to implicitly enforces conservation of angular momentum and material-frame indifference in geometrically linear elasticity and Lagrangian finite strain elasticity. In Conti et al. (2020), the former is encoded in the material data set by using an orbit representation: if a material point $(\boldsymbol{F}, \boldsymbol{P})$ is in the database, then so is its orbit $(\boldsymbol{QF}, \boldsymbol{QP})$, $\boldsymbol{Q} \in SO(n)$ under the left action of the Special Orthogonal group $SO(n) = \left\{\boldsymbol{Q} \in \mathbb{R}^{n\times n} \| \; \boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{Q}\boldsymbol{Q}^T = \boldsymbol{I}\right\}$. The latter can be treated as an additional constraint to the minimization problem, as discussed in the following.

This section is dedicated to recalling the continuum mechanics formulation as introduced by Conti et al. (2020) and developing a FE data-driven solver on that basis, analogous to the one derived above for the Lagrangian formulation.

Algorithm III.1 Data-driven solver in finite strain - Lagrangian formulation.

INPUT: A mesh of M integration points and N nodes, boundary and loading conditions, material data set $\mathcal{D}$.

OUTPUT: mechanical response (displacements $\{\boldsymbol{u}_a\}_{a=1}^{N}$, mechanical states $\{(\boldsymbol{E}_e, \boldsymbol{S}_e)\}_{e=1}^{M}$, residual forces, *etc.*), material states $\{(\boldsymbol{E}^*_{\mathsf{ie}}, \boldsymbol{S}^*_{\mathsf{ie}})\}_{e=1}^{M}$.

Require: tolerance parameter $r_{\text{tol}} > 0$, maximum number of iterations $k_{\max}$ and $j_{\max}$.

1) Initialize all material states and mapping

$k \leftarrow 0$
for $e = 1 \,..\, M$ **do**
  $\mathsf{ie}^{(0)} \leftarrow i,\ i \in [1 \,..\, m_e]$
end for

$\boldsymbol{q}^{(k=0)} \leftarrow \boldsymbol{0}$
for $k = 1 \,..\, k_{\max}$ **do**
  2) Compute mechanical states
  2)a. Solve Eqs. (III.39a) *and* (III.39b) *with a Newton-Raphson scheme*
  $j \leftarrow 0$
  $\boldsymbol{q}^{(j=0)} \leftarrow \boldsymbol{q}^{(k-1)}$ ▶ *store previous unknowns*
  $\boldsymbol{R}^{(j=0)} \leftarrow \boldsymbol{R}\left(\boldsymbol{q}^{(j=0)}\right)$ ▶ *compute initial residual*
  $R^{(j=0)} \leftarrow \|\boldsymbol{R}^{(j=0)}\|_2$
  repeat ▶ *Newton-Raphson iterations*
    $j \leftarrow j + 1$
    $\boldsymbol{q}^{(j)} \leftarrow$ solve Eq. (III.42)
    $\boldsymbol{R}^{(j)} \leftarrow \boldsymbol{R}(\boldsymbol{q}^{(j)})$ ▶ *update residual*
    $R^{(j)} \leftarrow \|\boldsymbol{R}^{(j)}\|_2$ ▶ *update residual norm using degrees of freedom only*
  until $R^{(j)} \leq r_{\text{tol}} R^{(0)}$ **or** $j \geq j_{\max}$ ▶ *convergence within* r_{tol} *or* $j_{\max}$
  $\boldsymbol{q}^{(k)} \leftarrow \boldsymbol{q}^{(j)}$
  2)b. Update mechanical states
  for $e = 1 \,..\, M$ **do**
    $\boldsymbol{E}_e^{(k)} \leftarrow \frac{1}{2}\left(\boldsymbol{F}_e^T(\boldsymbol{u}^{(k)})\boldsymbol{F}_e(\boldsymbol{u}^{(k)}) - \boldsymbol{I}\right)$
    $\boldsymbol{S}_e^{(k)} \leftarrow \mathbb{C} : \left(\boldsymbol{F}_e^T(\boldsymbol{u}^{(k)})\boldsymbol{\nabla}_0^e\boldsymbol{\eta}^{(k)}\right) + \boldsymbol{S}^*_{\mathsf{ie}^{(k-1)}}$
  end for
  3) Update material states and mapping
  for $e = 1 \,..\, M$ **do**
    $\mathsf{ie}^{(k)} \leftarrow \left(\boldsymbol{E}^*_{\mathsf{ie}^{(k)}}, \boldsymbol{S}^*_{\mathsf{ie}^{(k)}}\right)$ from Eq. (III.32)
  end for
  4) Test convergence
  if $\mathsf{ie}^{(k)} = \mathsf{ie}^{(k-1)}$ for all $e = 1 \,..\, M$ **then**
    $\boldsymbol{u} \leftarrow \boldsymbol{u}^{(k)}$
    $(\boldsymbol{E}_e, \boldsymbol{S}_e) \leftarrow (\boldsymbol{E}_e^{(k)}, \boldsymbol{S}_e^{(k)})$ for all $e = 1 \,..\, M$
    exit.
  else
    $\mathsf{ie}^{(k+1)} \leftarrow \mathsf{ie}^{(k)}$ for all $e = 1 \,..\, M$
  end if
end for

2.2.1 Continuum mechanics formulation

Let us again consider an elastic body occupying a domain $\Omega_0 \in \mathbb{R}^n$ in the reference configuration and $\Omega \in \mathbb{R}^n$ under quasi-static loading, in dimension $n = 3$. A material point has $\boldsymbol{X}$ and $\boldsymbol{x}$ as initial and current coordinates, respectively. Like the Lagrangian formulation, the deformation gradient and first Piola-Kirchhoff stress tensor fields are defined in Ω_0. The local state function z then maps every material point $\boldsymbol{X}$ of Ω_0 to its nominal local state:

$$\begin{aligned} z: \quad & \Omega_0 \mapsto \mathcal{Z}_{\text{loc}} \\ & \boldsymbol{X} \to z(\boldsymbol{X}) = (\boldsymbol{F}(\boldsymbol{X}), \boldsymbol{P}(\boldsymbol{X})) \end{aligned} \tag{III.49}$$

where the local phase space $\mathcal{Z}_{\text{loc}}$ is now $\mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n}$. The global phase space $\mathcal{Z}$ is again the collection of local state functions $z : \Omega_0 \to \mathcal{Z}_{\text{loc}}$. As shown in Conti et al. (2020), to ensure convergence properties, the deformation gradient and the stress fields must each belong to a given L^p space, *i.e.* they must be measurable functions for which the $p-$th power of the absolute value is Lesbegue integrable. The global phase space is then defined as

$$\mathcal{Z}_{p,q}(\Omega_0) = L^p\left(\Omega_0, \mathbb{R}^{n\times n}\right) \times L^q\left(\Omega_0, \mathbb{R}^{n\times n}\right) \tag{III.50}$$

with $(p, q) \in (1, \infty)$ such that $1/p + 1/q = 1$. As aforementioned, the local phase space no longer consists of symmetric tensors, as compared with the small strain or Lagrangian finite strain settings. Material frame-indifference and conservation of angular momentum constraints must then be enforced specifically.

The material data set $\mathcal{D}$ is defined by the collection of local state functions mapping to nominal strain-stress pairs of a local material database $\mathcal{D}_{\text{loc}}$:

$$\mathcal{D} = \{z \in \mathcal{Z} \mid z(\boldsymbol{X}) \in \mathcal{D}_{\text{loc}}\}, \tag{III.51a}$$

with

$$\mathcal{D}_{\text{loc}} = \{(\boldsymbol{F}_i, \boldsymbol{P}_i), i \in [1\,..\,m]\} \subset \mathcal{Z}_{\text{loc}}, \tag{III.51b}$$

with m the number of material data points. As aforementioned, Conti et al. (2020) proposed an orbit representation of local material data sets $\mathcal{D}_{\text{loc}}$ which are material-frame indifferent, *i.e.* every point in the set satisfies material-frame indifference:

$$\mathcal{D}_{\text{loc}} = \{(\boldsymbol{QF}, \boldsymbol{QP}) \mid \boldsymbol{Q} \in SO(n), (\boldsymbol{F}, \boldsymbol{P}) \in \mathcal{U}_{\text{loc}}\} \tag{III.52}$$

with $\mathcal{U}_{\text{loc}} \subseteq \mathbb{R}^{n\times n}_{\text{sym}} \times \mathbb{R}^{n\times n}$. In addition, the authors remark that local data sets that are generated by a hyperelastic constitutive relation of the type

$$\boldsymbol{P}(\boldsymbol{F}) = D\Psi(\boldsymbol{F}) = \frac{\partial \Psi(\boldsymbol{F})}{\partial \boldsymbol{F}} \tag{III.53}$$

such that

$$\Psi(\boldsymbol{F}) = \Psi(\boldsymbol{U}), \text{ with } \boldsymbol{F} = \boldsymbol{R}\boldsymbol{U}, \tag{III.54}$$

are both material-frame indifferent and satisfy moment equilibrium. Such a *graph local material data set* writes

$$\mathcal{D}_{\text{loc}} = \left\{ (\boldsymbol{F}, D\Psi(\boldsymbol{F})) \mid \boldsymbol{F} \in \mathbb{R}^{n\times n} \right\}. \tag{III.55}$$

The constraint set $\mathcal{E}$ consists of local state functions verifying mechanical admissibility. It now comprises three constraints, namely

1. the kinematics constraints, defined as

$$\boldsymbol{F}(\boldsymbol{X}) = \boldsymbol{\nabla}_0 \boldsymbol{u}(\boldsymbol{X}) + \boldsymbol{I} \qquad \text{in } \Omega_0, \tag{III.56a}$$
$$\boldsymbol{u}(\boldsymbol{X}) = \boldsymbol{u}_D(\boldsymbol{X}) \qquad \text{on } \Gamma_{0_D}, \tag{III.56b}$$

2. the translational equilibrium, written as

$$\text{DIV}\, \boldsymbol{P}(\boldsymbol{X}) + \boldsymbol{f}_0(\boldsymbol{X}) = \boldsymbol{0} \qquad \text{in } \Omega_0, \tag{III.57a}$$
$$\boldsymbol{P}(\boldsymbol{X})\boldsymbol{N}(\boldsymbol{X}) = \boldsymbol{t}_0(\boldsymbol{X}) \qquad \text{on } \Gamma_{0_N}, \tag{III.57b}$$

3. the rotational equilibrium, written as

$$\boldsymbol{F}\boldsymbol{P}^T = \boldsymbol{P}^T\boldsymbol{F}, \text{ in } \Omega_0, \tag{III.58}$$

using the same notations as above and in Chapter I. The set $\mathcal{E}$ then writes

$$\mathcal{E} = \{z \in \mathcal{Z} \mid \text{Eqs. (III.56) and (III.58)}\}. \tag{III.59}$$

Note that the first two constraints Eqs. (III.56) and (III.57) are linear while the rotational equilibrium Eq. (III.58) is a bi-linear constraint and more difficult to enforce in a discrete setting. It can then be more convenient to work with the affine subspace $\mathcal{E}_0 \subset \mathcal{E}$, defined as the collection of local state functions verifying compatibility and translational equilibrium Eqs. (III.56) and (III.57) only:

$$\mathcal{E}_0 = \{z \in \mathcal{Z} \mid \text{Eqs. (III.56) and (III.57)}\} \subset \mathcal{E} \tag{III.60}$$

For later purposes, the translational equilibrium and attached Neumann conditions Eq. (III.57) are reformulated in their weak form. The nominal formulation of the principle of virtual work writes (see Eq. (I.27) in Chapter I):

$$\delta W = \int_{\Omega_0} \boldsymbol{P} : \delta\dot{\boldsymbol{F}}\, \mathrm{d}V - \int_{\Omega_0} \boldsymbol{f}_0 \cdot \delta\boldsymbol{v}\, \mathrm{d}V - \int_{\partial\Omega_0} \boldsymbol{t}_0 \cdot \delta\boldsymbol{v}\, \mathrm{d}A = 0, \tag{III.61}$$

for any arbitrary kinematically admissible virtual velocity $\delta\boldsymbol{v}$ from the current position of the body. In addition, the conservation of angular momentum Eq. (III.58) is more conveniently expressed by the vector equation

$$\boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}(\boldsymbol{X})\boldsymbol{P}^T(\boldsymbol{X})\right) = \boldsymbol{0}, \quad \forall \boldsymbol{X} \in \Omega_0, \tag{III.62}$$

where the bold-face symbol $\boldsymbol{\mathcal{E}}$ denotes the Levi-Civita third-order tensor, *c.f.* Chapter I. The solution set $\mathcal{S}$ of the data-driven BVP corresponds to the set of local state functions verifying mechanical admissibility meanwhile being closest to the material data set. It then results from the double minimization problem

$$\min_{z\in\mathcal{E}} \min_{z'\in\mathcal{D}} \quad d(z,z') \tag{III.63}$$

as

$$\mathcal{S} = \arg\min_{z\in\mathcal{E}} \min_{z'\in\mathcal{D}} \quad d(z,z'), \tag{III.64}$$

with d an appropriate deviation or distance function from the material data set. The deviation function proposed by Conti et al. (2020) is

$$d(z,z') = \int_{\Omega_0} \left(V\left(\boldsymbol{F}(\boldsymbol{X}) - \boldsymbol{F}'(\boldsymbol{X})\right) + V^*\left(\boldsymbol{P}(\boldsymbol{X}) - \boldsymbol{P}'(\boldsymbol{X})\right)\right) \mathrm{d}\boldsymbol{X}, \tag{III.65}$$
$$\forall (z : \boldsymbol{X} \mapsto (\boldsymbol{F},\boldsymbol{P}),\, z' : \boldsymbol{X} \mapsto (\boldsymbol{F}',\boldsymbol{P}')) \in \mathcal{Z}\times\mathcal{Z}.$$

where V is a convex function, with convex conjugate V^*. An appropriate choice for V and V^* is (see Conti et al. (2020, Assumption 2.15)):

$$V(\boldsymbol{F}) = \frac{1}{p}|\boldsymbol{F}|^p = \frac{1}{p}\left(\operatorname{tr}\boldsymbol{F}^T\boldsymbol{F}\right)^{p/2}, \tag{III.66a}$$

$$V^*(\boldsymbol{P}) = \frac{1}{q}|\boldsymbol{P}|^q = \frac{1}{q}\left(\operatorname{tr}\boldsymbol{P}^T\boldsymbol{P}\right)^{q/2}. \tag{III.66b}$$

The framework derived by Conti et al. (2020), and recalled above, ensures the existence of so-called *classical solutions* to the minimization problem, provided specific conditions are verified by the local material data set $\mathcal{D}_{\text{loc}}$. A classical solution is met when the mechanical state field $z \in \mathcal{E}$ and the material state field $z' \in \mathcal{D}$ coincide such that the minimum of $d(z,z')$ is 0. This typically arises in cases where the material data set is achieved from a hyperelastic potential as in Eq. (III.55). For more details on the conditions of existence of classical solutions and examples of such material data sets, the reader is referred to Conti et al. (2020).

Note that, following the same rationale as in Section 2.1.2, the data-driven nominal setting derived above is equally valid for three-dimensional problems ($n = 3$) and two-dimensional problems ($n = 2$) arising from plane strain or plane stress conditions.

2.2.2 Finite element formulation

In Conti et al. (2020), the existence of *generalized* data-driven solutions, with $z \neq z'$, is not addressed. These solutions are expected when the local material data set is discrete, *i.e.* contains a finite number of material data points, as is the case in typical DDCM applications. To evaluate the corresponding data-driven solutions, we derive in the following a finite element formulation of the data-driven nominal BVP formulated by Conti et al. (2020). We must emphasize that to date, and to the best of our knowledge, this is the first attempt to such a derivation.

Preliminary considerations. We slightly modify the deviation functions defined in Eq. (III.66) to better suit them to numerical implementation. We use a scalar C to scale deformation gradient and stress values; without loss of generality, V and V^* are now defined as

$$V(\boldsymbol{F}) = \frac{1}{p}\, C^{p/2}\, |\boldsymbol{F}|^p, \text{ and } V^*(\boldsymbol{P}) = \frac{1}{q}\, C^{-q/2}\, |\boldsymbol{P}|^q. \tag{III.67}$$

In principle, the values of (p, q) and C could be optimized as well, as their optimal values depend on the BVP and the material data set (see examples provided in Conti et al. (2020)). In practice, taking C as the linearized stiffness of the material (when $\boldsymbol{F} = \boldsymbol{I}$) was found to be a reasonable choice. The values of (p, q) are discussed later on.

Global phase space and metrics. As in the Lagrangian formulation, we transition from the continuum mechanics to the finite element setting by interpolating the displacement field on the mesh as in Eq. (III.16) and by evaluating the stress field $\boldsymbol{P}$ at the integration points.

We consider again a FE mesh of N nodes, labeled $a \in [1\,..\,N]$, and a total of M integration points, labeled $e \in [1\,..\,M]$. Then, the approximate deformation gradient tensor writes:

$$\boldsymbol{F}(\boldsymbol{X}) = \boldsymbol{\nabla}_0 \boldsymbol{u}\,(\boldsymbol{X}) + \boldsymbol{I}, \tag{III.68}$$

with $\boldsymbol{\nabla}_0 \boldsymbol{u}(\boldsymbol{X})$ as in Eq. (III.17b). The stresses (which are no longer related to the displacement field) are merely evaluated at the integration points:

$$\int_{\Omega_0} \boldsymbol{P}(\boldsymbol{X})\, \mathrm{d}V = \sum_{e=1}^{M} w_e \boldsymbol{P}_e \tag{III.69}$$

with $\boldsymbol{P}_e = \boldsymbol{P}(\boldsymbol{X}_e)$.

The FE local state function z_e then maps integration point e in the reference configuration to the corresponding local state pair:

$$\begin{array}{rccl} z_e: & \Omega_0 & \to \mathcal{Z}_{\mathrm{loc}}^{(e)} = \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n} \\ & \boldsymbol{X}_e & \mapsto (\boldsymbol{F}_e, \boldsymbol{P}_e)\,, \end{array} \tag{III.70}$$

with $\boldsymbol{F}_e = \boldsymbol{F}(\boldsymbol{X}_e)$ from Eq. (III.68). The global phase space being the product set $\mathcal{Z} = \mathcal{Z}_{\mathrm{loc}}^{(1)} \times \cdots \times \mathcal{Z}_{\mathrm{loc}}^{(M)}$, a global point z in $\mathcal{Z}$ is the collection of M local state functions: $z = \{z_e\}_{e=1}^{M}$. The global deviation function Eq. (III.65) is then evaluated on the discrete mesh as

$$\begin{aligned} d(z, z') = \sum_{e=1}^{M} w_e \left(V\left(\boldsymbol{F}_e - \boldsymbol{F}'_e\right) + V^*\left(\boldsymbol{P}_e - \boldsymbol{P}'_e\right)\right), \\ \forall \left(z = \{z_e = (\boldsymbol{F}_e, \boldsymbol{P}_e)\}_{e=1}^{M},\, z' = \left\{z'_e = \left(\boldsymbol{F}'_e, \boldsymbol{P}'_e\right)\right\}_{e=1}^{M}\right) \in \mathcal{Z}\times\mathcal{Z}, \end{aligned} \tag{III.71}$$

with V and V^* as in Eq. (III.67).

Constraint set and material data set. In like manner as the Lagrangian FE formulation, the discretized subspaces $\mathcal{E}$ and $\mathcal{D}$ of $\mathcal{Z}$ are obtained by replacing the deformation gradient and the stress tensor fields with their respective discrete evaluations Eqs. (III.68) and (III.69).

On the one hand, the mechanical admissibility constraints are discretized as follows

1. The kinematic constraints Eq. (III.56) write

$$\boldsymbol{F}_e = \boldsymbol{\nabla}_0^e \boldsymbol{u} + \boldsymbol{I}, \qquad \forall e \in [1\,..\,M], \tag{III.72a}$$
$$\boldsymbol{u}_a \cdot \boldsymbol{e}_i = u_\alpha^{\mathrm{D}}, \qquad \forall (a,i) : \alpha(a,i) \in D. \tag{III.72b}$$

   where, as a reminder, u_α^{D} is the prescribed displacement on node a along direction i, such that $\alpha = \alpha(a,i)$.

2. The translational equilibrium Eq. (III.57) is again enforced in its weak form Eq. (III.61), written as

$$\boldsymbol{R}_a \cdot \boldsymbol{e}_i = 0, \qquad \forall (a,i) : \alpha(a,i) \notin D, \tag{III.73a}$$

   with

$$\boldsymbol{R}_a = \sum_{e=1}^{M} w_e \, \boldsymbol{P}_e \boldsymbol{B}_{ea} - \boldsymbol{T}_a^{\mathrm{ext}}, \qquad \forall a \in [1\,..\,N]. \tag{III.73b}$$

   where we recall that $\boldsymbol{B}_{ea} = \boldsymbol{\nabla}_0 \mathcal{N}_a(\boldsymbol{X}_e)$.

3. As for the conservation of angular momentum, Eq. (III.58) and its vector form Eq. (III.62) hold at every integration point. Thus, the third mechanical constraint simply writes

$$\boldsymbol{r}_e = \boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}_e \boldsymbol{P}_e^T\right) = \boldsymbol{0}, \quad \forall e \in [1\,..\,M], \tag{III.74}$$

   with $\boldsymbol{r}_e$ the local rotational residual at integration point e.

The set of mechanically admissible FE local state functions is then

$$\mathcal{E} = \{ z \in \mathcal{Z} \mid \text{Eqs. (III.72) to (III.74)} \}. \tag{III.75}$$

On the other hand, the material data set $\mathcal{D} \subset \mathcal{Z}$ for the finite element problem is easily derived as previously. For each integration point e, a material database $\mathcal{D}_{\mathrm{loc}}^{(e)}$ comprises m_e deformation gradient-stress pairs and the material data set is the collection of local state functions mapping every integration point to a material data point in $\mathcal{D}_{\mathrm{loc}}^{(e)}$:

$$\mathcal{D} = \left\{ z \in \mathcal{Z} \mid \forall e \in [1\,..\,M],\, z_e(\boldsymbol{X}_e) \in \mathcal{D}_{\mathrm{loc}}^{(e)} \right\}, \tag{III.76a}$$

with

$$\mathcal{D}_{\mathrm{loc}}^{(e)} = \{ (\boldsymbol{F}_i, \boldsymbol{P}_i) \mid i \in [1\,..\,m_e] \} \subset \mathcal{Z}_{\mathrm{loc}}^{(e)}. \tag{III.76b}$$

Double minimization problem. As for the Lagrangian formulation, the nominal data-driven BVP in the finite element setting writes as the double minimization problem

$$\mathcal{S} = \arg\min_{z\in\mathcal{E}} \min_{z'\in\mathcal{D}} \sum_{e=1}^{M} w_e \, d_e\left(z_e, z'_e\right) \tag{III.77}$$

where d_e denotes the local deviation function in $\mathcal{Z}_{\text{loc}}^{(e)} \times \mathcal{D}_{\text{loc}}^{(e)}$, now defined as

$$\begin{aligned} d_e\left(z_e, z_i\right) &= V\left(\boldsymbol{F}_e - \boldsymbol{F}_i\right) + V^*\left(\boldsymbol{P}_e - \boldsymbol{P}_i\right), \\ &\forall e \in [1\,..\,M],\ \forall z_e = (\boldsymbol{F}_e, \boldsymbol{P}_e) \in \mathcal{Z}_{\text{loc}}^{(e)},\ \forall z_i = (\boldsymbol{F}_i, \boldsymbol{P}_i) \in \mathcal{D}_{\text{loc}}^{(e)}, \end{aligned} \tag{III.78}$$

with V and V^* as in Eq. (III.67).

2.2.3 Nominal data-driven solver

Alternated minimization. As for any data-driven system we have seen so far, the nominal formulation yields a minimization problem coupled with a combinatorial optimization. For the sake of simplicity, this first attempt for a nominal finite strain data-driven solver also employs the alternated minimization represented in Fig. III.1.

As previously, the data-driven problem Eq. (III.77) is split into two distinct problems:

- Let us first consider the stage where the mechanical state

$$z = \{z_e = (\boldsymbol{F}_e, \boldsymbol{P}_e)\}_{e=1}^{M} \in \mathcal{E} \tag{III.79}$$

is fixed. The material solution set $\mathcal{S}^{\text{mat}}$ is then determined from Eq. (III.77) as

$$\mathcal{S}^{\text{mat}} = \arg\min_{z'\in\mathcal{D}} \sum_{e=1}^{M} w_e \, d_e(z_e, z'_e). \tag{III.80}$$

or equivalently as (all terms in the summation being independent from one another)

$$\mathcal{S}^{\text{mat}}(z) = \left\{ \arg\min_{z_i\in\mathcal{D}_{\text{loc}}^{(e)}} d_e\left(z_e, z_i\right) \;\middle|\; \forall e \in [1\,..\,M] \right\}. \tag{III.81}$$

As in Eq. (III.32), we refer to $z_{\mathsf{i}e}^* = (\boldsymbol{F}_{\mathsf{i}e}^*, \boldsymbol{P}_{\mathsf{i}e}^*)$ as the *material state* associated to e, defined as the material data point closest to the current local mechanical state $z_e = (\boldsymbol{F}_e, \boldsymbol{P}_e)$:

$$d_e\left(z_e, z_{\mathsf{i}e}^*\right) \leq d_e\left(z_e, z_i\right), \forall e \in [1\,..\,M], \forall z_i \in \mathcal{D}_{\text{loc}}^{(e)}. \tag{III.82}$$

Again, the integer index $\mathsf{i}e$ is then the actual optimization variable.

- Let us now consider the stage where the material state

$$z^* = \{z^*_{\mathrm{ie}} = (\boldsymbol{F}^*_{\mathrm{ie}}, \boldsymbol{P}^*_{\mathrm{ie}})\}_{e=1}^{M} \in \mathcal{D} \tag{III.83}$$

is determined. Eq. (III.77) is then reduced to the following *constrained* minimization problem

$$\mathcal{S}^{\mathrm{mec}}(z^*) = \arg\min_{z \in \mathcal{Z}} \sum_{e=1}^{M} w_e \; d_e(z_e, z^*_{\mathrm{ie}}) \tag{III.84a}$$

subject to

$$\boldsymbol{F}_e = \boldsymbol{\nabla}_0^e \boldsymbol{u} + \boldsymbol{I}, \qquad \forall e \in [1 \,..\, M], \tag{III.84b}$$

$$\boldsymbol{u}_a \cdot \boldsymbol{e}_i - u^{\mathrm{D}}_\alpha = 0, \qquad \forall (a,i) : \alpha(a,i) \in D, \tag{III.84c}$$

$$\sum_{e=1}^{M} w_e \; P^e_{iJ} \; B^{ea}_J - \boldsymbol{T}^{\mathrm{ext}}_a \cdot \boldsymbol{e}_i = 0, \qquad \forall (a,i) : \alpha(a,i) \notin D, \tag{III.84d}$$

$$\boldsymbol{r}_e = \boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}_e \boldsymbol{P}_e^T\right) = \boldsymbol{0}, \qquad \forall e \in [1 \,..\, M]. \tag{III.84e}$$

The important points regarding the different stages of this algorithm we raised in Section 2.1.4 are still valid:

- the core of the algorithm is stage 2, the computation of the mechanical states from the resolution of Eq. (III.84);
- the question of the initialization of the material states must be specifically addressed and will therefore be investigated in Section 3;
- the third stage of the algorithm in Fig. III.1 boils down to the search for nearest neighbor in the local phase space defined by Eq. (III.81).

However, the nominal formulation differs from the Lagrangian for two main reasons:

1. the conservation of angular momentum constitutes an additional constraint to the minimization problem, at integration points.
2. the distance function Eq. (III.71) measuring the deviation to the material database is no longer quadratic, for general values of p and q;

In that respect, the construction and implementation of a nominal data-driven solver is more complex than its Lagrangian counterpart. Indeed, both features prevented us from using the same method of Lagrange multipliers: (i) one would need a new Lagrange multiplier attached to the integration points to enforce rotational equilibrium Eq. (III.74), and (ii) the stationary equation obtained from the partial derivatives with respect to the stress would yield a nonlinear equation in $\boldsymbol{P}$, which we could not render into a stress correction of the form Eqs. (III.40) and (III.41c). Together with the issue of adding many unknowns to the numerical system, this difficulty stopped

us from going further with the method of Lagrange multipliers. As the addition of unknowns is not favorable, we turned to penalty methods. We elaborated an augmented Lagrangian approach, which offers more robustness than the quadratic penalty function method (Bertsekas, 1996). The details of the implementation are given next.

The augmented Lagrangian method. The augmented Lagrangian method, originally known as the method of multipliers, was introduced as an alternative method to the quadratic penalty function method, which avoids the problem of ill-conditioning, and provides a faster rate of convergence. To describe the approach, we adopt the formalism and numerical advice given in the reference book of Bertsekas (1996). Consider the following equality constrained problem, involving the objective function $f : \mathbb{R}^n \to \mathbb{R}$, and the constraints $h : \mathbb{R}^n \to \mathbb{R}^m$:

$$\text{minimize} \quad f(x) \qquad \text{subject to} \quad h(x) = 0. \tag{III.85}$$

The component of h are $h_1, \ldots, h_m$. For any scalar c consider also the augmented Lagrangian function

$$L_c(x, \lambda) = f(x) + \frac{1}{2} c \, |h(x)|^2 + \lambda^T h(x). \tag{III.86}$$

The method of multipliers is an iterative procedure, which aims at solving a series of unconstrained minimization problems $L_{c_j}(x, \lambda_j)$. After each iteration, the penalty parameter c_j together with the Lagrange multiplier λ_j are updated according to the rules:

$$\lambda_{j+1} = \lambda_j + c_j h(x_j) \text{ and } c_{j+1} \geq c_j. \tag{III.87}$$

Bertsekas (1996, Section 2.2) recommend the following scheme for updating the penalty parameter c: choose a moderate c_0 and monotonically increase c_j via the equation $c_{j+1} = \beta c_j$, where β is a scalar with $\beta > 1$. "Typical choices are $\beta \in [4, 10]$". In addition, "it is possible to use a different penalty parameter for each constraint $h_i(x) = 0$", which can be "beneficial in a situation where the constraints are 'poorly scaled'". In the next paragraph, we reformulate the constrained minimization problem Eq. (III.84) to fit in the above augmented Lagrangian method.

Application of the augmented Lagrangian method. As before, the compatibility constraint Eq. (III.84b) is directly enforced by replacing the deformation gradient with its relationship to the displacement field. The unknowns in Eq. (III.84) then become the displacement field $\boldsymbol{u}$ and the stress field $\boldsymbol{P}$. Thanks to the FE element discretization, the actual unknowns are the nodal displacements $\{\boldsymbol{u}_a\}_{a=1}^N$ and the values of the stress tensor at the integration points $\{\boldsymbol{P}_e\}_{e=1}^M$. These can be gathered in the unknown vector x, defined as the concatenation of all flattened displacements arrays and all flattened stress tensors: These are recast into arrays[4]

[4] From here on out, "numerical" arrays are denoted with light-face symbols.

u and P, of shape $Nn \times 1$ and $Mn^2 \times 1$ respectively, as

$$u = [u_{1,1}, u_{1,2}, u_{1,3}, \dots, u_{a,1}, u_{a,2}, u_{a,3}, \dots, u_{N,1}, u_{N,2}, u_{N,n}]^T \tag{III.88a}$$
$$P = \left[P^1_{11}, P^1_{12}, P^1_{13}, P^1_{21}, P^1_{22}, \dots P^1_{33}, \dots, \right. \tag{III.88b}$$
$$P^e_{11}, P^e_{12}, P^e_{13}, P^e_{21}, P^e_{22}, \dots P^e_{33}, \dots, \tag{III.88c}$$
$$\left. P^M_{11}, P^M_{12}, P^M_{13}, P^M_{21}, P^M_{22}, \dots P^M_{33}\right]^T, \tag{III.88d}$$

where $u_{a,i}$ is the displacement of node a in direction i and P^e_{ij} is the (i,j)-th component of the stress tensor $\boldsymbol{P}_e$. The unknown array x is the concatenation of u and P

$$x = \begin{pmatrix} u \\ P \end{pmatrix}. \tag{III.89}$$

$$\begin{aligned} x = [&u_{1,1}, u_{1,2}, u_{1,3}, \dots, u_{a,1}, u_{a,2}, u_{a,3}, \dots, u_{N,3}, u_{N,2}, u_{N,n}, \\ &P^1_{11}, P^1_{12}, P^1_{13}, P^1_{21}, P^1_{22}, \dots P^1_{33}, \dots, \\ &P^e_{11}, P^e_{12}, P^e_{13}, P^e_{21}, P^e_{22}, \dots P^e_{33}, \dots, \\ &P^M_{11}, P^M_{12}, P^M_{13}, P^M_{21}, P^M_{22}, \dots P^M_{33}]^T, \end{aligned} \tag{III.90}$$

For a two-dimensional problem ($n = 2$), the array x is obviously reduced accordingly. In general, the total number of independent variables to determine in the minimization is then $(Nn) \times (Mn^2)$. The objective function of the minimization problem is denoted $f(u, P)$ and writes

$$f(u, P) = \sum_{e=1}^{M} w_e \, d_e \left(V(\boldsymbol{F}_e(u) - \boldsymbol{F}^*_{\mathsf{ie}}) + V(\boldsymbol{P}_e - \boldsymbol{P}^*_{\mathsf{ie}}) \right). \tag{III.91}$$

The remaining minimization constraints to enforce are (i) the Dirichlet conditions Eq. (III.84c), (ii) the translational equilibrium Eq. (III.84d) and (iii) the conservation of angular momentum Eq. (III.84e). In like manner, we store these constraints in dedicated arrays, defined respectively as

$$h^{\mathrm{D}} = Su - u^{\mathrm{D}} \tag{III.92a}$$
$$h^{\mathrm{teq}} = \bar{B}^T W P - \bar{T}^{\mathrm{ext}} \tag{III.92b}$$
$$h^{\mathrm{req}} = [r_{1,1}, \dots, r_{e,i}, \dots, r_{M,n_r}]^T, \tag{III.92c}$$

where

- $u^{\mathrm{D}} = \left\{u^{\mathrm{D}}_\alpha\right\}_{\alpha \in D}$ is the array of prescribed displacements encoding the Dirichlet boundary conditions of size n_{D},
- S is a $n_{\mathrm{D}} \times (Nn)$ selection matrix of 0 and 1 ($S_{k\alpha} = 1$ if $\alpha = \alpha_k \in D$ and 0 otherwise),

- the matrix B collects the vectors $\boldsymbol{B}_{ea}$ and encodes the geometry and connectivity of the mesh,
- W is the diagonal matrix of weights w_e such that $WP = \sum_{e=1}^{M} w_e P_{ij}^e$,
- T^{ext} is the array of nodal external applied forces, flattened in like manner as u.

The bar symbol $\bar{\bullet}$ over B and T^{ext} means that they have been reduced to the degrees of freedom only. Finally, n_r is the dimension of the vector $\boldsymbol{r}_e$ measuring the conservation of angular momentum (see (III.84e)). In short, $n_r = 1$ for a two-dimensional problem and $n_r = 3$ for a three-dimensional problem. Then, h^{D} is an array of shape $(n_{\text{D}} \times 1)$, h^{teq} is an array of shape $(n_{\text{dof}} \times 1)$ and h^{req} is an array of shape $((Mn_r) \times 1)$, with n_{dof} the total number of degrees of freedom (note that $n_{\text{dof}} + n_{\text{D}} = Nn$). With each type of constraints, we associate a dedicated penalty parameter and a Lagrange multiplier vector of appropriate shape, as summarized in Table III.1.

Table III.1 – Penalty parameters and Lagrange multipliers associated with the optimization constraints of the nominal formulation.

Constraint	**Penalty parameter**	**Lagrange multiplier**
Dirichlet conditions $h^{\text{D}} = Su - u_D$	c^{D}	λ, shape $(n_D \times 1)$
Translational equilibrium $h^{\text{teq}} = \bar{B}^T WP - \bar{T}^{\text{ext}}$	c^{teq}	η, shape $(n_{\text{dof}} \times 1)$
Angular momentum $h^{\text{req}} = [r_{1,1}, \ldots, r_{M,n_r}]^T$	c^{req}	μ, shape $(Mn_c \times 1)$

The constrained minimization problem Eq. (III.84) is then conveniently rewritten as

$$\text{minimize} \quad f(u,P) \quad \text{subject to} \quad h = \begin{pmatrix} h^{\text{D}} \\ h^{\text{teq}} \\ h^{\text{req}} \end{pmatrix} = 0. \tag{III.93}$$

The corresponding augmented Lagrangian function is

$$\begin{aligned} L_{(c^{\text{D}},c^{\text{teq}},c^{\text{req}})}(x,(\lambda,\eta,\mu)) = f(u,P) + \frac{1}{2}c^{\text{D}}\,|h^{\text{D}}(x)|^2 + \frac{1}{2}c^{\text{teq}}\,|h^{\text{teq}}(x)|^2 \\ + \frac{1}{2}c^{\text{req}}\,|h^{\text{req}}(x)|^2 - \lambda^T h^{\text{D}}(x) - \eta^T h^{\text{teq}}(x) - \mu^T h^{\text{req}}(x). \end{aligned} \tag{III.94}$$

A typical iteration of the approach then writes[5]

$$\lambda_{j+1} = \lambda_j - c_j^{\mathrm{D}} h^{\mathrm{D}}(x_j), \qquad c_{j+1}^{\mathrm{D}} = \beta c_j^{\mathrm{D}} \tag{III.95a}$$

$$\eta_{j+1} = \eta_j - c_j^{\mathrm{teq}} h^{\mathrm{teq}}(x_j), \qquad c_{j+1}^{\mathrm{teq}} = \beta c_j^{\mathrm{teq}} \tag{III.95b}$$

$$\mu_{j+1} = \mu_j - c_j^{\mathrm{req}} h^{\mathrm{req}}(x_j), \qquad c_{j+1}^{\mathrm{req}} = \beta c_j^{\mathrm{req}} \tag{III.95c}$$

with x_j the solution of the unconstrained minimization of augmented Lagrangian function $L_{(c_j^{\mathrm{D}}, c_j^{\mathrm{teq}}, c_j^{\mathrm{req}})}(x, (\lambda_j, \eta_j, \mu_j))$. The iterations Eq. (III.95) stop when the Karush-Kuhn-Tucker (KKT) conditions are satisfied within a given tolerance tol:

$$\begin{pmatrix} \nabla_u f \\ \nabla_P f \end{pmatrix}(x) - (\nabla h(x))^T \begin{pmatrix} \lambda \\ \eta \\ \mu \end{pmatrix} < \mathrm{tol}. \tag{III.96}$$

Now that the data-driven is suitably written for the augmented Lagrangian method, we can proceed to give the resolution method for the unconstrained minimization of the augmented Lagrangian function Eq. (III.94).

Minimization of the augmented Lagrangian function. With the experience we had in implementing the Newton-Raphson procedure used in the Lagrangian formulation, we chose to minimize the augmented Lagrangian function with a Newton-Raphson scheme as well. Indeed, the gradients and hessians of the objective function and the constraints are available without too much pain and can be implemented rather easily as well. In this paragraph, we drop the j indices to reduce to amount of notation. We also simply denote L the augmented Lagrangian function. A typical Newton-Raphson iteration then writes:

$$x^{(j+1)} = x^{(j)} + \delta x^{(j)} \tag{III.97a}$$

$$\text{with } \begin{pmatrix} \nabla_u L \\ \nabla_u P \end{pmatrix} + \begin{pmatrix} \nabla_u^2 L & \nabla_P \nabla_u L \\ \nabla_u \nabla_P L & \nabla_P^2 L \end{pmatrix} \qquad \left(x^{(j)}\right) \delta x^{(j)} = 0 \tag{III.97b}$$

where the expressions of the gradient vectors $\nabla_u L$ and $\nabla_P L$ and Hessian matrices $\nabla_u^2 L$, $\nabla_P \nabla_u L$, $\nabla_u \nabla_P L$, and $\nabla_P^2 L$ are given in the following.

Note that this approach requires to build and solve a $(n \times N + n_{\mathrm{states}}) \times (n \times N + n_{\mathrm{states}})$ linear system of algebraic equations at every iteration of the Newton-Raphson scheme, every step of the augmented Lagrangian process, every iteration of the data-driven alternated minimization(, every load step of the incremental loading loop), where $n_{\mathrm{states}} = n^2 \times M$ is the number of stress unknowns (n^2 independent components in each stress tensor of all M integration points).

[5] To be consistent with the physical interpretation of the Lagrange multiplier $\boldsymbol{\eta}$ derived in the small strain or Lagrangian formulations, we chose to use "negative" Lagrange multipliers. The updating rule is modified in consequence, without loss of consistency with the original method of Bertsekas (1996). The nodal vectors $\boldsymbol{\eta}_a$ are then again interpreted as virtual displacements which, associated to a stiffness C, generate stresses that correct the out-of-balance gap between the material stresses and the external nodal forces.

First, we evaluate the gradients $\nabla_u L$ and $\nabla_P L$ of the augmented Lagrangian function:

$$\nabla_u L = \nabla_u f + c^{\mathrm{D}} (\nabla_u h^{\mathrm{D}})^T \, h^{\mathrm{D}} + c^{\mathrm{req}} (\nabla_u h^{\mathrm{req}})^T \, h^{\mathrm{req}} - (\nabla_u h^{\mathrm{D}})^T \, \lambda - (\nabla_u h^{\mathrm{req}})^T \, \mu, \quad \text{(III.98a)}$$

and

$$\nabla_P L = \nabla_P f + c^{\mathrm{teq}} (\nabla_P h^{\mathrm{teq}})^T \, h^{\mathrm{teq}} + c^{\mathrm{req}} (\nabla_P h^{\mathrm{req}})^T \, h^{\mathrm{req}} - (\nabla_P h^{\mathrm{teq}})^T \, \eta - (\nabla_P h^{\mathrm{req}})^T \, \mu, \quad \text{(III.98b)}$$

where the different terms are defined in what follows.

- The gradients of the objective function f respectively write

$$[\nabla_u f]_\alpha = \frac{\partial f}{\partial u_{a,i}} = \sum_{e=1}^{M} w_e \, C^{p/2} \, (I\!I_{\Delta \boldsymbol{F}_e})^{p/2-1} \; \Delta F^e_{iJ} \, B^{ea}_J, \qquad \forall \alpha(a,i) \quad \text{(III.99a)}$$

$$[\nabla_P f]_\iota = \frac{\partial f}{\partial P^e_{iJ}} = w_e \, C^{-q/2} \; (I\!I_{\Delta \boldsymbol{P}_e})^{q/2-1} \, \Delta P^e_{iJ}, \qquad \forall \iota(e,i,J) \quad \text{(III.99b)}$$

  where $I\!I_{\boldsymbol{A}} = \mathrm{tr}\left(\boldsymbol{A}^T \boldsymbol{A}\right)$ denotes the second invariant of a tensor, $\Delta \boldsymbol{F}_e = \boldsymbol{F}_e - \boldsymbol{F}^*_{\mathsf{ie}}$ and $\Delta \boldsymbol{P}_e = \boldsymbol{P}_e - \boldsymbol{P}^*_{\mathsf{ie}}$. The integer α is the global numeration of the displacement array u and ι is the global numeration of the stress array P, related to the local numerations as

$$\alpha(a,i) = (a-1)n + i, \quad \text{(III.100a)}$$

$$\iota(e,i,J) = (e-1)n^2 + (i-1)n + J, \quad \text{(III.100b)}$$

  with a the node, i, J the directions, e the integration point and n the dimension of the problem.

- In addition, the gradients of the constraints respectively write

$$\nabla_u h^{\mathrm{D}} = S \quad \text{(III.101a)}$$

$$\nabla_P h^{\mathrm{teq}} = \bar{B}^T W \quad \text{(III.101b)}$$

$$[\nabla_u h^{\mathrm{req}}]_{c,\alpha} = \frac{\partial r_{e,i}}{\partial u_{a,j}} = [\boldsymbol{\mathcal{E}}]_{ijk} \, P^e_{kL} \, B^{ea}_L = [\boldsymbol{\mathcal{E}} \cdot (\boldsymbol{P}_e \boldsymbol{B}_{ea})]_{ij}\,, \qquad \forall (c,\alpha) \quad \text{(III.101c)}$$

$$[\nabla_P h^{\mathrm{req}}]_{c,\iota} = \frac{\partial r_{e,i}}{\partial P^e_{kL}} = [\boldsymbol{\mathcal{E}}]_{ijk} \, F^e_{jL} = [-\boldsymbol{\mathcal{E}} \cdot \boldsymbol{F}_e]_{ikL}\,, \qquad \forall (c,\iota) \quad \text{(III.101d)}$$

  where c is the global numeration of the conservation of angular momentum constraint h^{req}, related to the number of integration point and the direction as

$$c = (e-1)n^2 + i, \; \forall i \in [1 \,..\, n_r]. \quad \text{(III.102)}$$

We proceed to give the expressions of the terms in the tangent matrix of the Newton-Raphson iteration.

- The diagonal blocks respectively write

$$\nabla_u^2 L = \nabla_u^2 f + c^{\mathrm{D}} (\nabla_u h^{\mathrm{D}})^T \nabla_u h^{\mathrm{D}} + c^{\mathrm{req}} (\nabla_u h^{\mathrm{req}})^T \nabla_u h^{\mathrm{req}} \tag{III.103a}$$
$$\nabla_P^2 L = \nabla_P^2 f + c^{\mathrm{teq}} (\nabla_P h^{\mathrm{teq}})^T \nabla_P h^{\mathrm{teq}} + c^{\mathrm{req}} (\nabla_P h^{\mathrm{req}})^T \nabla_P h^{\mathrm{req}} \tag{III.103b}$$

  where the hessian blocks of the objective function are defined as

$$\begin{aligned}[\nabla_u^2 f]_{\alpha_1,\alpha_2} &= \frac{\partial^2 f}{\partial u_{b,j}\, \partial u_{a,i}} \\ &= \sum_{e=1}^{M} w_e\, C^{p/2} (p-2)\, (I\!I_{\Delta \boldsymbol{F}_e})^{p/2-2} \left[\Delta \boldsymbol{F}_e \boldsymbol{B}_{ea}\right]_i \left[\Delta \boldsymbol{F}_e \boldsymbol{B}_{eb}\right]_j \\ &\qquad + \sum_{e=1}^{M} w_e\, C^{p/2}\, (I\!I_{\Delta \boldsymbol{F}_e})^{p/2-1} \left(\boldsymbol{B}_{ea} \cdot \boldsymbol{B}_{eb}\right) \delta_{ij}, \end{aligned} \tag{III.104a}$$

  for all a, i, b, j such that $\alpha_1 = \alpha(a,i)$ and $\alpha_2 = \alpha(b,j)$, and

$$\begin{aligned}[\nabla_P^2 f]_{\iota_1,\iota_2} = \frac{\partial^2 f}{\partial P^e_{iJ}\, \partial P^e_{kL}} &= w_e C^{-q/2} (q-2)\, (I\!I_{\Delta \boldsymbol{P}_e})^{q/2-2} \left(\Delta P^e_{iJ} P^e_{kL}\right) \\ &\qquad + w_e C^{-q/2}\, (I\!I_{\Delta \boldsymbol{P}_e})^{q/2-1}\, \delta_{ik} \delta_{JL}, \end{aligned} \tag{III.104b}$$

  for all e, i, J, k, L such that $\iota_1 = \iota(e,i,J)$ and $\iota_2 = \iota(e,k,L)$, with δ_{ij} the Kronecker symbol ($\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$).

- The anti-diagonal blocks of the tangent matrix only involve the gradient and Hessian of the bi-linear constraint h^{req}, as it is the only one coupling u and P. The two blocks are symmetrical to one another such that

$$\begin{aligned}\nabla_P \nabla_u L = c^{\mathrm{req}} \Big((\nabla_u h^{\mathrm{req}})^T \nabla_P h^{\mathrm{req}} + \nabla_P \nabla_u \left(h^{\mathrm{req}} \cdot h^{\mathrm{req}}\right) \Big) \\ - \nabla_P \nabla_u \left(\mu \cdot h^{\mathrm{req}}\right), \end{aligned} \tag{III.105a}$$

  and

$$\nabla_u \nabla_P L = \left(\nabla_P \nabla_u L\right)^T, \tag{III.105b}$$

  where the Hessian term of the constraint h^{req} is defined, for any array y of the same size, as

$$\left[\nabla_P \nabla_u \left(y \cdot h^{\mathrm{req}}\right)\right]_{\alpha,\iota} = \sum_{i=1}^{n_c} y_i \left[\boldsymbol{\mathcal{E}}\right]_{ijk} \boldsymbol{B}^{ea}_L, \qquad \forall \alpha(a,j),\, \iota(e,k,L) \tag{III.106}$$

Values of the parameters (p, q)**.** As we can see, the gradient vector (respectively the Hessian matrix) of the objective function involves scalar terms raised to the $(p/2-1)$-th or $(q/2-1)$-th power (respectively $(p/2-2)$-th and $(q/2-2)$-th power). These cause numerical problems as soon as the operand is close to zero for certain value of (p, q): the terms can become infinite. This situation is especially expected at convergence: when the mechanical states are increasingly close to the material states, $\Delta \boldsymbol{F}_e \to \boldsymbol{0}$ and $\Delta \boldsymbol{P}_e \to \boldsymbol{0}$ and hence, $I\!I_{\Delta \boldsymbol{F}_e} \to 0$ and $I\!I_{\Delta \boldsymbol{P}_e} \to 0$. To avoid this kind of problem[6], we restrict from here on out to the only possible (p, q) couple for which the gradient and hessian are defined everywhere, *i.e.* such that $p/2-1 \geq 0$ and $q/2-1 \geq 0$ with $1/p+1/q = 1$, *i.e.* $p = q = 2$. Consequently, the gradient and Hessian of the objective function are respectively simplified into

$$[\nabla_u f]_\alpha = \frac{\partial f}{\partial u_{a,i}} = \sum_{e=1}^{M} w_e \, C \, \Delta F^e_{iJ} \, B^{ea}_J, \tag{III.107a}$$
$$\forall a, i, \alpha(a, i)$$
$$[\nabla_P f]_\iota = \frac{\partial f}{\partial P^e_{iJ}} = w_e \, C^{-1} \, \Delta P^e_{iJ}, \tag{III.107b}$$
$$\forall e, i, J, \iota(e, i, J)$$

and

$$[\nabla^2_u f]_{\alpha_1,\alpha_2} = \frac{\partial^2 f}{\partial u_{b,j} \, \partial u_{a,i}} = \sum_{e=1}^{M} w_e \, C \, (\boldsymbol{B}_{ea} \cdot \boldsymbol{B}_{\boldsymbol{eb}}) \, \delta_{ij}, \tag{III.108a}$$
$$\forall a, i, b, j \text{ such that } \alpha_1 = \alpha(a, i) \text{ and } \alpha_2 = \alpha(b, j)$$
$$[\nabla^2_P f]_{\iota_1,\iota_2} = \frac{\partial^2 f}{\partial P^e_{iJ} \, \partial P^e_{kL}} = w_e \, C^{-1} \, \delta_{ik} \delta_{JL}, \tag{III.108b}$$
$$\forall e, i, J, k, L \text{ such that } \iota_1 = \iota(e, i, J) \text{ and } \iota_2 = \iota(e, k, L)$$

Let us also mention another implementation constraint that advocates for the choice of $p = q = 2$. The tree-based nearest neighbor search algorithms available in both Matlab and Python do not offer to specify a custom distance function. Instead, the samples and query points must be scaled so that their distance can be measured with the standard Euclidean distance. We could not find a way to easily formulate such a transformation of the phase space for the general (p, q) distance. The so-called Euclidean mapping for the nominal state pairs is given in Appendix B, along with its aforementioned Lagrangian counterpart.

Summary. We are now ready to present the augmented Lagrangian algorithm we implemented. The algorithm in listed in Algorithm III.2. It fits in stage 2 of the alternated minimization scheme represented in Fig. III.1 and listed in Algorithm III.3.

Along with its Lagrangian counterpart, the finite strain data-driven solver in the nominal formulation presents two main features:

[6] The general case of $(p, q) \in (1, \infty)^2$ could be addressed close to zero with a specific numerical scheme. However, this possibility has not been explored in this thesis.

- the computation of the mechanical states in two steps, namely
  - the resolution constrained minimization problem Eq. (III.93) with an augmented Lagrangian method inspired from Bertsekas (1996), which uses a Newton-Raphson scheme for the minimization of the augmented Lagrangian function,
  - the computation of the local deformation gradients only via the compatibility Eq. (III.72a), the stress being up-to-date as unknowns of the above minimization;
- the nearest-neighbor search for the optimal states in the material database from Eq. (III.82), using an unsupervised tree-based search algorithm.

2.2.4 Linear nominal data-driven solver

A simpler version of the solver consists in enforcing the compatibility and translational equilibrium only, *i.e.* minimizing the mechanical states in $\mathcal{E}_0$ instead of $\mathcal{E}$ (see Eq. (III.60)). In that case, the augmented Lagrangian function is reduced to

$$L_{(c^{\mathrm{D}},c^{\mathrm{teq}})}(x,(\lambda,\eta)) = f(u,P) + \frac{1}{2}c^{\mathrm{D}}\,|h^{\mathrm{D}}(x)|^2 + \frac{1}{2}c^{\mathrm{teq}}\,|h^{\mathrm{teq}}(x)|^2 - \lambda^T h^{\mathrm{D}}(x) - \eta^T h^{\mathrm{teq}}(x). \quad \text{(III.109)}$$

The stationary equations yield two independent linear systems

$$\nabla_u L = \nabla_u f + c^{\mathrm{D}}(\nabla_u h^{\mathrm{D}})^T\, h^{\mathrm{D}} - (\nabla_u h^{\mathrm{D}})^T\,\lambda = 0, \quad \text{(III.110a)}$$

$$\nabla_P L = \nabla_P f + c^{\mathrm{teq}}(\nabla_P h^{\mathrm{teq}})^T\, h^{\mathrm{teq}} - (\nabla_P h^{\mathrm{teq}})^T\,\eta = 0. \quad \text{(III.110b)}$$

The pseudo-stiffness matrix of the two systems are respectively of size $Nn \times Nn$ and $n_{\mathrm{states}} \times n_{\mathrm{states}}$. This is computationally more advantageous than the complete nominal solver derived above. However, we need to assess whether the conservation of angular momentum is sufficiently enforced in the material data set $\mathcal{D}$ rather than in the constraint set $\mathcal{E}$. Indeed, the constraint of rotational equilibrium $\boldsymbol{r}_e = \mathbf{0}$ is purely local. It could then be verified at each integration point in the mesh thanks to the associated material state instead of enforced in the minimization of the mechanical state.

Algorithm III.2 Augmented Lagrangian solver for the nominal formulation of the data-driven constrained minimization problem.

INPUT: Results from the $(k-1)$-th iteration of the data-driven solver.
OUTPUT: Displacement and stress fields satisfying mechanically admissibility.
Require: Initial penalty parameters c_0^{D}, c_0^{teq}, c_0^{req}, initial Lagrange multipliers λ_0, η_0, μ_0, tolerance parameter $r_{\mathrm{tol}} > 0$, maximum number of iterations $j_{\max}$.

$j \leftarrow 0$
$x_{j=0} \leftarrow u^{(k-1)}, P^{(k-1)}$ ▶ *start from the results of the previous data-driven iteration*
$c_{j=0}^{\mathrm{D}}, c_{j=0}^{\mathrm{teq}}, c_{j=0}^{\mathrm{req}} \leftarrow c_0^{\mathrm{D}}, c_0^{\mathrm{teq}}, c_0^{\mathrm{req}}$
$\lambda_{j=0}, \eta_{j=0}, \mu_{j=0} \leftarrow \lambda_0, \eta_0, \mu_0$
$\mathrm{KKT}_{j=0} \leftarrow \|\nabla f(x_0) - [\lambda_0^T, \eta_0^T, \mu_0^T]\nabla h(x_0)\|_2$ ▶ *initialize KKT conditions*
for $j = 1 \,..\, j_{\max}$ **do** ▶ *Augmented Lagrangian iterations*
    $x_j \leftarrow \text{minimize } L_{c_j^{\mathrm{D}}, c_j^{\mathrm{teq}}, c^j}(x_j, \lambda_j, \eta_j, \mu_j)$ ▶ *Newton-Raphson Eq. (III.97) with initial guess x_{j-1}*
    $\mathrm{KKT}_j \leftarrow \|\nabla f(x_j) - [\lambda_j^T, \eta_j^T, \mu_j^T]\nabla h(x_j)\|_2$ ▶ *update KKT conditions*
    if $\mathrm{KKT}_j \leq r_{\mathrm{tol}}\mathrm{KKT}_0$ **or** $\mathrm{KKT}_j > \mathrm{KKT}_{j-1}$ **then**
        return x_j
    else
        update Lagrange multipliers
        $\lambda_{j+1} \leftarrow \lambda_j - c_j^{\mathrm{D}} h^{\mathrm{D}}(x_j)$
        $\eta_{j+1} \leftarrow \eta_j - c_j^{\mathrm{teq}} h^{\mathrm{teq}}(x_j)$
        $\mu_{j+1} \leftarrow \mu_j - c_j^{\mathrm{req}} h^{\mathrm{req}}(x_j)$
        update penalty parameters (Bertsekas, 1996)
        if $\|h(x_j)\|_2 > \gamma\|h(x_{j-1})\|_2$ **then**
            $c_{j+1} \leftarrow \beta c_j$
        else
            $c_{j+1} \leftarrow c_j$
        end if
    end if
end for

Algorithm III.3 Data-driven solver - nominal formulation.

INPUT: A mesh of M integration points and N nodes, boundary and loading conditions, material data set $\mathcal{D}$.

OUTPUT: mechanical response (displacements $\{\boldsymbol{u}_a\}_{a=1}^{N}$, mechanical states $\{(\boldsymbol{F}_e, \boldsymbol{P}_e)\}_{e=1}^{M}$, residual forces, *etc.*), material states $\{(\boldsymbol{F}^*_{\mathsf{ie}}, \boldsymbol{P}^*_{\mathsf{ie}})\}_{e=1}^{M}$.

Require: maximum number of iterations $k_{\max}$.

1) Initialize all material states and mapping

$k \leftarrow 0$
for $e = 1 \,..\, M$ **do**
    $\mathsf{ie}^{(0)} \leftarrow i,\ i \in [1 \,..\, m_e]$
end for

$u^{(k=0)} \leftarrow 0$
$P^{(k=0)} \leftarrow 0$
for $k = 1 \,..\, k_{\max}$ **do**
    2) Compute mechanical states
    2)a. Minimize Eq. (III.93)
    $(u^{(k)}, P^{(k)}) \leftarrow x$ from Algorithm III.2
    2)b. Update mechanical states
    for $e = 1 \,..\, M$ **do**
        $\boldsymbol{F}_e^{(k)} \leftarrow \boldsymbol{\nabla}_0 \boldsymbol{u}^{(k)} + \boldsymbol{I}$
        $\boldsymbol{P}_e^{(k)} \leftarrow P^{(k)}$
    end for
    3) Update material states and mapping
    for $e = 1 \,..\, M$ **do**
        $\mathsf{ie}^{(k)} \leftarrow \left(\boldsymbol{F}^*_{\mathsf{ie}^{(k)}}, \boldsymbol{P}^*_{\mathsf{ie}^{(k)}}\right)$ from Eq. (III.82)
    end for
    4) Test convergence
    if $\mathsf{ie}^{(k)} = \mathsf{ie}^{(k-1)}$ for all $e = 1 \,..\, M$ **then**
        $\boldsymbol{u} \leftarrow \boldsymbol{u}^{(k)}$
        $(\boldsymbol{F}_e, \boldsymbol{P}_e) \leftarrow (\boldsymbol{F}_e^{(k)}, \boldsymbol{P}_e^{(k)})$ for all $e = 1 \,..\, M$
        exit.
    else
        $\mathsf{ie}^{(k+1)} \leftarrow \mathsf{ie}^{(k)}$ for all $e = 1 \,..\, M$
    end if
end for

2.3 Theoretical comparison of the two approaches

Before moving to the numerical validation of both finite strain data-driven solvers, we discuss some of their main differences.

First, from a theoretical point of view, we must emphasize that no mathematical analysis proving the existence and convergence of data-driven solutions has been conducted for the Lagrangian formulation, to the best of our knowledge. In particular, the quadratic nature of the compatibility constraint highly complicates the problem. Conversely, following the work done for geometrically linear elasticity in Conti et al. (2018), Conti et al. (2020) theoretically assessed the existence and the convergence of data-driven solutions for the nominal formulation. However, the theorems required several topological conditions to be met both on the material data set and the constraint set, which are difficult to implement numerically. We refer the interested reader to the publication for more details. In particular, Conti et al. (2020) proposed two hyperelastic models which generate material data sets satisfying these conditions, one in 2D and one in 3D. The respective strain density functions W_2 and W_3 write

$$W_2(\boldsymbol{C}) = \frac{1}{2} I_{\boldsymbol{C}} + \frac{1}{4} a I_{\boldsymbol{C}}^2 + g_2(j), \quad \forall \boldsymbol{C} \in \mathbb{R}^{2\times 2}_{\text{sym}} \tag{III.111a}$$

$$W_3(\boldsymbol{C}) = \frac{1}{2} I_{\boldsymbol{C}} + \frac{1}{4} a I_{\boldsymbol{C}}^2 + \frac{1}{6} e I_{\boldsymbol{C}}^3 + g_3(J), \quad \forall \boldsymbol{C} \in \mathbb{R}^{3\times 3}_{\text{sym}}, \tag{III.111b}$$

with a, b, e positive constants. The volumetric strain energy functions $g_2(j = \det \boldsymbol{F})$ for $\boldsymbol{F} \in \mathbb{R}^{2\times 2}$ and $g_3(J = \det \boldsymbol{F})$ for $\boldsymbol{F} \in \mathbb{R}^{3\times 3}$ are defined as

$$g_2(j) = \frac{1}{2}\beta \left(j - 1 - \frac{1 + 2a}{\beta} \right)^2, \tag{III.112a}$$

$$g_3(J) = \frac{1}{2}\beta \left(J - 1 - \frac{1 + 3a + 9e}{\beta} \right)^2. \tag{III.112b}$$

Conti et al. (2020) stated that when the material data set is generated with the strain energy density function W_2 (respectively W_3), the data-driven problem Eq. (III.64) with the deviation functions Eq. (III.66) has a unique solution, the classical solution, when $(p, q) = (4, 4/3)$ (respectively when $(p, q) = (6, 6/5)$). Unfortunately, we could not numerically assess these models for the following reasons:

- both material data sets require to solve the problem with $(p, q) \neq (2, 2)$, which is currently out of our reach;
- we bring out that the volumetric functions triggered non-physical response of the material in compression and uniaxial tension. Namely, a tensile specimen would grow thicker in the directions perpendicular to the loading, corresponding to a sort of negative Poisson effect. See Appendix D for details.

Second, the implementation of the nominal formulation is more complex than that of the Lagrangian formulation. In particular, its requires to use an unconstrained minimization scheme (here we used a Newton-Raphson method) inside an

augmented Lagrangian loop, which adds another layer to the algorithm. However, the nominal solver (provided that $(p, q) = (2, 2)$) could be rendered much simpler if the conservation of angular momentum need not be enforced, but merely satisfied by the material data set, as suggested in Section 2.2.4. This statement has yet to be verified numerically where the material data set is discrete and comprises a finite (and possibly limited) number of points.

Third, the curse of dimensionality is even more acute in the nominal formulation as the phase space consists of non-symmetric tensor pairs. The nominal phase space is 8 and 18-dimensional while the Lagrangian phase space is 6 and 12-dimensional for 2 and 3-dimensional problems respectively. This could have an important impact on the coverage and sampling of the phase space by the material data set and possibly on the ability for the solver to find a satisfactory solution. These aspects are further discussed in the two next chapters.

In summary, on the one hand, the Lagrangian approach provides an "engineering" solution for the finite strain data-driven problem. On the other hand, the nominal approach presents some mathematical grounds but amounts to a more complex numerical problem. With this in mind, we can now proceed to the next section, which aims at validating both solvers on "sanity check" test cases.

3 Numerical validation

This section discusses some of the practical aspects in using finite strain data-driven solvers. To this end, we build artificial material databases from a constitutive relation and compare the data-driven results to classical finite element analysis achieved from the same model. It is then expected that the data-driven solution recovers or at least converges to the classical solution, which we take as a reference solution. To reduce the dimensionality of the phase space, we only consider two-dimensional problems, assuming plane stress conditions and incompressibility of the material. In the following, all tensors are then two-dimensional.

To generate the standard FE response, we use the standard neo-Hookean model, reduced to its two-dimensional expression. As shown in Appendix C, under plane stress and incompressibility constraints, the in-plane second Piola-Kirchhoff stress tensor $\boldsymbol{S} \in \mathbb{R}^{2\times 2}_{\text{sym}}$ is indeed expressed as a function of the in-plane right Cauchy-Green stretch tensor $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F} \in \mathbb{R}^{2\times 2}_{\text{sym}}$ by

$$\boldsymbol{S} = \mu \left(\boldsymbol{I} - (\det \boldsymbol{C})^{-1} \boldsymbol{C}^{-1} \right), \tag{III.113}$$

where μ is the shear modulus. In this study, we take $\mu = 1.2\,\text{MPa}$ (arbitrary value).

We investigate three 2D test cases: (i) uniaxial tension of a thin membrane, (ii) uniaxial tension of a thin membrane with the bottom clamped, and (iii) more complex loading (simultaneous tension and shear) of a thin membrane with a hole.

3.1 Validation: uniaxial tension

The first test case is used to validate the data-driven solvers in the simplest loading conditions: a uniaxial tensile test of a thin hyperelastic membrane. It also enables the visualization of the phase space in two dimensions and an easy synthetic generation of the database. Indeed, the stress tensors only have one non-zero component and the deformation-gradient tensors are entirely characterized by two inter-dependent quantities (longitudinal and transversal stretch ratios).

3.1.1 Computational problem

We consider a rectangular thin membrane of dimensions $l \times h = 5 \times 10\,\mathrm{mm}^2$, under plane stress conditions. The problem is then reduced to two dimensions. The geometry and mesh of the sample are shown in Fig. III.3. The mesh comprises

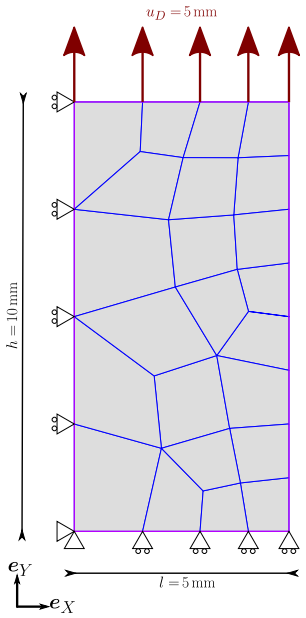


Figure III.3 – Geometry, boundary conditions and mesh of the sample in uniaxial tension.

$N = 36$ nodes and 23 bi-linear quadrangular elements with 4 integration points each, which amounts to $M = 92$ integration points. The boundary conditions are also represented in Fig. III.3 and write

$$\boldsymbol{u}_a \cdot \boldsymbol{e}_Y = 0, \qquad \forall a \in [1\,..\,N] : \boldsymbol{X}_a = (X_a, 0) \tag{III.114a}$$

$$\boldsymbol{u}_a \cdot \boldsymbol{e}_X = 0, \qquad \forall a \in [1\,..\,N] : \boldsymbol{X}_a = (0, Y_a) \tag{III.114b}$$

$$\boldsymbol{u}_a \cdot \boldsymbol{e}_Y = u_D, \qquad \forall a \in [1\,..\,N] : \boldsymbol{X}_a = (X_a, h) \tag{III.114c}$$

with $u_D = 5\,\mathrm{mm}$, which corresponds to a stretch $\lambda_Y^{\mathrm{ref}} = 1.5$.

The material database is generated by a sampling of the standard uniaxial stretch-stress curve of the neo-Hookean model Eq. (III.113). In uniaxial tension of direction Y, the different deformation and stress tensors are functions of the longitudinal and transversal stretch ratios λ_Y and λ_X, as summarized in Table III.2. Note that, due to the incompressibility of the material, we have $\lambda_X = \frac{1}{\sqrt{\lambda_Y}}$.

Table III.2 – Expressions of the deformation gradient, strain and stress tensors in plane stress uniaxial tension with an incompressible neoHookean model.

Description	Deformation tensors	Stress tensors
Lagrangian	$\boldsymbol{E} = 1/2(\lambda_X^2 - 1)\,\boldsymbol{e}_X \otimes \boldsymbol{e}_X$ $+1/2(\lambda_Y^2 - 1)\,\boldsymbol{e}_Y \otimes \boldsymbol{e}_Y$	$\boldsymbol{S} = \mu(1 - \lambda_Y^{-3})\,\boldsymbol{e}_Y \otimes \boldsymbol{e}_Y$
Nominal	$\boldsymbol{F} = \lambda_X \boldsymbol{e}_X \otimes \boldsymbol{e}_X + \lambda_Y \boldsymbol{e}_Y \otimes \boldsymbol{e}_Y$	$\boldsymbol{P} = \mu(\lambda_Y - \lambda_Y^{-2})\,\boldsymbol{e}_Y \otimes \boldsymbol{e}_Y$

The material database $\mathcal{D}_{\text{loc}}^m$ used in the Lagrangien (nominal) formulation results from a sampling of m points along the strain-stress (stretch-stress) curve given in Table III.2. In pratice, it results from a sampling of m longitudinal stretch ratios λ_Y, evenly spaced in the interval $[0.9, 2]$. For both formulations, we generate two types of databases:

1. A first database $\mathcal{D}_{\text{loc}}^{100}$, which contains the reference solution at $\lambda_Y^{\text{ref}} = 1.5$ together with 99 other material data points.

2. A second family of databases $\mathcal{D}_{\text{loc}}^m$, which do not contain the reference solution and consist of several samplings of increasing density:
$m \in \{11,\, 51,\, 101,\, 1001,\, 10\,001,\, 100\,001\}$.

Finally, the data-driven solutions were obtained with the following parameters:

C-parameter: the amplitude of the scaling parameter is set to the linearized Young modulus of the model $E^Y = 3\mu$, for both formulations: $C = E^Y$ in the nominal solver and, $\mathbb{C} = C\,\mathbb{I}^{\text{S}}$ in the Lagrangian solver, with $\mathbb{I}^{\text{SYM}}$ the symmetric fourth-order tensor.

Initialization: the material states are initialized to the zero strain-stress state, for both formulations (see Line 3 in Algorithm III.1 and Line 3 in Algorithm III.3): $(\boldsymbol{F}_{\text{ie}}^*, \boldsymbol{P}_{\text{ie}}^*) = (\boldsymbol{I}, \boldsymbol{0})$, $\forall e$ and $(\boldsymbol{E}_{\text{ie}}^*, \boldsymbol{S}_{\text{ie}}^*) = (\boldsymbol{0}, \boldsymbol{0})$, $\forall e$.

Augmented Lagrangian parameters: the initial penalty parameters $c_0^{\text{D}}, c_0^{\text{teq}}, c_0^{\text{req}}$ are respectively set to $10^3 C$, $10^3 C$, C. The initial Lagrange multipliers λ_0, η_0, μ_0 are all set to zero (see Lines 3 and 4 in Algorithm III.2).

3.1.2 Results and discussion

First, we demonstrate in Fig. III.4 that the reference solution is recovered by both data-driven solvers with the material database $\mathcal{D}_{\text{loc}}^{100}$, which contains the solution. Both solvers converged within very few iterations. In addition, the mechanical states (red circles) always remain on the constraint set throughout the iterations, while the material states are successively selected as the closest data points[7]. This is illustrated by the projection $P_{\mathcal{D}}$ and $P_{\mathcal{E}}$ on the material data set and on the constraint set (dotted and solid gray lines) respectively. On the one hand, the slope of the projection $P_{\mathcal{D}}$ is controlled by the C-parameter. On the other hand, the slope of the projection $P_{\mathcal{E}}$ reveals that the computation of the mechanical states for this displacement-controlled problem boils down to a correction of the stretch ratio λ_Y. Fig. III.4 also shows the deformation of the mesh throughout the iterations (purple lines). We can see that, in that case, the first iteration consists in enforcing the boundary conditions (semi-transparent purple lines at the furthest right delimits the deformed mesh at the end of the first iteration $k = 1$).

After this sanity check example, we can now proceed to show the results obtained from material data sets which do not contain the reference solution. In Fig. III.5 we plot the RMS percent errors in strain and stress against the size of the material database, re-defined as

$$E_{(\%\text{RMS})} = \left(\frac{\sum_{e=1}^{M} w_e W\left(\boldsymbol{E}_e - \boldsymbol{E}_e^{\text{ref}}\right)}{\sum_{e=1}^{M} w_e W\left(\boldsymbol{F}_e^{\text{ref}}\right)} \right)^{\frac{1}{2}} \tag{III.115a}$$

$$S_{(\%\text{RMS})} = \left(\frac{\sum_{e=1}^{M} w_e W^*\left(\boldsymbol{S}_e - \boldsymbol{S}_e^{\text{ref}}\right)}{\sum_{e=1}^{M} w_e W^*\left(\boldsymbol{S}_e^{\text{ref}}\right)} \right)^{\frac{1}{2}}, \tag{III.115b}$$

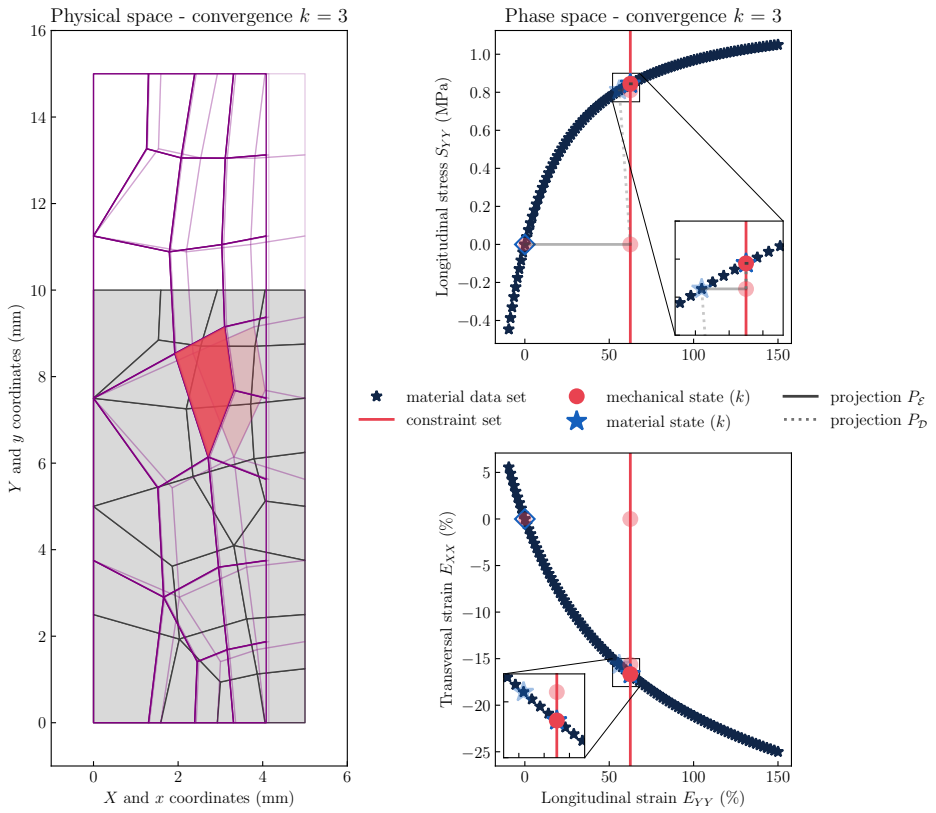
and

$$F_{(\%\text{RMS})} = \left(\frac{\sum_{e=1}^{M} w_e V\left(\boldsymbol{F}_e - \boldsymbol{F}_e^{\text{ref}}\right)}{\sum_{e=1}^{M} w_e V\left(\boldsymbol{F}_e^{\text{ref}}\right)} \right)^{\frac{1}{2}} \tag{III.115c}$$

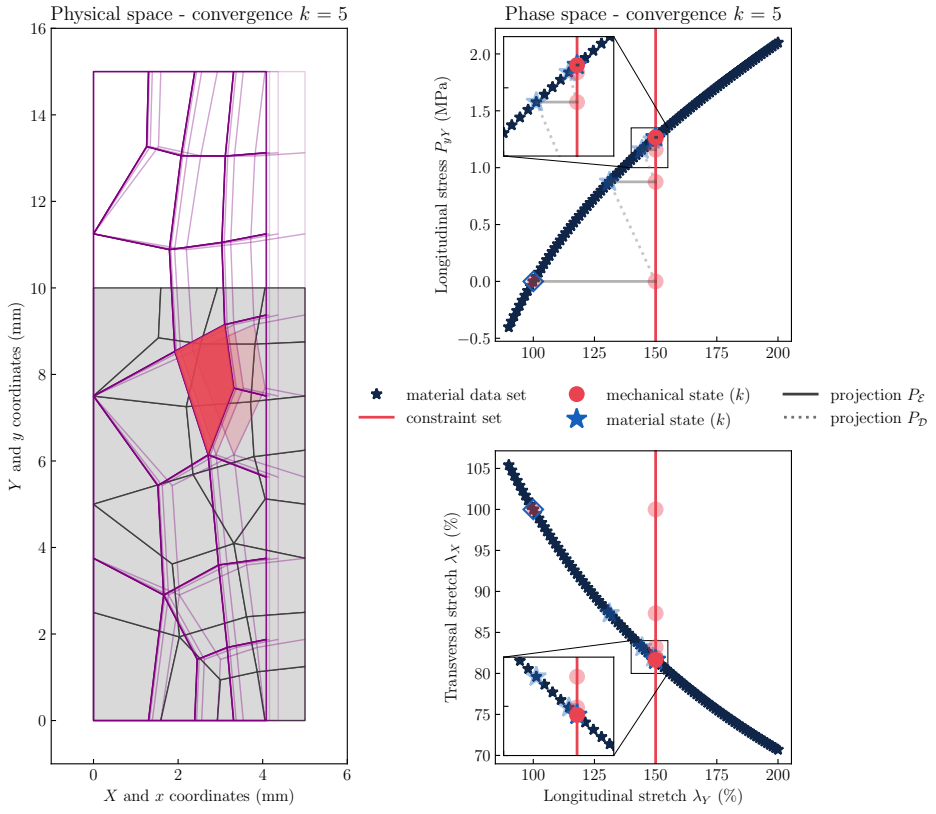
$$P_{(\%\text{RMS})} = \left(\frac{\sum_{e=1}^{M} w_e V^*\left(\boldsymbol{P}_e - \boldsymbol{P}_e^{\text{ref}}\right)}{\sum_{e=1}^{M} w_e V^*\left(\boldsymbol{P}_e^{\text{ref}}\right)} \right)^{\frac{1}{2}}, \tag{III.115d}$$

with W and W^* as in Eq. (III.9) and V and V^* are as in Eq. (III.67) using $(p, q) = (2, 2)$. As expected, all errors with respect to the solution converge to zero with increasing number of material data points. It is also notable that the two finite strain formulations lead to two different solutions. Yet, both material databases represent the exact same sampling of the constitutive model. The arrangement of the material data points in the different phase spaces could explain this result: the distance between two points in the Lagrangian phase space is different than the distance between the same points in the nominal phase space.

[7] Note that, here, the constraint set represented in Fig. III.4 is a projection of it in the 2D strain-stress plane. It is actually much more complex.



(a)



(b)

Figure III.4 – Physical space *vs.* phase space representation of the data-driven results obtained with the material database $\mathcal{D}^{100}_{\text{loc}}$. In phase space, we only plot the mechanical and material states of the corresponding red colored element in physical space. The results from previous iterations are semi-transparent. (a) Lagrangian formulation. (b) Nominal formulation.

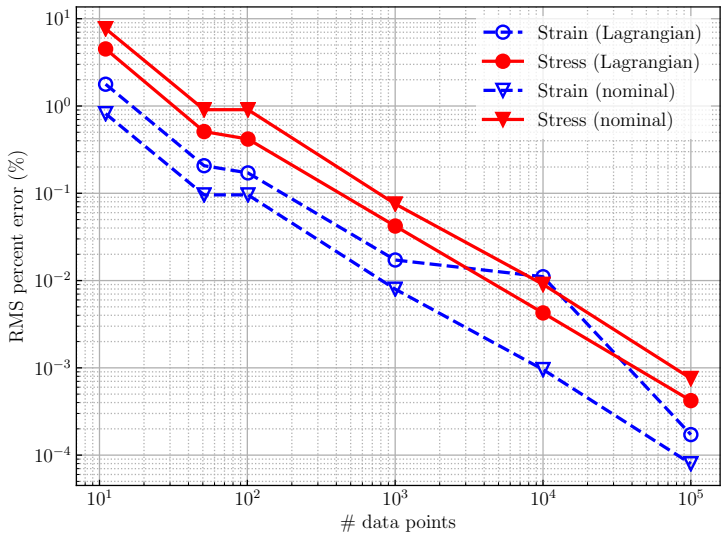


Figure III.5 – Convergence of the RMS percent errors in strain and stress with respect to the size of the database, for the uniaxial tensile test.

This very simple example has already shown interesting features of the finite strain data-driven solver:

- when the reference solution is in the data set, the data-driven solvers are able to recover it;
- when the intersection between the constraint set and the material data set is empty, the data-driven solution improves with the number of material data points, as the best material data point gets increasingly closer to the reference solution.

3.2 Non homogeneous case: clamped tension

Let us now consider a slightly more complex problem. We perform data-driven simulations of a thin membrane in tension, which is clamped at the bottom. The mechanical fields should then be homogeneous in most of the structure, except near the clamped edge.

3.2.1 Computational problem

The problem is again two dimensional, as we make the assumption of plane stress conditions. The geometry, mesh and loading of the sample are given in Fig. III.6. The mesh comprises $N = 340$ nodes and 378 bi-linear quadrangular elements with 4 integration points each, which amounts to $M = 1512$ integration points. The

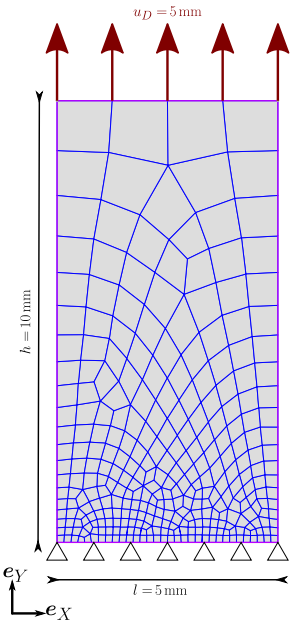


Figure III.6 – Geometry, boundary conditions and mesh of the sample in tension, clamped at the bottom.

boundary conditions are also represented in Fig. III.6 and write

$$\boldsymbol{u}_a = \boldsymbol{0}, \qquad \forall a \in [1\,..\,N] : \boldsymbol{X}_a = (X_a, 0); \tag{III.116}$$

$$\boldsymbol{u}_a \cdot \boldsymbol{e}_Y = u_D, \qquad \forall a \in [1\,..\,N] : \boldsymbol{X}_a = (X_a, h). \tag{III.117}$$

with $u_D = 5\,\mathrm{mm}$. In that case, the simulation is performed incrementally: the displacements are gradually prescribed in 10 regular steps.

The material database is only constituted with the classical FE solution of the problem, obtained with the incompressible neoHookean model Eq. (III.113). For all 10 load steps, we concatenate the reference deformation-stress state of all integration points into the Lagrangian and nominal databases respectively.

$$\mathcal{D}^{\mathrm{Lag}}_{\mathrm{loc}} = \left\{ \left\{ \left(\boldsymbol{E}_e^{\mathrm{ref},t}, \boldsymbol{S}_e^{\mathrm{ref},t} \right) \right\}_{e=1}^{M} \right\}_{t=1}^{10} \tag{III.118a}$$

$$\mathcal{D}^{\mathrm{nom}}_{\mathrm{loc}} = \left\{ \left\{ \left(\boldsymbol{F}_e^{\mathrm{ref},t}, \boldsymbol{P}_e^{\mathrm{ref},t} \right) \right\}_{e=1}^{M} \right\}_{t=1}^{10} \tag{III.118b}$$

The data-driven solutions were obtained with the following parameters:

C-parameter: the amplitude of the scaling parameter remains unchanged from the previous example.

Initialization: the material states are initialized with several techniques, identical in both formulations (see Line 3 in Algorithm III.1 and Line 3 in Algorithm III.3):

"zero" all material states are initialized to the zero-deformation state: $(\boldsymbol{F}^*_{\mathsf{ie}}, \boldsymbol{P}^*_{\mathsf{ie}}) = (\boldsymbol{I}, \boldsymbol{0})$, $\forall e$ and $(\boldsymbol{E}^*_{\mathsf{ie}}, \boldsymbol{S}^*_{\mathsf{ie}}) = (\boldsymbol{0}, \boldsymbol{0})$, $\forall e$.

"random" all material states are initialized to the same random data point: $(\boldsymbol{F}^*_{\mathsf{ie}}, \boldsymbol{P}^*_{\mathsf{ie}}) = (\boldsymbol{F}'_{i_r}, \boldsymbol{P}'_{i_r})$, $\forall e$ and $(\boldsymbol{E}^*_{\mathsf{ie}}, \boldsymbol{S}^*_{\mathsf{ie}}) = (\boldsymbol{E}'_{i_r}, \boldsymbol{S}'_{i_r})$, $\forall e$, with i_r a random integer in $[1 \,..\, m]$.

"random point-wise" each material state is initialized to a different random data point: $(\boldsymbol{F}^*_{\mathsf{ie}}, \boldsymbol{P}^*_{\mathsf{ie}}) = (\boldsymbol{F}'_{e_r}, \boldsymbol{P}'_{e_r})$, $\forall e$ and $(\boldsymbol{E}^*_{\mathsf{ie}}, \boldsymbol{S}^*_{\mathsf{ie}}) = (\boldsymbol{E}'_{e_r}, \boldsymbol{S}'_{e_r})$, $\forall e$, with e_r a different random integer in $[1 \,..\, m]$ for each e.

"reference" the material states are initialized to the classical FE solution: $(\boldsymbol{F}^*_{\mathsf{ie}}, \boldsymbol{P}^*_{\mathsf{ie}}) = (\boldsymbol{F}^{\mathrm{ref}}_e, \boldsymbol{P}^{\mathrm{ref}}_e)$, $\forall e$ and $(\boldsymbol{E}^*_{\mathsf{ie}}, \boldsymbol{S}^*_{\mathsf{ie}}) = (\boldsymbol{E}^{\mathrm{ref}}_e, \boldsymbol{S}^{\mathrm{ref}}_e)$, $\forall e$.

Note that this initialization is performed at the beginning of the simulation, *i.e.* for the first load step only. Between two load steps t and $t-1$, the material states remain unchanged:

$$(\boldsymbol{F}^*_{\mathsf{ie}}, \boldsymbol{P}^*_{\mathsf{ie}})^{(t,\,k=0)} = (\boldsymbol{F}^*_{\mathsf{ie}}, \boldsymbol{P}^*_{\mathsf{ie}})^{(t-1,\,k_{\max})} \tag{III.119a}$$

$$(\boldsymbol{E}^*_{\mathsf{ie}}, \boldsymbol{S}^*_{\mathsf{ie}})^{(t,\,k=0)} = (\boldsymbol{E}^*_{\mathsf{ie}}, \boldsymbol{S}^*_{\mathsf{ie}})^{(t-1,\,k_{\max})} \tag{III.119b}$$

with $k_{\max}$ the number of data-driven iterations performed in the previous load step.

Augmented Lagrangian parameters: the penalty parameters and Lagrange multipliers are initialized as previously.

Computer: all simulations are performed on a laptop with the following specifications: processor Intel Core i5-6200U CPU @ 2.30 GHz × 4, with 15.5 Gio of RAM.

With these settings, three kinds of data-driven simulations are performed: (i) using the Lagrangian solver, (ii) using the nominal solver, (iii) using the linear nominal solver, where the conservation of angular momentum is not enforced (see Section 2.2.4). The latter test is aimed at assessing whether this condition must be enforced in the constraint set when it is already satisfied by the material data set (see Conti et al. (2020, Theorem 3.5)).

3.2.2 Results and discussion

General results. We first compare the results obtained with the "zero" initialization method, for the three above-mentioned computations. Table III.3 compares the values of the objective function $\sum_e w_e\, d_e(z_e, z^*_{\mathsf{ie}})$ at convergence, for the final load step. First, the important result is that none of the three solvers reaches the global minimum: the alternated minimization is not able to find the reference solution, although it is entirely contained in the material data set. Second, from the computation times also listed in Table III.3, we confirm that the Lagrangian formulation finds a solution at a lower computational cost. Indeed, as aforementioned, at

Table III.3 – Value of the objective function at convergence, for the last load step of the clamped tensile test. The iterations correspond to the total number of local data assignment iterations throughout all 10 load steps.

Formulation	Objective function	Iterations	Computation time
Lagrangian	$4.46 \times 10^{-4}\,\mathrm{MPa\,mm^2}$	188	11 min
Nominal	$1.74 \times 10^{-3}\,\mathrm{MPa\,mm^2}$	256	94 min
Linear	$3.75 \times 10^{-3}\,\mathrm{MPa\,mm^2}$	175	19 min

a fixed data-driven iteration, the Lagrangian solver calls a Newton-Raphson scheme which successively builds and solves linear systems of size $(2n_{\mathrm{dof}})^2$. Conversely, the nominal formulation calls an augmented Lagrangian scheme, within which a Newton-Raphson iteration requires to build and solve a linear system of size $(Nn+n_{\mathrm{states}})^2$. Finally, the linear nominal solver only requires, within each augmented Lagrangian iteration, to solve two independent linear systems of size $(Nn)^2$ and $(n_{\mathrm{states}})^2$ respectively. In the latter case, the stiffness matrix of the system is constant and can be computed once and for all at the beginning of the simulation. In this particular example, $Nn = 680 \ll n_{\mathrm{states}} = Mn^2 = 6048$.

In Fig. III.7, we compare the Von Mises norm σ_{VM} of the Cauchy stress tensor $\boldsymbol{\sigma}$ with the reference solution, for all three data-driven computations:

$$\sigma_{\mathrm{VM}} = \sqrt{\frac{3}{2}}\,\mathrm{dev}(\boldsymbol{\sigma}_{3D}) : \mathrm{dev}(\boldsymbol{\sigma}_{3D}) \text{ with } \boldsymbol{\sigma}_{3D} = \begin{pmatrix} \boldsymbol{\sigma} & \mathbf{0} \\ \mathbf{0} & 0 \end{pmatrix}, \tag{III.120}$$

This stress tensor is of particular interest for comparing the data-driven results with the reference one as it combines the deformation and stress tensors that are actually optimized by the solvers. Indeed, the deformation and stress tensors are no longer related by the constitutive relation in the data-driven computation. A good prediction of the Cauchy stress tensor is then an indicator that both kinematics and stresses have been well optimized:

$$\boldsymbol{\sigma} = \boldsymbol{P}\,\boldsymbol{F}^T \text{ and } \boldsymbol{\sigma} = \boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T \tag{III.121}$$

as $J = \det \boldsymbol{F}_{3D} = 1$. As can be seen in Fig. III.7, the results are all very similar, indicating that albeit non-optimal, the data-driven solutions are very acceptable. Moreover, the homogeneity of the stress field in the top part of the structure (where uniaxial tension occurs) is well recovered by the Lagrangian and nominal solvers. A slight artifact is shown in this region by the linear nominal solver.

Initialization of material states. We now compare the different initialization methods on the first load step of the simulations in Fig. III.8. We demonstrate that the alternated minimization can diverge if the initial guess is too far from the solution, especially when using the nominal solver.

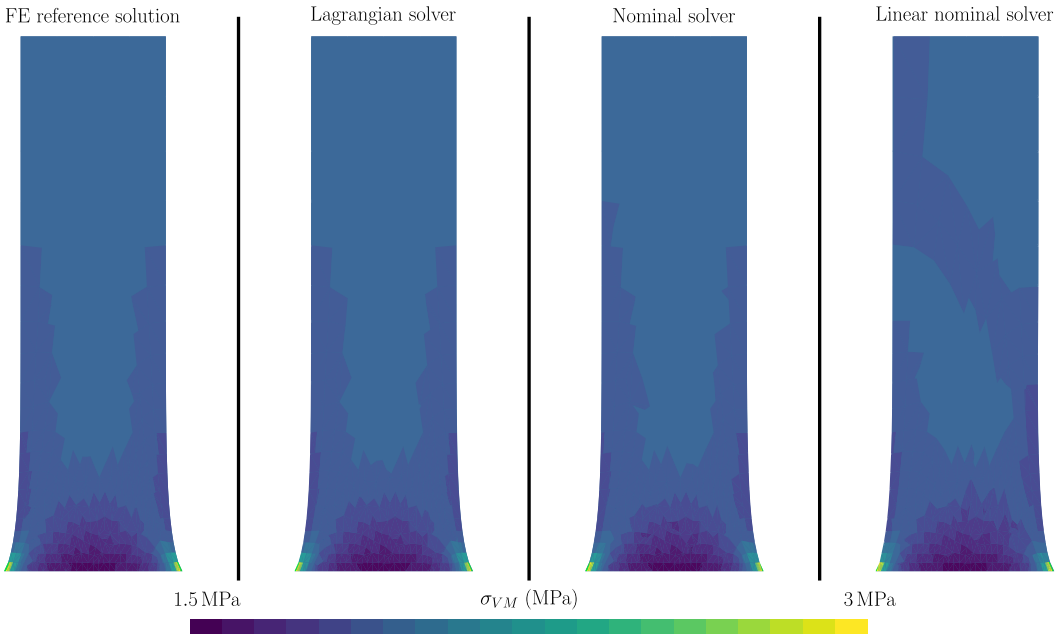


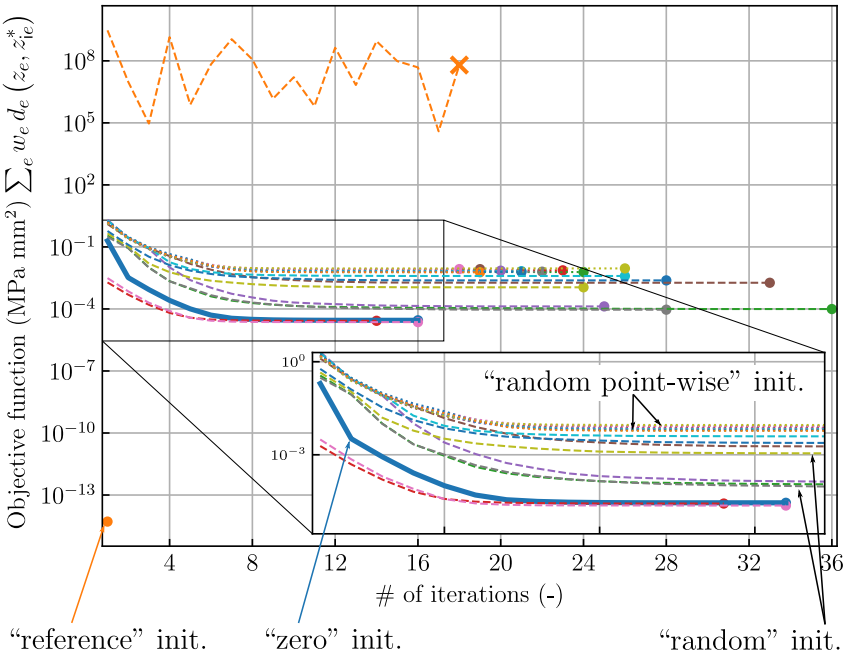
Figure III.7 – Filled isovalue view of the Von Mises norm σ_{VM} of the Cauchy stress tensor, plotted on the deformed configuration (21 isovalue regions).

On the left-hand bottom corner of both Figs. III.8(a) and III.8(b), we can see that, as expected, the "reference" initialization converged within one iteration and provides the global minimum of the objective function, which is a numerical zero for both formulations.

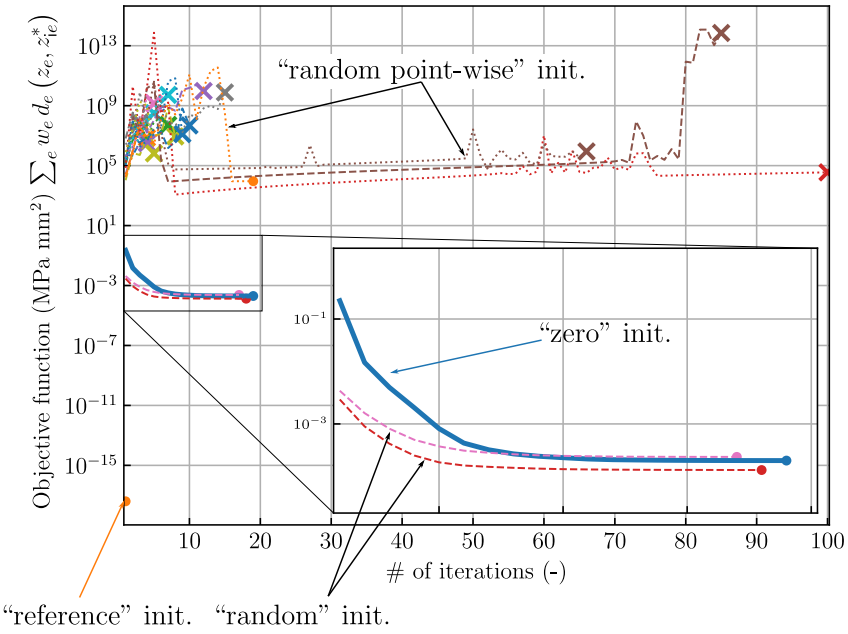
On the upper part of Fig. III.8(b), we show the convergence of the nominal solver for 8 and 10 different "random" and "random point-wise" initializations respectively. The initial value of the objective function (at $k = 1$) in these cases is already very high (about $1 \times 10^5\,\mathrm{MPa\,mm^2}$) and most of the simulations diverged (as indicated by a cross at the end of a line)[8]. It is worth noting that the Lagrangian solver only diverged once (see upper part of Fig. III.8(a)). A random initialization of the material states is then not recommended, whichever the finite strain solver.

The most robust and accurate choice is then "zero", as can be seen from the zoomed box of both Figs. III.8(a) and III.8(b): for example, the objective function smoothly decays from $7.37 \times 10^{-2}\,\mathrm{MPa\,mm^2}$ to $1.74 \times 10^{-3}\,\mathrm{MPa\,mm^2}$ within $k = 21$ iterations for the nominal solver. Furthermore, the "zero" initialization method actually behaves as a linear elastic initial guess. Indeed, let us get a closer look at the systems of equations that are to be solved at the first iteration of the first load step. For the Lagrangian solver, from Eqs. (III.39a) and (III.39b), the initial guess

[8] In fact, the initial material state is so far from the solution that the Newton-Raphson diverges right away. However, we let the simulation go on a bit to see if the alternated minimization can recover the path towards the solution, before stopping the simulation. This explains why the value of the objective function is increasing in some cases. Note that this issue is not possible in small strain as the minimization problem is quadratic with linear constraints.



(a) Lagrangian solver.



(b) Nominal solver.

Figure III.8 – Convergence of the data-driven finite strain solvers for the clamped tensile test. Comparison of different initialization methods: "zero" in solid line, "random" in dashed lines and "random point-wise" in dotted lines. A dot (respectively a cross) at the end of a line indicates that the solver did (respectively did not) converge. Results are given for the first load step only.

being $(\boldsymbol{u}, \boldsymbol{\eta}) = (\mathbf{0}, \mathbf{0})$ (see also Line 5 in Algorithm III.1) and the initial material states being $(\boldsymbol{E}^*_{\text{ie}}, \boldsymbol{S}^*_{\text{ie}}) = (\mathbf{0}, \mathbf{0})$, the computation of the mechanical states then write,

$$\boldsymbol{R}^{\eta}_a(\boldsymbol{u} = \mathbf{0}, \boldsymbol{\eta} = \mathbf{0}) = -\boldsymbol{T}^{\text{ext}}_a, \qquad \forall a \in [1\,..\,N] \tag{III.122a}$$

$$\boldsymbol{R}^{u}_a(\boldsymbol{u} = \mathbf{0}, \boldsymbol{\eta} = \mathbf{0}) = \sum_{e=1}^{M} w_e\, \boldsymbol{F}_e\, (\mathbb{C} : \boldsymbol{E}_e)\,, \qquad \forall a \in [1\,..\,N], \tag{III.122b}$$

where the summation term in $\boldsymbol{R}^u_a$ vanishes everywhere except at the integration points that are affected by the prescribed displacements. This system is merely the application of the boundary and loading conditions on the sample. The first estimated displacements then result from the linear elastic resolution of the system Eq. (III.122), with stiffness $\mathbb{C}$, as $\boldsymbol{E}_e \simeq \boldsymbol{\varepsilon}_e$ and $\boldsymbol{F}_e \simeq \boldsymbol{I}$ if the load step is small enough. This first estimator seems more reasonable than imposing a random value of material states and mechanical states, which act as pre-strain and pre-stress conditions on the structure. The same result can be shown for the nominal solver, where the stiffness of the system is $\mathbb{C} = C\,\mathbb{I}$, with $\mathbb{I}$ the fourth-order identity tensor ($I_{ijkl} = \delta_{ik}\delta_{jl}$) and C the parameter in the deviation functions Eq. (III.67).

Conservation of angular momentum. As shown earlier, the linear nominal solver greatly reduces the computational cost of the nominal formulation. To assess the quality of the corresponding solution, we now evaluate the conservation of angular momentum in the results. We refer to the vector $\boldsymbol{r} = \boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}\boldsymbol{P}^T\right)$ in Eq. (III.62) as the *rotational residual*. In addition, we define the *rotational residual percent error* $\epsilon_{(\%\text{rot})}$ as the ratio between the rotational residual and the stored strain energy density:

$$\epsilon_{(\%\text{rot})} = \frac{\|\boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}\boldsymbol{P}^T\right)\|_2}{\boldsymbol{F} : \boldsymbol{P}}. \tag{III.123}$$

Note that, in two-dimensional problems, the rotational residual is a one-dimensional vector, *i.e.* a scalar; specifically,

$$\|\boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}\boldsymbol{P}^T\right)\|_2 = |F^e_{11}P^e_{21} + F^e_{12}P^e_{22} - (F^e_{21}P^e_{11} + F^e_{22}P^e_{12})\,|. \tag{III.124}$$

The rotational residual percent error is computed for the nominal and linear nominal solvers respectively. The results are given in Fig. III.9, together with the rotational residual of the linear nominal solver solution. As can be seen from Fig. III.9(b), the linear nominal solver does not recover the conservation of angular momentum. However, the error does not exceed 10 % of the stored strain energy density $\boldsymbol{F} : \boldsymbol{P}$. Let recall that in the present case, the material database exactly satisfies moment equilibrium since it was obtained from standard FE computations.

3.3 More complex case: membrane with a hole

We now turn to a more complex two-dimensional problem. We study a thin membrane with a hole subject to simultaneous shear and tension. With this example, we evaluate the nominal and Lagrangian formulations separately:

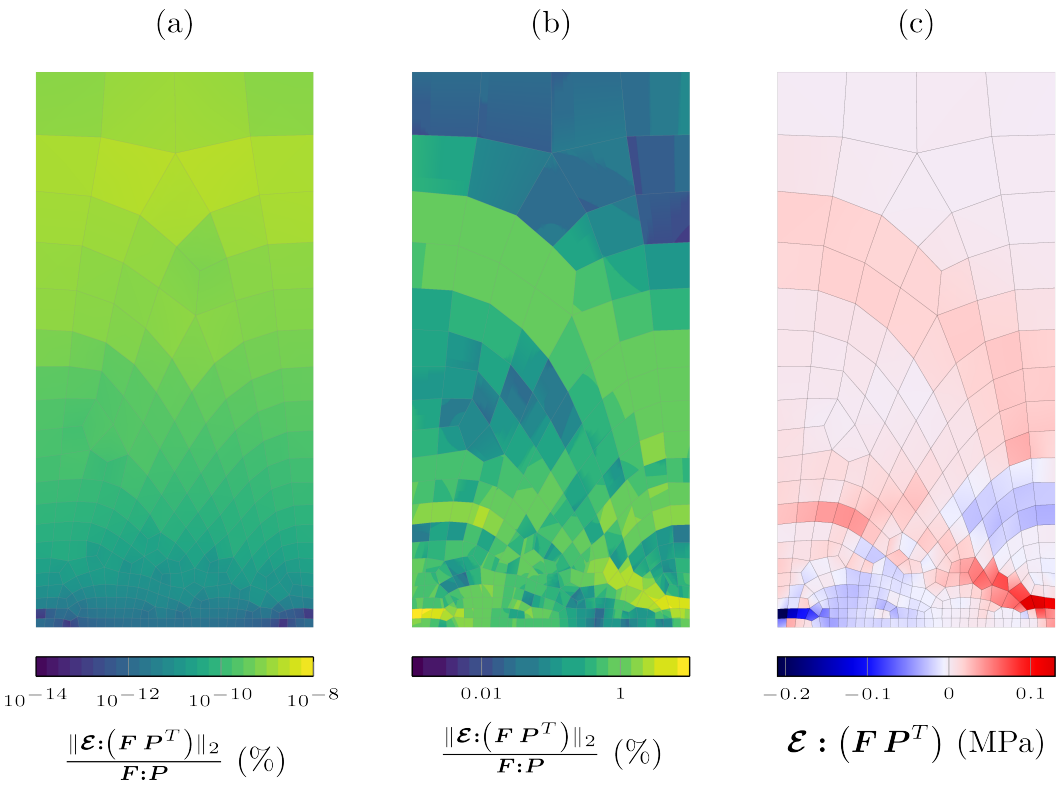


Figure III.9 – Conservation of angular momentum in the data-driven results for the clamped tensile test. (a) and (b) Rotational residual percent error. (a) Nominal solver solution. (b) Linear nominal solver solution. The color map is in log scale. As a comparison, the corresponding maximum value for the reference solution is 1.07×10^{-14} %. (c) Rotation residual $\boldsymbol{r}$ (one-dimensional vector plotted as a scalar) for the linear nominal solver.

1. In the previous example, the strain and stress fields were homogeneous in most of the structure, which could explain the relatively low impact of ignoring the rotational equilibrium constraint. With the addition of the hole and the more complex boundary conditions, we expect to see more heterogeneous strain and stress fields. We thus evaluate the linear nominal solver with a more challenging test case.

2. We then evaluate the convergence of the Lagrangian data-driven solution with the number of material data points: we develop a method to artificially enrich the material database for the FE reference solution.

3.3.1 Computational problem

The geometry, mesh and loading of the sample are given in Fig. III.10. The mesh comprises $N = 1092$ nodes and 1015 bi-linear quadrangular elements with 4 integration points each, which amounts to $M = 4060$ integration points. The boundary

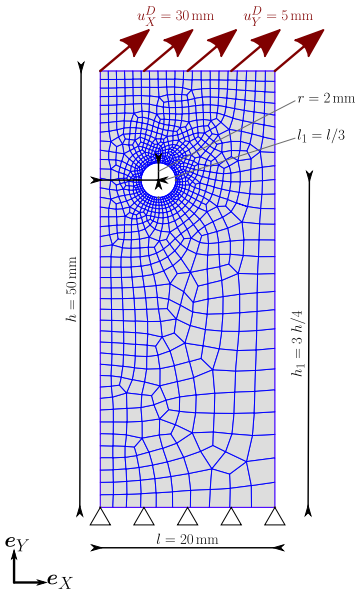


Figure III.10 – Geometry, boundary conditions and mesh for the membrane with a hole.

conditions are also represented in Fig. III.10 and write

$$\boldsymbol{u}_a = \boldsymbol{0}, \qquad \forall a \in [1\,..\,N] : \boldsymbol{X}_a = (X_a, 0); \tag{III.125}$$

$$\boldsymbol{u}_a = u_X^D \boldsymbol{e}_X + u_D^Y \boldsymbol{e}_Y, \qquad \forall a \in [1\,..\,N] : \boldsymbol{X}_a = (X_a, h); \tag{III.126}$$

with $u_X^D = 30\,\mathrm{mm}$ and $u_Y^D = 5\,\mathrm{mm}$. The simulation is again performed incrementally: the displacements are gradually prescribed in 40 regular steps.

As previously, the material database is only constituted with the classical FE solution of the problem, obtained with the incompressible neo-Hookean model, formulated in plane stress Eq. (III.113). For all 40 load steps, we concatenate the reference strain-stress state of all integration points, which yields

$$\mathcal{D}_{\mathrm{loc}}^{\mathrm{Lag}} = \left\{ \left\{ \left(\boldsymbol{E}_e^{\mathrm{ref},t}, \boldsymbol{S}_e^{\mathrm{ref},t} \right) \right\}_{e=1}^{M} \right\}_{t=1}^{40} \tag{III.127a}$$

$$\mathcal{D}_{\mathrm{loc}}^{\mathrm{nom}} = \left\{ \left\{ \left(\boldsymbol{F}_e^{\mathrm{ref},t}, \boldsymbol{P}_e^{\mathrm{ref},t} \right) \right\}_{e=1}^{M} \right\}_{t=1}^{40} \tag{III.127b}$$

For both formulations and following the objectives states above, we generate additional databases:

- We recall that the conservation of angular momentum is, in theory, equivalent to the principle of material-frame indifference. Hence, in the nominal formulation, it could be enforced in the material data set, rather than in the constraint set, by enriching the database with the orbits of every material data points.

That is, the local material data set is now:

$$\mathcal{D}_{\text{loc}}^{\text{nom},Q} = \left\{ \left\{ (\boldsymbol{Q}\boldsymbol{F}_e^{\text{ref}}, \boldsymbol{Q}\boldsymbol{P}_e^{\text{ref}}) \right\}_{e=1}^{M} \right\}_{t=1}^{40}, \tag{III.128}$$

with $Q \in SO(2)$. A discretization of $SO(2)$ is easily parameterized by one angle $\theta \in [0, 180)$. We compare the results obtained with three databases of increasing fidelity to material frame-indifference, generated *via* regular samplings of θ, with decreasing step $\Delta\theta \in \{15°; 10°; 5°\}$.

- For the Lagrangian formulation, we generate additional material databases which aim at densifying the region of phase space covered by the solution. To this end, we add to the material database $\mathcal{D}_{\text{loc}}^{\text{Lag}}$ the reference solution of the same problem with the same loading path, generated with coarser and finer meshes of the sample. The resolution of the corresponding meshes are given in Table III.4. The size of the resulting material databases are given in Table III.5.

Table III.4 – Resolutions of the meshes used to generate the Lagrangian material databases for the complex test case.

Mesh number	Nodes	Integration points
1	250	864
2	589	2136
3	1092	4060
4	1655	6228
5	2245	9500

Table III.5 – Size of the Lagrangian material databases generated with decreasing or increasing mesh resolutions of the reference solution.

Database number	Meshes	Size
(0)	3	162 400
(1)	1	8640
(2)	1 + 2	30 000
(3)	1 + 2 + 3	70 600
(4)	3 + 4 + 5	197 880
(5)	4 + 5	157 280
(6)	5	95 000

Finally, based on the preceding analysis, the solution is achieved from the "zero"

initialization method only, for all solvers. All other parameters of the simulations are unchanged from the previous example.

3.3.2 Results and discussion

General results. Again the global minimum of the problem is not found by the alternated minimization, which ever the solver, as shown by the values of the objective function listed in Table III.6. Note that the computation time was divided by a factor 2 with the linear solver as compared to the nominal solver. Nevertheless, the Lagrangian solver is definitely faster.

Table III.6 – Value of the objective function at convergence, for the last load step of the complex loading case. The iterations correspond to the total number of local data assignment iterations throughout all of the 40 load steps.

Formulation	Objective function	Iterations	Computation time
Lagrangian	$2.98 \times 10^{-2}\,\mathrm{MPa\,mm^2}$	491	1.57 h
Nominal	$2.46 \times 10^{-1}\,\mathrm{MPa\,mm^2}$	962	20.22 h
Linear nominal	$3.98 \times 10^{-1}\,\mathrm{MPa\,mm^2}$	1191	12.1 h

Conservation of angular momentum in the nominal formulation. We again measure the rotational residual percent error in Fig. III.11. In this more complex case, the error made by the "linear" solver is higher: it is mostly comprised between 1 % and 10 % but it can reach over 100 % around the holes.

The local error at integration points also has an impact on the global response of the structure, as measured by the loading curve of the simulation. We compare in Fig. III.12 the displacement-force response of the clampled tensile test with the one of the complex case for both solvers. In the clamped tensile test where most of the structure is subject to uniaxial tension only, the impact of the error in rotational equilibrium on the reaction forces is invisible. Conversely, the local error made in the membrane with a hole weakens the accuracy of the reaction forces at the grip.

Finally, we evaluate whether the above results could be improved with the material databases $\mathcal{D}_{\mathrm{loc}}^{\mathrm{nom},Q}$, enriched with the orbits of the reference solution as in Eq. (III.128). We analyze in Fig. III.13 the statistics of the rotational residual for each database $\mathcal{D}_{\mathrm{loc}}^{\mathrm{nom},Q}(\Delta\theta)$ with $\Delta\theta \in \{5°, 10° \text{ and } 15°\}$. As can be seen from the figure, the rotational residual is not lowered with this method (in particular, median and mean values are almost unchanged).

Density of the material database for the Lagrangian formulation. First, we examine the data-driven solution obtained with the database (0), *i.e.* the one containing the reference solution of the current mesh resolution only. In Fig. III.14, we compare the component of the Cauchy stress tensor with the reference solution. As can be seen from the figure, the results are indistinguishable to the eye.

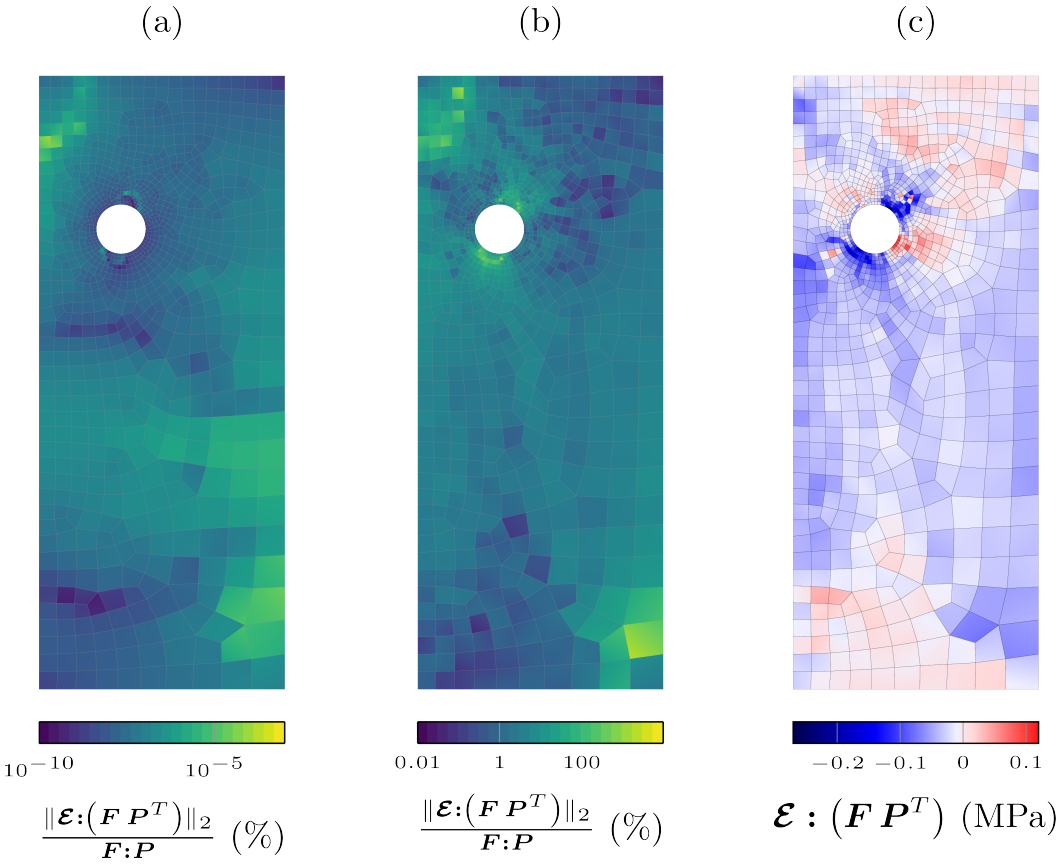


Figure III.11 – Conservation of angular momentum in the data-driven results for the membrane with a hole. (a) and (b) Rotational residual percent error. (a) Nominal solver solution. (b) Linear solver solution. The color map is in log scale. As a comparison, the corresponding maximum value for the reference solution is 1.93×10^{-11} %. (c) Rotation residual (one-dimensional vector plotted as a scalar) for the "linear" solver.

Second, the convergence of the strain and stress RMS percent errors Eq. (III.115) with respect to the size of the data set is shown in Fig. III.15. The solution is slightly improved with the database (3), *i.e.* when the solution is added to the database (2). However, the impact is not so clear when the solution is added to the databases containing the finer meshes (database (4)). This mitigates our intuition that refining the mesh of the FE solution used to generate the database actually densifies the appropriate regions of phase space. Alternatively, the more the material data points, the more complex the combinatorial problem. There could be a competition between the complexity of the optimization problem, degrading the solution due to a non-optimal solver, and the density of the database, expected to improve the results.

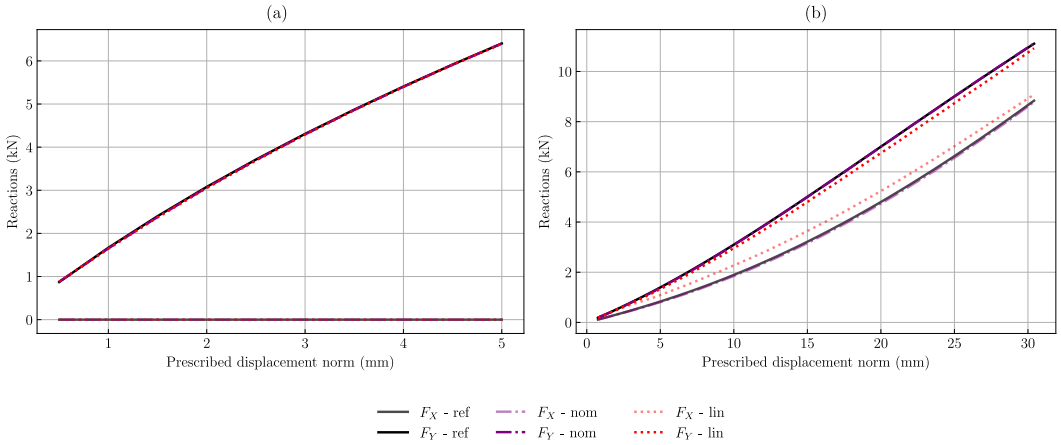


Figure III.12 – Loading curves for the clamped tensile test (a) and the complex case (b). In the legend, "nom" designates the nominal solver, "lin" designates the "linear" nominal solver and "ref" corresponds to the FE reference solution.

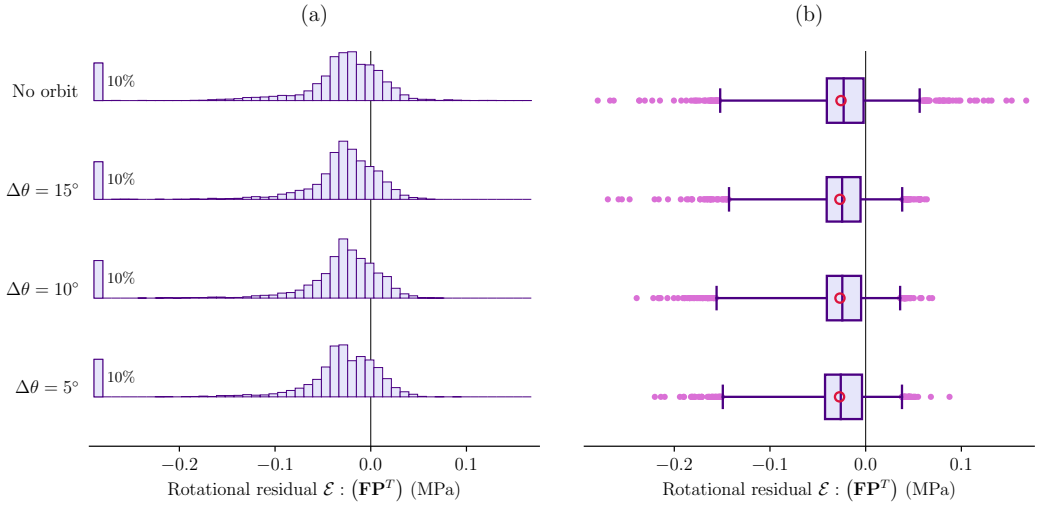


Figure III.13 – Rotational residual at every integration point. The databases are obtained with different angular discretizations $\Delta\theta$ of the orbit $Q \in SO(2)$. (a) Histogram of the data, in percent of number of integration points. At the furthest left, a bar corresponding to 10 % of occurrence is given for scale. (b) The box extends from the 25^{th} and 75^{th} percentile of the data, with a line at the median. The red circle stands for the mean value. The whiskers show the 1^{th} and 99^{th} percentiles. Small purple dots represent data points out of that range (fliers).

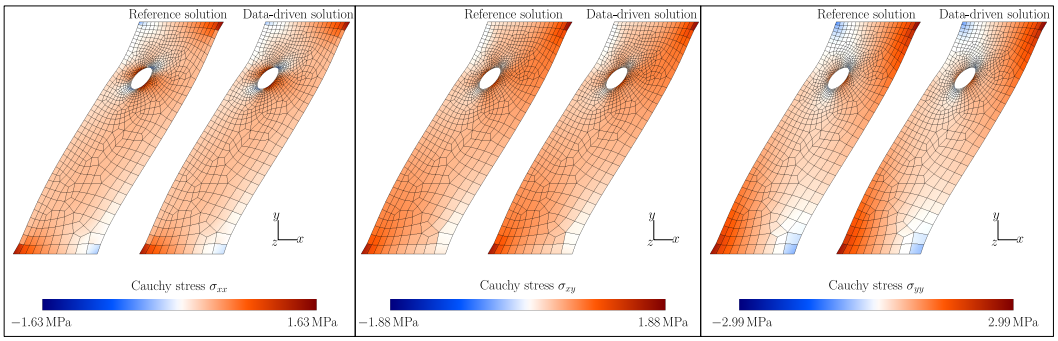


Figure III.14 – Components of the Cauchy stress tensor, plotted on the deformed configuration, obtained with the Lagrangian solver.

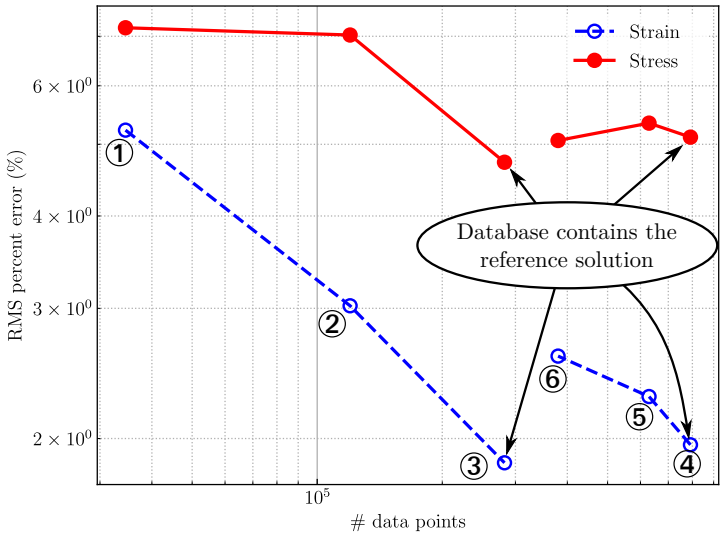


Figure III.15 – Convergence of the RMS percent errors in strain and stress with respect to the size of the database, for the complex test case and the Lagrangian solver. A number in circle indicates to the number of the corresponding database from Table III.5.

4 Conclusion

We conclude this chapter with a summary of the main features of the data-driven finite strain solvers that we developed and of the first numerical results we obtained.

The extension of the data-driven computing paradigm to finite strain elasticity required to reformulate the problem in another strain-stress or phase space. We explored two possibilities: (i) the so-called Lagrangian phase space, constituted with Green-Lagrange strain–second Piola Kirchhoff stress tensors and (ii) the so-called nominal phase space of deformation gradient–first Piola-Kirchhoff stress pairs. The former choice was first introduced by Nguyen and Keip (2018) while the latter was recommended by Conti et al. (2020), based on mathematical considerations. Based on their pioneer work, we proposed a generic FE solver for each formulation. While the Lagrangian solver is greatly inspired by the work of Nguyen and Keip (2018), the nominal solver is an original production of this thesis, to the best of our knowledge. Both approaches rely on the alternated minimization scheme that was introduced in the original DDCM (Kirchdoerfer and Ortiz, 2016). The optimization of the material states is then conducted with an efficient tree-based nearest neighbor search algorithm. The constrained minimization of the mechanical states is addressed differently in each formulation: (i) the Lagrangian formulation uses a method of Lagrange multipliers, and (ii) the nominal formulation uses an augmented Lagrangian method. The deviation functions Eq. (III.66) proposed by Conti et al. (2020) as functions of the p-th power and q-th power of the tensors norms are not differentiable at zero, except when $(p, q) = (2, 2)$, which was not recommended by the authors. Yet, we have not developed a solver capable of handling other values of (p, q), and hence could not numerically assess the statements made in their paper.

Nevertheless, we successfully evaluated our solvers with the aid of three sanity check examples, in two dimensions. In particular, we demonstrated that the alternated minimization is, as expected, not able to find the global minimum of the combinatorial and optimization problem: the reference solution is not recovered even if the material database does contain it. This statement is however mitigated by the facts that (i) the solution improves as the density of the database increases, and (ii) the solution is satisfactory enough when comparing the Cauchy stress field with the reference solution.

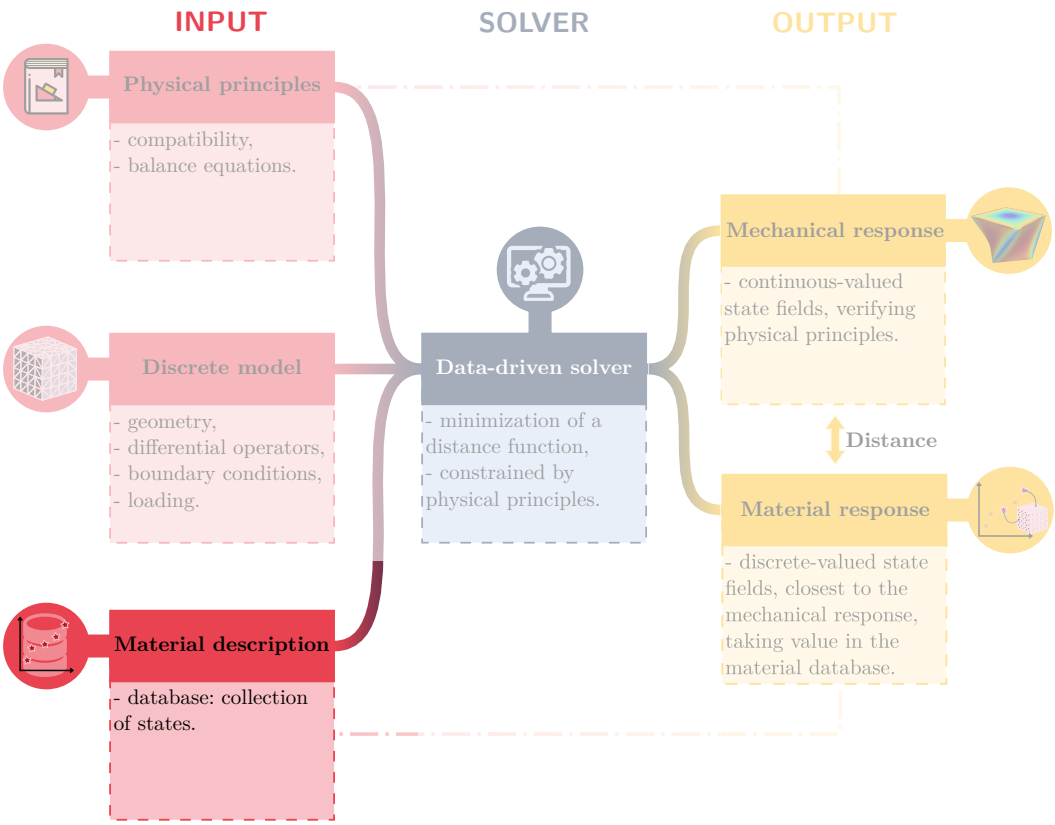
Along with the complete nominal formulation, we investigated the possibility to relax the enforcement of conservation of angular momentum, as it constitutes the only nonlinear constraint to the minimization problem. The solver is then simplified into a so-called linear nominal solver, which provides a substantial gain in computational cost. Notwithstanding, its systematic use is subject to the upcoming proof that the consequent error in moment balance is bounded. We also explored a workaround which consists in encoding material-frame indifference in the material data set instead: we enriched material data points with their orbits in $SO(2)$, in an off-line stage. This method did not lower the error in conservation of angular momentum as expected. Another possibility could be to minimize the orbit of the material data points in-line, in the search for the optimal material states, in like

manner as the approach presented for geometrically linear elasticity in Kirchdoerfer and Ortiz (2016). Until future investigation completely elucidates this question, we recommend enforcing conservation of angular momentum as a constraint, to avoid misleading results. Note that in the Lagrangian solver, this constraint is directly encoded in the symmetry of the stress tensor $\boldsymbol{S}$.

All-in-all the Lagrangian solver outperforms, for now, its nominal counterpart. Thus, we used the former to start investigating the notion of density of the material database in our last example. We tried enriching the material database by refining the mesh used to generate the strain-stress states from classical FE method. The convergence to the reference solution was little improved with this approach, demonstrating that data coverage and data sampling of the phase space are not easy to control. Indeed, the multi-dimensionality of the phase space makes these concepts difficult to grasp. The next chapter is then entirely dedicated to the generation and analysis of optimal material databases and their implication on the data-driven solution.

Chapter IV

Generation of material databases



Contents

1 Introduction

1.1 Motivation

The material database is a key ingredient of the data-driven computing paradigm. As shown in the previous chapter, in large strain, it can consist either of Lagrangian strain-stress pairs $(\boldsymbol{E}, \boldsymbol{S})$ or of deformation gradient-nominal stress pairs $(\boldsymbol{F}, \boldsymbol{P})$. In practice, these tensors are expressed in the canonical basis $\mathcal{B} = (\boldsymbol{e}_X, \boldsymbol{e}_Y, \boldsymbol{e}_Z)$. The material database is then a collection of the tensors components in this basis. For a two-dimensional problem, the material database for the Lagrangian and nominal formulations respectively writes

$$\mathcal{D}^{\text{Lag}}_{\text{loc}} = \left\{ \left(E^i_{XX}, E^i_{XY}, E^i_{YY}, S^i_{XX}, S^i_{XY}, S^i_{YY} \right) | i \in [1 \,..\, m] \right\} \tag{IV.1a}$$

$$\mathcal{D}^{\text{nom}}_{\text{loc}} = \left\{ \left(F^i_{XX}, F^i_{XY}, F^i Y X, F^i_{YY}, P^i_{XX}, P^i_{XY}, P^i_{YX}, P^i_{YY} \right) | i \in [1 \,..\, m] \right\}, \tag{IV.1b}$$

with m the number of material data points, and where the symmetry of the Lagrangian strain and stress tensors $\boldsymbol{E}$ and $\boldsymbol{S}$ has been taken into account. Note that the Lagrangian material database $\mathcal{D}^{\text{Lag}}_{\text{loc}}$ is then 6-dimensional whereas the nominal data set $\mathcal{D}^{\text{nom}}_{\text{loc}}$ is 8-dimensional. The questions addressed in the present chapter are:

1. how can a material database, such as Eq. (IV.1), be obtained for an homogeneous material?

2. how many material data points are needed and how must they span the phase space for a data-driven simulation to be reliable?

We restrict our investigation to only one formulation of the data-driven two-dimensional finite strain elasticity. We choose the Lagrangian approach, as it requires a material database of the smallest dimension. The first question is answered in the present introduction: we next propose general considerations on the acquisition (or generation) of material data. The second question is addressed in the remaining of the chapter: first analytically in Section 2 and then through a numerical application in Section 3.

1.2 Experimental point of view

The data-driven computing has been introduced by Kirchdoerfer and Ortiz (2016) has a way of by-passing the constitutive modeling which introduces a bias in the representation of the material response. In the simulations performed afterwards, this bias is inevitably transfered to the structural response. Instead, with the development of advanced data acquisition techniques for the characterization of materials, it seems now possible to use directly the material response data as input of structural simulations.

As mentioned in Section 2.2 of Chapter II, the acquisition of material strain-stress pairs from complex experimental tests and full-field measurements is now possible with the approach developed by Dalémat (2019). It is based on the inverse

data-driven problem introduced by Leygue et al. (2018), which we referred to as the DDI technique. The output of the DDI approach is double: (i) the mechanically admissible stress fields, in balance with the (known) applied forces, (ii) the strain-stress pairs which sample the material response and act as a regularization of the problem. The latter constitutes in fact the material database we need as input of the data-driven simulation. For a first application of the DDI-DDCM loop, the reader can refer to the proof of concept of Stainier et al. (2019) with synthetic data.

In Dalémat et al. (2019), the method has been assessed for two-dimensional problems, under plane stress conditions for incompressible materials. In particular, they conducted tests on a thin membrane of elastomer with several holes, designed to generate heterogeneous strain and stress fields. The material databases thus obtained are expressed as Hencky strain-Cauchy stress pairs $(\boldsymbol{H}^*, \boldsymbol{\sigma}^*)$. The Hencky strain is defined as $\boldsymbol{H}^* = \ln \boldsymbol{V}^*$, where $\boldsymbol{V}^*$ is obtained from the spatial polar decomposition of $\boldsymbol{F}^* = \boldsymbol{V}^* \boldsymbol{R}^*$ and ln denotes the natural logarithm. To fit such data sets into the Lagrangian formulation (or even the nominal), one would need to pull-back the stress tensor $\boldsymbol{\sigma}^*$ in the reference configuration, as

$$\boldsymbol{S} = J \boldsymbol{F}^{-1} \boldsymbol{\sigma} \boldsymbol{F}^{-T}. \tag{IV.2}$$

However, the material deformation gradient $\boldsymbol{F}^* = \boldsymbol{V}^* \boldsymbol{R}^*$ is not available with this approach as the corresponding rotation $\boldsymbol{R}^*$ is not computed in the process. We only have access to the mechanically admissible deformation gradient tensor $\boldsymbol{F} = \boldsymbol{\nabla}_0 \boldsymbol{u} + \boldsymbol{I}$ at each point in the membrane, which directly derives from the displacement field $\boldsymbol{u}$, measured by DIC. In DDI, the material states are the centroids of clusters of mechanical states in phase space. Then, several mechanical states are associated to the same material data point. It is then not clear how to choose the right mechanical deformation gradient tensor to transform the spatial pairs $(\boldsymbol{H}^*, \boldsymbol{\sigma}^*)$ into the Lagrangian pair $(\boldsymbol{E}^*, \boldsymbol{S}^*)$ through the pull-back operation Eq. (IV.2). In addition, the heterogeneity of the fields in the structure was associated to the "richness" of the subsequent material database. However, as shown by Dalémat (2019) at the end of Chapter 6, it is very difficult to generate strain states that differ from uniaxial tension and pure shear with a single actuator. In Chapter 8, a second actuator is added to perform multiaxial experiments on the membrane (complex loading with tension, shear and rotations). Again, the resulting "richness" of the strain state is not clear: a lot of points in the membrane are in uniaxial tension or pure shear state. Given the two above-mentioned limitations, and since the results are very new, it was difficult to directly use them in the present thesis.

All-in-all, the generation of experimental material databases suitable for DDCM simulations is an open and challenging subject. We then chose the assess the robustness of the data-driven finite strain solvers we developed in this thesis only with synthetic, *i.e.* numerical, material data. More precisely, the above-mentioned experimental results suggest that a sample, however complex, may only exhibit a limited variety of modes of deformation. Hence, it may not be necessary for the material database to be the richest possible (to contain the material response to any mode of deformation, in any direction), to perform DDCM simulations of a given

structure. In the following, we explore this question by artificially generating material databases of controlled richness, *i.e.* which contain a tailored number of modes of deformation.

1.3 Numerical point of view

Material databases for the data-driven problem can also be created synthetically: either with a constitutive model to compare with the FE reference solution or from the computational homogenization of the overall behavior of a given microstructure. In both cases, we must sample an appropriate region of the strain (stress) space and compute the associated stress tensors via the analytical or numerical (inverse) constitutive relation. As aforementioned, the strain space is understood as the space spanned by the independent components of the strain tensor. For three-dimensional problems in small strain, it consists of $\{\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{33}, \varepsilon_{12}, \varepsilon_{13}, \varepsilon_{23}\} \subset \mathbb{R}^6$. In all data-driven applications so far, the strain space is sampled regularly on hyper-cubes of pre-specified range in $\mathbb{R}^3$ or $\mathbb{R}^6$ (Eggersmann et al., 2020; Kirchdoerfer and Ortiz, 2016; Nguyen et al., 2020). However, as shown by Stainier et al. (2019), databases generated from regular grids in strain space behave poorly, as compared to more carefully designed data sets. In addition, if the regular grid can seem easy and appealing in small strain, it is not suitable for large strain problems.

Indeed, let us consider a two-dimensional problem. One would need to sample the strain space generated by the three independent components of the Green-Lagrange strain tensor, namely $\{E_{11}, E_{12}, E_{22}\} \subset \mathbb{R}^3$. Depending on the range of the grid used for sampling, the resulting strain tensor could be ill-defined, *i.e.* correspond to deformation gradient tensor of negative Jacobian. For instance, let us consider a regular grid of 10^3 points in strain space, of dimensions given by the FE reference response of the membrane presented in the last numerical example of Chapter III. Then, several points yield a negative Jacobian $J < 0$, included

$$\boldsymbol{E} = \begin{pmatrix} -0.213 & 0.303 \\ 0.303 & -0.189 \end{pmatrix} \Rightarrow J = \sqrt{\det \boldsymbol{C}} = \sqrt{\det\left(2\boldsymbol{E} + \boldsymbol{I}\right)} = -0.197 < 0.$$

Hence, sampling the strain space with a regular grid can lead to non-physical data points in large strain material databases.

For both reasons (poor performance in data-driven simulation and lack of physical consistency), we did not adopt this method in our thesis. Instead, in the next section, we explore three different ways of generating synthetic material databases for finite strain simulations:

1. Based on the work of Kunc and Fritzen (2019a), we generate as dense and as rich as possible material databases (Section 2.1).

2. In Section 2.2, we synthetically reproduce the three standard tests (uniaxial tension, pure shear, and equi-biaxial tension) usually used to calibrate hyper-elastic constitutive models.

3. We simulate the output of the DDI in Section 2.3, as it would result from a Lagrangian re-formulation of Leygue et al. (2018); Dalémat et al. (2019).

Finally, in Section 3, we analyze the prediction of data-driven simulations obtained with these databases. In particular, we compare them with a FE reference solution to assess their respective performance. We also analyze the data-driven results for themselves, without a reference solution.

2 Synthetic databases

As above-mentioned, synthetic databases are generated by means of sampling a constitutive model. Here, we restrict ourselves to two-dimensional problems arising in plane stress conditions and incompressibility of the material. We then use the purely two-dimensional neo-Hookean model employed in Chapter III and presented in Appendix C.

$$\check{\boldsymbol{S}}(\boldsymbol{C}) = \mu \left(\boldsymbol{I} - (\det \boldsymbol{C})^{-1} \boldsymbol{C}^{-1} \right) \in \mathbb{R}^{2\times 2}_{\text{sym}} \tag{IV.3}$$

for all $\boldsymbol{C} \in \mathbb{R}^{2\times 2}_{\text{sym}}$.

2.1 Dense sampling method

The first sampling strategy adopted herein consists of a dense sampling of the space of admissible stretch tensors $\boldsymbol{U}$, the symmetric positive definite part of the polar decomposition of $\boldsymbol{F} = \boldsymbol{R}\boldsymbol{U}$, with $\boldsymbol{R}$ a rotation tensor. The method was first introduced by Kunc and Fritzen (2019a) as a way of sampling the space of admissible macroscopic deformation gradient tensors. It was developed for the purpose of efficient computational homogenization of hyperelastic solids based on a surrogate model approach, but it applies whenever one needs to generate a database of finite strain deformation tensors. It is then of particular interest in our case, where we aim at building a discrete material database for the Lagrangian formulation of the data-driven BVP. Indeed, we need to build a (high) number of pairs of Green-Lagrange strain–second Piola-Kirchhoff stress tensors $(\boldsymbol{E}_i, \boldsymbol{S}_i)$, which should cover as homogeneously as possible an appropriate region of the phase space $(\boldsymbol{E}, \boldsymbol{S})$. We can use the dense sampling approach of Kunc and Fritzen (2019a) in our case by recalling that

$$\boldsymbol{E} = \tfrac{1}{2} \left(\boldsymbol{U}^2 - \boldsymbol{I} \right). \tag{IV.4}$$

We then generate Lagrangian material databases for finite strain DDCM as follows:

1. We sample the space of admissible $\boldsymbol{U}$ following the method of Kunc and Fritzen (2019a);

2. We compute the corresponding Green-Lagrange strain tensors from Eq. (IV.4);

3. We compute the corresponding second Piola-Kirchhoff stress tensors from the neo-Hookean model $\check{\boldsymbol{S}}$ given in Eq. (IV.3) as

$$\boldsymbol{S} = \check{\boldsymbol{S}}\left(\boldsymbol{C}\right) = \check{\boldsymbol{S}}\left(\boldsymbol{C} = \boldsymbol{U}^2\right). \tag{IV.5}$$

From here on out, we refer such databases as dense material database (DB-DENSE).

The present section is organized as follows: first we present the original method of Kunc and Fritzen (2019a). Second, we adapt it to two-dimensional problems arising in plane stress conditions and incompressibility of the material. Third, we provide a physical interpretation of the method, based on the invariants of the Hencky strain tensor $\boldsymbol{H} = \ln \boldsymbol{V}$, which were introduced by Criscione et al. (2000).

2.1.1 An amplitude-direction split strategy

Let us present the main features of Kunc and Fritzen (2019a) method to sample the "space of practically relevant stretch tensors", *i.e.* tensors $\boldsymbol{U}$ such that $J = \det \boldsymbol{F} > 0$ with $\boldsymbol{F} = \boldsymbol{R}\boldsymbol{U}$. First, the authors apply the Dilatational-Deviatoric Multiplicative Split (DDMS) to the stretch tensor $\boldsymbol{U}$ such that

$$\boldsymbol{U} = J^{1/3}\hat{\boldsymbol{U}} \tag{IV.6}$$

The deviatoric stretch tensor $\hat{\boldsymbol{U}}$ is purely deviatoric, *i.e.* $\det \hat{\boldsymbol{U}} = 1$. The set of stretch tensors $\mathcal{U}$ can be sampled *via* sampling both the determinants $\left\{J^{(m)}\right\}_{m=1}^{N_{\text{det}}} \subset \mathbb{R}^+$ and the deviatoric stretch tensors $\left\{\hat{\boldsymbol{U}}^{(j)}\right\}_{j=1}^{N_{\text{dev}}}$, where N_{det} and N_{dev} are the number of samples. The sampling set is then determined by the product set (Kunc and Fritzen, 2019a)

$$\left\{\left(J^{(m)}\right)^{1/3}\hat{\boldsymbol{U}}^{(j)}\right\}_{m,j=1}^{m=N_{\text{det}},j=N_{\text{dev}}} \subset \mathcal{U}. \tag{IV.7}$$

Providing a set of deviatoric stretch tensors $\hat{\boldsymbol{U}}$ for a numerical simulations boils down to providing a set of *matrices* $\hat{U}$ representing the tensors, *i.e.* a set of the components of the tensors in a given basis. Kunc and Fritzen (2019a) built upon Lie group theory to sample the space of matrices $\hat{U}$. The matrices $\hat{U}$ are *unimodular*, *i.e.* symmetric, positive definite and of determinant 1. The "manifold" SymSL_+ of unimodular matrices then writes

$$\mathrm{SymSL}_+ = \left\{U \in \mathbb{R}^{3\times 3} \mid U = U^T,\, \det U = 1,\, x^T\, U\, x > 0,\, \forall x \in \mathbb{R}^3\right\}. \tag{IV.8}$$

The "tangent space" of symmetric matrices of trace 0 is written

$$\mathrm{symsl} = \left\{Y \in \mathbb{R}^{3\times 3} \mid Y = Y^T,\, \mathrm{tr}\, Y = 0\right\}. \tag{IV.9}$$

Then, the matrix exponential maps the tangent space symsl bijectively onto the manifold SymSl_+:

$$\begin{aligned} \exp :\quad & \mathrm{symsl} \to \mathrm{SymSl}_+ \\ & Y \mapsto U = \exp Y. \end{aligned} \tag{IV.10}$$

As the authors point out, the set SymSL_+ is the set of matrix representations of deviatoric stretch tensors $\hat{\boldsymbol{U}}$ while symsl is the counter part of Lagrangian deviatoric Hencky strain tensors $\hat{\boldsymbol{Y}} = \ln \hat{\boldsymbol{U}} = \mathrm{dev}\,(\ln \boldsymbol{U})$. The former is a nonlinear manifold while the latter is a linear subspace of $\mathbb{R}^{3\times 3}$. Thanks to the matrix exponential map, it is sufficient to sample the linear space of Hencky strains to obtain a relevant sampling of the stretch tensors. In addition, both sets are 5-dimensional, by virtue of symmetry and zero trace: any matrix Y in symsl typically writes with 5 independent components $(a,\, b,\, c,\, d,\, e) \in \mathbb{R}^5$ as

$$Y = \begin{pmatrix} a & b & c \\ b & d & e \\ c & e & -(a+d) \end{pmatrix};$$

the matrix $U = \exp Y$ in SymSL_+ follows by virtue of the exponential.

To sample the tangent space symsl, Kunc and Fritzen (2019a) proposed not to sample the components of the matrix in a regular grid of $\mathbb{R}^5$ but instead to sample the unique decomposition of the matrix on a basis of the space. An element Y of symsl is then expressed by its unique decomposition on the orthonormal basis $\mathcal{Y} = \left(Y^{(1)},\, Y^{(2)},\, Y^{(3)},\, Y^{(4)},\, Y^{(5)}\right)$ as

$$Y = \sum_{k=1}^{5} \alpha_k Y^{(k)}. \tag{IV.11}$$

with $\alpha = [\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5]^T \in \mathbb{R}^5$ linearly independent coordinates. Kunc and Fritzen (2019a) provide the following orthonormal basis $\mathcal{Y}$ ($Y^{(k)} : Y^{(l)} = \delta_{kl}$):

$$\begin{aligned} Y^{(1)} &= \frac{\sqrt{6}}{6}\begin{pmatrix} 2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \; Y^{(2)} = \frac{\sqrt{2}}{2}\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \\ Y^{(3)} &= \frac{\sqrt{2}}{2}\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \; Y^{(4)} = \frac{\sqrt{2}}{2}\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \; Y^{(5)} = \frac{\sqrt{2}}{2}\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \end{aligned} \tag{IV.12}$$

In addition, Kunc and Fritzen (2019a) proposed to split the coordinates $\{\alpha_k\}_{k=1}^5$ into an amplitude and directional part:

$$Y = \beta \sum_{k=1}^{5} a_k Y^{(k)} \tag{IV.13}$$

such that $\beta = 1/\|\alpha\| \in \mathbb{R}^+$ is the "deviatoric amplitude" and the unit vector $a = [a_1, a_2, a_3, a_4, a_5]^T \in \mathbb{R}^5$ ($\|a\| = 1$) is the "direction" of the matrix Y and hence the stretch tensor $\boldsymbol{U}$ (Kunc and Fritzen, 2019a)[1].

The set of stretch tensors $\mathcal{U}$ is then sampled with this method as follows (Kunc and Fritzen, 2019a):

[1] Here, we changed the notations: in Kunc and Fritzen (2019a), β is denoted t and a is denoted $\underline{N}$.

1. Place N_{det} determinants J regularly between the extremal values $J_{\min}$ and $J_{\max}$.

2. Generate an approximately uniform distribution of N_{dir} directions a in the unit sphere of $\mathbb{R}^5$.

3. Place N_{amp} deviatoric amplitude β regularly between 0 and the expected maximum value $\beta_{\max}$.

4. Return the set of matrix representations of $\boldsymbol{U}$:

$$\left\{ \left(J^{(i)}\right)^{1/3} \exp\left(\beta^{(j)} \sum_{k=1}^{5} \left[a^{(l)}\right]_k Y^{(k)}\right) \right\}_{i,j,l=1}^{i=N_{\text{det}},j=N_{\text{amp}},l=N_{\text{dir}}} \subset \mathcal{U}. \quad \text{(IV.14)}$$

All steps are very easy to perform, except for the isotropic distribution of directions in $\mathbb{R}^5$ where many possibilities exist, as emphasized by the authors. This topic is discussed later on.

To sum up, the method of Kunc and Fritzen (2019a) provides a very simple strategy to sample the space of admissible stretch tensors. It follows two main steps: (i) the DDMS allows for sampling the dilatational and the deviatoric parts of the tensors separately, (ii) the deviatoric stretch tensors are expressed with the matrix exponential of the decomposition of the corresponding Hencky strains on an orthonormal basis of a linear space.

To illustrate the method, we reproduce the two-dimensional example given in Kunc and Fritzen (2019a), which arises under *plane strain* assumption. Let us consider a BVP problem expressed in the canonical basis $\mathcal{B} = (\boldsymbol{e}_X, \boldsymbol{e}_Y, \boldsymbol{e}_Z)$ of $\mathbb{R}^3$. Let assume that $\boldsymbol{e}_Z$ is a plane strain direction. Then the matrix representation U of the stretch tensor $\boldsymbol{U}$ in the basis $\mathcal{B}$ takes the form

$$U = \begin{pmatrix} c & b & 0 \\ b & d & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{(IV.15)}$$

with c, b, $d \in \mathbb{R}$. The matrix can safely be reduced to its two-dimensional upper part $U_{(2)}$ such that the DDMS writes:

$$U_{(2)} = J^{1/2} \hat{U}_{(2)} \quad \text{(IV.16)}$$

where $J = \det U_{(2)} = cd - b^2$ and $\hat{U}_{(2)} = J^{-1/2} U_{(2)} \in \mathbb{R}^{2\times 2}$. The manifold of deviatoric stretch tensors and corresponding tangent space of deviatoric Hencky strains now respectively write

$$\text{SymSL}_+ = \left\{ U \in \mathbb{R}^{2\times 2} \mid U = U^T, \det(U) = 1, x^T U\, x > 0, \forall x \in \mathbb{R}^2 \right\}, \quad \text{(IV.17)}$$

$$\text{symsl} = \left\{ Y \in \mathbb{R}^{2\times 2} \mid Y = Y^T, \operatorname{tr}(Y) = 0 \right\}. \quad \text{(IV.18)}$$

An orthonormal basis of symsl is simply given as (Kunc and Fritzen, 2019a)

$$Y^{(1)}_{(2)} = \frac{\sqrt{2}}{2}\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}, Y^{(2)}_{(2)} = \frac{\sqrt{2}}{2}\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}. \tag{IV.19}$$

In Fig. IV.1, we show the c, b, d components of the matrices U (see Eq. (IV.15)) resulting from the sampling

$$U_{(2)} = J^{1/2}\exp\left(\beta\left(a_1 Y^{(1)}_{(2)} + a_2 Y^{(2)}_{(2)}\right)\right), \tag{IV.20}$$

where $(a_1, a_2) = (\cos\phi, \sin\phi)$ are obtained from an uniform sampling of $\phi \in [0, 2\pi)$, $\beta \in (0, 1]$ results from a regular sampling and J takes four equidistant values in $[0.1, 4]$. The manifolds spanned by each family stretch tensors of same determinant

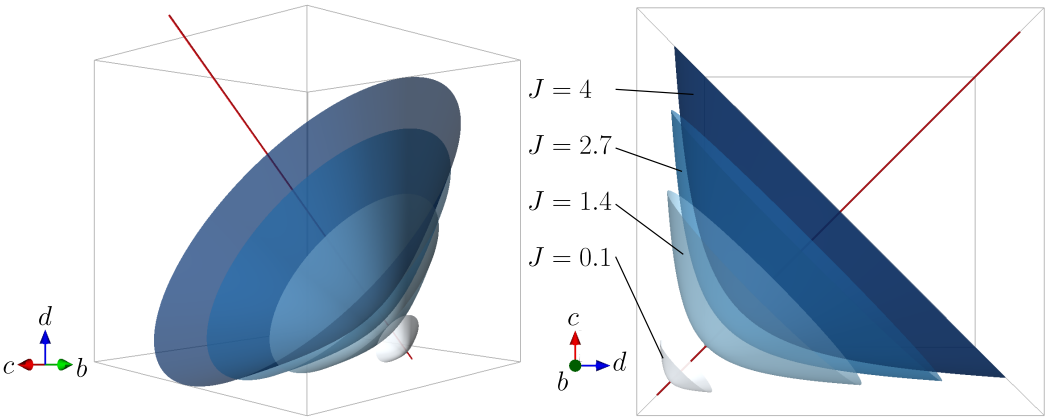


Figure IV.1 – Surfaces spanned by the c, b, d components of the two-dimensional stretch tensor $\boldsymbol{U}$ in the canonical basis (see Eq. (IV.15)), for four different sampling of the determinant $J \in \{0.1; 1.4; 2.7; 4\}$. We show two different perspectives of the same surfaces.

is a cone-like shape. It is then much smaller than the surrounding cube that would result from a regular grid sampling of the corresponding components. This illustrative example shows the great advantages of the sampling method proposed by Kunc and Fritzen (2019a). To further understand the sampling procedure, we plot in Fig. IV.2 the values of β and ϕ on a cone generated by the family of tensors of determinant $J = 1.4$. As can be seen from Fig. IV.2, the β parameter is indeed an amplitude factor: the further from the origin of the cone, the higher the value of β, the isovalues drawing ellipses on the cone. The parameter ϕ mapping the unit sphere in $\mathbb{R}^2$ to generate the a_1, a_2 coordinates is the angle of rotation of the cone.

2.1.2 Adaptation to two-dimensional, incompressible problems, under plane stress conditions

For the purpose of data-driven simulations conducted later in Section 3, we now adapt the dense sampling method to two-dimensional plane stress problems, for incompressible materials.

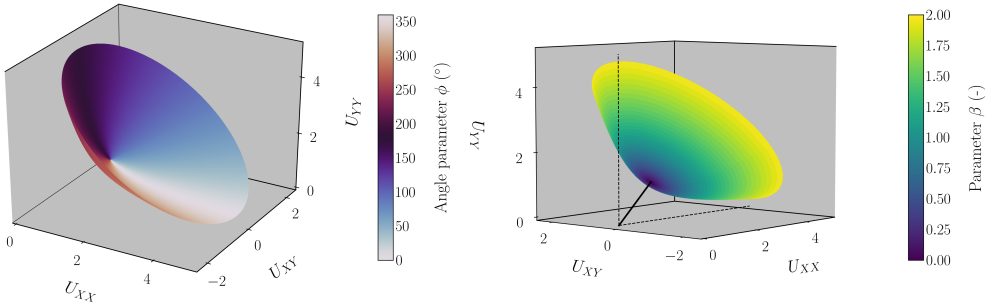


Figure IV.2 – Surface spanned by the c, b, d components of the two-dimensional stretch tensor $\boldsymbol{U}$ in the canonical basis (see Eq. (IV.15)), generated with $J = 1.4$ and a sampling of the amplitude β and the coordinates a_1 and a_2. Left: values of the angular parameter ϕ used to sample $a_1 = \cos\phi$ and $a_2 = \sin\phi$. Right: values of the amplitude parameter β.

First, the assumption of incompressibility reduces the sampling procedure to the deviatoric stretch tensors only: $J = \det \boldsymbol{U} = 1$, then $\hat{\boldsymbol{U}} = \boldsymbol{U}$ and $\hat{\boldsymbol{Y}} = \boldsymbol{Y}$. We then only need to sample the tangent space symsl of trace-less Hencky strains. To reduce the amount of notation, we do not write the $\hat{\cdot}$ decorators on the tensors and the matrices in this section. Second, the plane stress assumption together with the trace-less condition constrain the components of the Hencky strain tensors.

Let us again consider that the BVP problem is expressed in the canonical basis $\mathcal{B} = (\boldsymbol{e}_X, \boldsymbol{e}_Y, \boldsymbol{e}_Z)$ of $\mathbb{R}^3$ and assume that $\boldsymbol{e}_Z$ is the plane stress direction. Then, a Hencky strain tensor $\boldsymbol{Y}$ can be expressed with at-most three independent components. The matrix representation Y of $\boldsymbol{Y}$ in basis $\mathcal{B}$ then writes:

$$Y = [\boldsymbol{Y}]_{\mathcal{B}} = \begin{pmatrix} c & b & 0 \\ b & d & 0 \\ 0 & 0 & -(c+d) \end{pmatrix} \tag{IV.21}$$

with $c, b, d \in \mathbb{R}$. An orthonormal basis $\mathcal{Y}$ of the corresponding tangent space is given by the first three basis matrices of the general three-dimensional case, defined in Eq. (IV.12): $\mathcal{Y} = \left(Y^{(1)}, Y^{(2)}, Y^{(3)}\right)$. We remark that the basis matrices are also the representations of given tensors in basis $\mathcal{B}$. The set $\bar{\mathcal{U}}$ of practically admissible stretch tensors, in the case of incompressibility and plane stress conditions can then be sampled *via* a sampling of N_{amp} amplitudes $\left\{\beta^{(i)}\right\}$ and N_{dir} directions $\left\{a^{(j)}\right\}$ in $\mathbb{R}^3$ as

$$\left\{ \beta^{(i)} \exp\left(\sum_{k=1}^{3} \left[a^{(j)}\right]_k Y^{(k)} \right) \right\}_{i,\,j=1}^{i=N_{\text{amp}},\, j=N_{\text{dir}}} \subset \bar{\mathcal{U}}. \tag{IV.22}$$

In Fig. IV.3 we illustrate the set Eq. (IV.22) in the (U_{XX}, U_{XY}, U_{YY}) three-dimensional space, as previously, for four values of $\beta = 0.5$, 1, 1.5, 2 and a regular

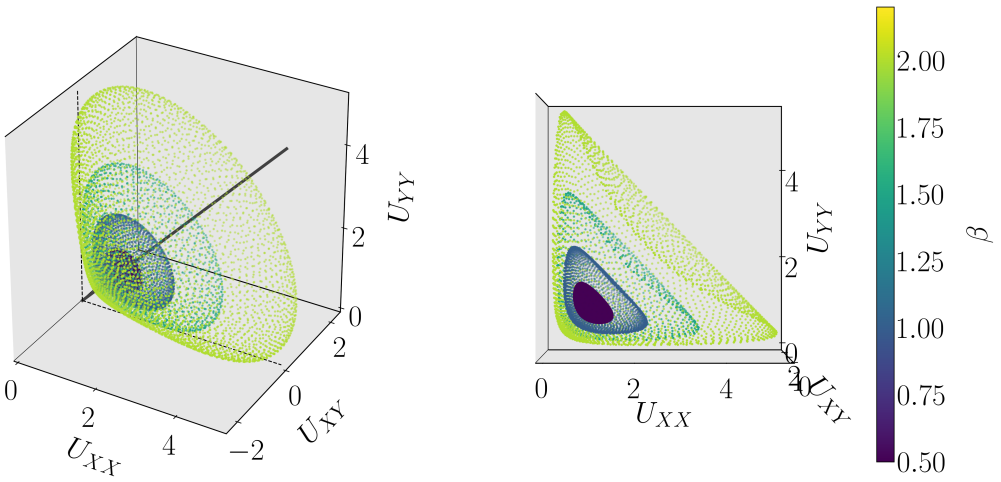


Figure IV.3 – Sampling of $\bar{\mathcal{U}}$ with constant $\beta \in \{0.5, 1, 1.5, 2\}$ for the two-dimensional case in plane stress conditions and incompressibility. Each scatter point represent a matrix generated by a pair (β, a) (see Eq. (IV.22)). Both plots give two different perspectives of the same set. In (a), the thick black line represents the set $\{\lambda \boldsymbol{I} \mid \lambda > 0\}$.In (b), only half of the surfaces are shown, such that $U_{XY} > 0$.

sampling of directions a on the unit sphere of $\mathbb{R}^3$. Unlike the plane strain case represented in Fig. IV.1, the three-dimensional surface spanned by the components of stretch tensors $\boldsymbol{U}$ is closed. It is still of rounded cone-like shape, with the top closing surface plane when $\beta = 1$, concave when $\beta < 1$ and convex when $\beta > 1$ (see Fig. IV.3(b)).

The procedure to generate DB-DENSE databases for two-dimensional, incompressible, plane stress data-driven problems is then given in Algorithm IV.1. Steps 2 and 3 of Algorithm IV.1 are straightforward. Let us give some details about Step 1.

The deviatoric amplitude β can either be regularly sampled in the interval $(0, \beta_{\max}]$ or adaptively sampled as suggested by Kunc and Fritzen (2019a), *e.g.* with a geometric progression to have more data at small strains than at high strains. In our databases, we adopt the latter method. For Step 1.2, "the generation of uniform point distributions on spheres is a research topic on its own" as pointed out by Kunc and Fritzen (2019a). In our case, to avoid the multiplication of sets necessary to compute several realizations of a random sampling, we want to use a *regular* sampling of directions in $\mathbb{R}^3$. The regular placement of points on a sphere is also a research topic on its own. For example, the numerical integration on the sphere has applications in geomechanics, see *e.g.* Hesse et al. (2010) dedicated chapter in the Handbook of geomathematics. In mechanics of materials, numerical integration schemes on the sphere are also needed by constitutive modeling. Three different methods have been evaluated in Verron (2015) with respect to their ability to com-

Algorithm IV.1 Generation of a DB-DENSE database.

Step 1. Generate a set of admissible three-dimensional deviatoric stretch tensors $\boldsymbol{U}$ ($\det(\boldsymbol{U}) = 1$), constrained by plane stress conditions along $\boldsymbol{e}_Z$, with the dense sampling method Eq. (IV.22).

1. Sample N_{amp} deviatoric amplitudes β in $(0, \beta_{\max})$.
2. Sample N_{dir} directions a in $\mathbb{R}^3$, such that $\|a\| = 1$.
3. Compute the linear combination of Hencky strains

$$\bar{Y}^{(j)} = \sum_{k=1}^{3} a_k^{(j)} Y^{(k)}, \ \forall j \in [1 \,..\, N_{\text{dir}}]. \tag{IV.23}$$

4. Return the set of samples

$$\bar{\mathcal{U}}_{N_{\text{amp}},\, N_{\text{dir}}} = \left\{ \beta^{(i)} \exp \bar{Y}^{(j)} \right\}_{i,\, j=1}^{i=N_{\text{amp}},\, j=N_{\text{dir}}} \subset \bar{\mathcal{U}}. \tag{IV.24}$$

Step 2. Generate the set of corresponding two-dimensional Green-Lagrange strain tensors from

$$\boldsymbol{E} = \tfrac{1}{2}\left(\boldsymbol{U}^2 - \boldsymbol{I}\right),$$

with $\boldsymbol{U} \in \mathbb{R}^{2\times 2}$ obtained from the $\boldsymbol{e}_X \otimes \boldsymbol{e}_X$, $\boldsymbol{e}_X \otimes \boldsymbol{e}_Y$, $\boldsymbol{e}_Y \otimes \boldsymbol{e}_Y$ components of $\boldsymbol{U}_{(3)} \in \bar{\mathcal{U}}_{N_{\text{amp}},\, N_{\text{dir}}}$ ($\boldsymbol{U}_{(3)} = \boldsymbol{U}_{(3)}^T$).

Step 3. Generate the set of corresponding two-dimensional second Piola-Kirchhoff stress tensors $\boldsymbol{S} = \boldsymbol{S}(\boldsymbol{U}^2)$ from the incompressible neo-Hookean relation Eq. (IV.3).

Return the material database

$$\mathcal{D}_{\text{loc}} = \left\{ \left(\boldsymbol{E}^{(i,j)}, \boldsymbol{S}^{(i,j)} \right) \mid i \in [1 \,..\, N_{\text{amp}}],\, j \in [1, .. \, N_{\text{dir}}] \right\} \subset \mathbb{R}^{2\times 2}_{\text{sym}} \times \mathbb{R}^{2\times 2}_{\text{sym}}.$$

pute the principal strain invariants in large strain: the classical Gaussian scheme of Bažant and Oh (1986) developed for the computation of so-called microplane models for the constitutive modeling of concrete and rock; the geometrical meshing approach proposed by Badel and Leblond (2004) to overcome the limitations of the previous method; the mathematical method of Sloan and Womersley (2004) consisting in extremal systems of points on the unit sphere, widely used in different fields such as geoscience or wave propagation, for the interpolation of polynomials on the sphere. Based on the findings of Verron (2015), we use in our databases the method of Sloan and Womersley (2004), which proved its efficiency with the fewest number of points. Another reason for this choice arises from one of the geometric properties

of the sets generated by this method, as briefly explained in the following.

The minimal geodesic distance between two points in the generated set decreases as π/n, with n the highest degree of the (spherical) polynomial interpolation ($\pi/(2n)$ being the theoretical value for extremal systems (Sloan and Womersley, 2004)). The number of points on the sphere is then $d = (n+1)^2$ (Sloan and Womersley, 2004). The geodesic distance between two points x and y on the unit sphere in $\mathbb{R}^3$ is $\mathrm{dist}(x, y) = \cos^{-1}(x^T y)$. The method used by Kunc and Fritzen (2019a) to generate uniform distributions on the sphere was introduced in a previous work (Fritzen and Kunc, 2018b) which also focused on minimizing the geodesic distance between two points. We shall not give further details on the approach here, but the interested reader is referred to the paper Sloan and Womersley (2004) or the chapter Hesse et al. (2010).

A large number of sets of points obtained by the approach are freely available online at Womersley (2007) and were simply downloaded to generate our DB-DENSE databases. In particular, the samples of stretch tensors represented in Fig. IV.3 have been generated with those sets. In Fig. IV.4 we plot the resulting sets of points on the unit sphere for $N_{\mathrm{dir}} = 25$, 256 and 2500, as an illustration of the method.

2.1.3 Mechanical analysis of the dense sampling method

The dense sampling method, as introduced by Kunc and Fritzen (2019a), adapted to the two-dimensional case arising under plane stress and incompressible conditions with the approach of Sloan and Womersley (2004) provides a very efficient and clean way to sample the space of admissible stretch tensors. However, while the sampling is dense, *i.e.* any neighborhood of any stretch tensor $\boldsymbol{U}$ contains at least one point in the sampling set, the mechanical relevance of the sampling set is not clear. In particular, it can be of interest to know which modes and amplitudes of deformation are spanned by the sampling set, in order to tailor the database to specific needs.

To answer these questions, we build upon the "invariant basis for natural strain" developed by Criscione et al. (2000). In their work, Criscione et al. (2000) introduced three physically meaningful invariants of the Eulerian Hencky strain tensor $\boldsymbol{H} = \ln \boldsymbol{V}$. The definitions of these invariants also apply to the Lagrangian Hencky strain tensor $\boldsymbol{Y} = \ln \boldsymbol{U}$ as $\boldsymbol{U}$ is merely a rigid body rotation of $\boldsymbol{V}$. We then reformulate the invariant basis theory of Criscione et al. (2000) for the Lagrangian Hencky strain tensor $\boldsymbol{Y}$:

- the first invariant K_1 describes the "amount-of-dilatation" and is defined as

$$K_1 = \mathrm{tr}\, \boldsymbol{Y} = \ln J. \tag{IV.25}$$

- the second invariant K_2 is a measure of the "magnitude-of-distortion" and is defined as

$$K_2 = \sqrt{\mathrm{dev}\, \boldsymbol{Y} : \mathrm{dev}\, \boldsymbol{Y}}; \tag{IV.26}$$

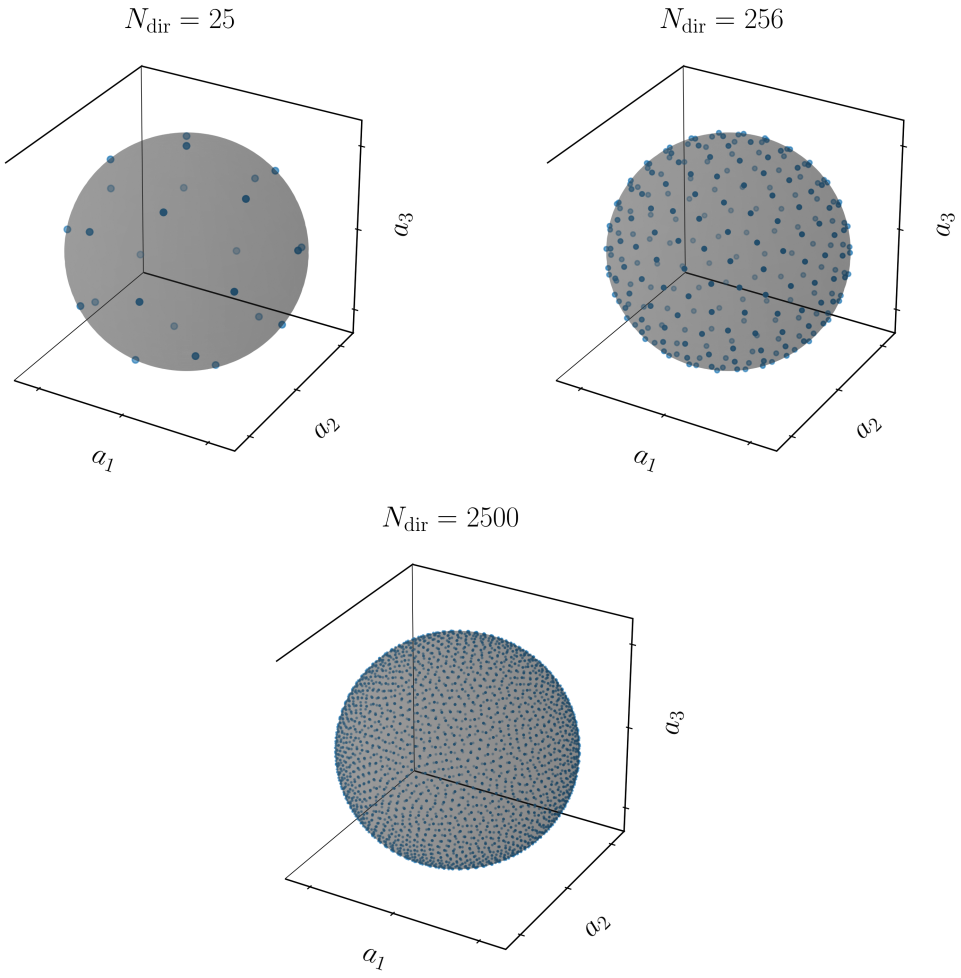


Figure IV.4 – Set of N_{dir} points $(\alpha_1, \alpha_2, \alpha_3)$ on the unit sphere of $\mathbb{R}^3$, generated with the method of Sloan and Womersley (2004), downloaded from Womersley (2007).

- the third invariant K_3 describes the "mode-of-distortion" and is defined as

$$K_3 = \frac{3\sqrt{6}}{(K_2)^3} \det(\operatorname{dev} \boldsymbol{Y}), \tag{IV.27}$$

  where the constant $3\sqrt{6}$ was chosen so that $K_3 \in [-1, 1]$ (Criscione et al., 2000).

Invariant $K_1 \in (-\infty, \infty)$ gives both magnitude and sign of dilatation, while $K_2 \in [0, \infty)$ only gives an absolute magnitude of the distortion. Remarkable values of the three invariants are given Table IV.1. For any Hencky strain tensor $\boldsymbol{Y}$, the triplet (K_1, K_2, K_3) gives unambiguous and physical information on the deformation. In the following, we then analyze the dense sampling method with these tools to give insight on the mechanical signification of the database.

Table IV.1 – Remarkable values of the three invariants of $\boldsymbol{Y}$ and their physical meaning.

Invariant	Value	Description
K_1	0	Isochoric deformation (or incompressible material)
	< 0	contraction
	> 0	dilatation
K_2	0	no distortion, the deformation is a pure dilatation: $\boldsymbol{U} = \lambda \boldsymbol{I}$, or equivalently $\boldsymbol{Y} = \ln(\lambda)\, \boldsymbol{I}$ with $\lambda > 0$.
K_3	-1	equi-biaxial tension (or uniaxial compression)
	0	pure shear
	1	uniaxial tension (or equi-biaxial compression)

Original three-dimensional method Let us consider a stretch tensor $\boldsymbol{U}$ obtained from the sampling method of Kunc and Fritzen (2019a) recalled in Eq. (IV.14). Let denote $\boldsymbol{Y}^{(k)}$ the basis deviatoric Hencky strain tensors with matrix representations given by Eq. (IV.12). Then, there is a pair $(\beta, a) \in \mathbb{R}_+ \times \mathbb{R}^5$, with $\|a\| = 1$ such that

$$\boldsymbol{U} = J^{1/3} \exp\left(\beta \sum_{k=1}^{5} a_k \boldsymbol{Y}^{(k)}\right), \tag{IV.28}$$

with $J = \det \boldsymbol{U}$. The corresponding Hencky strain tensor $\boldsymbol{Y} = \ln \boldsymbol{U}$ then writes

$$\boldsymbol{Y} = \frac{1}{3} \ln(J) \boldsymbol{I} + \beta \sum_{k=1}^{5} a_k \boldsymbol{Y}^{(k)}. \tag{IV.29}$$

Let us now compute the invariants K_1, K_2, K_3 of $\boldsymbol{Y}$.

- First, we retrieve, as expected, that the first invariant K_1 is the natural logarithm of $J = \det \boldsymbol{U}$:

$$K_1 = \operatorname{tr} \boldsymbol{Y} = \frac{1}{3} \ln J \operatorname{tr} \boldsymbol{I} + \beta \sum_{k=1}^{5} a_k \operatorname{tr} \boldsymbol{Y}^{(k)} = \ln J. \tag{IV.30}$$

- Second, to compute K_2, let us first write the deviatoric part of $\boldsymbol{Y}$. Recalling that (i) the deviatoric operator is linear, (ii) $\operatorname{dev} \boldsymbol{I} = \boldsymbol{0}$ and, (iii) $\operatorname{dev} \boldsymbol{Y}^{(k)} = \boldsymbol{Y}^{(k)}$, $\forall k$, we have:

$$\operatorname{dev} \boldsymbol{Y} = \frac{1}{3} \ln(J) \operatorname{dev} \boldsymbol{I} + \beta \sum_{k=1}^{5} a_k \operatorname{dev} \boldsymbol{Y}^{(k)} = \beta \sum_{k=1}^{5} a_k \boldsymbol{Y}^{(k)}. \tag{IV.31}$$

Then, recalling that the basis of $\boldsymbol{Y}^{(k)}$ is orthonormal, *i.e.* $\boldsymbol{Y}^{(k)} : \boldsymbol{Y}^{(l)} = \delta_{kl}$, and that the vectors $a^{(k)}$ are unit vectors, *i.e.* $\|a^{(k)}\|^2 = 1$, the invariant K_2 is equal to the "deviatoric amplitude" β:

$$\begin{aligned}\operatorname{dev}\boldsymbol{Y} : \operatorname{dev}\boldsymbol{Y} &= \left(\beta\sum_{k=1}^{5} a_k\boldsymbol{Y}^{(k)}\right) : \left(\beta\sum_{l=1}^{5} a_l\boldsymbol{Y}^{(l)}\right)\\ &= \beta^2\sum_{k=1}^{5}\sum_{l=1}^{5} a_k a_l \boldsymbol{Y}^{(k)} : \boldsymbol{Y}^{(l)} = \beta^2\sum_{k=1}^{5}\sum_{l=1}^{5} a_k a_l \delta_{kl} \qquad \text{(IV.32)}\\ &= \beta^2\sum_{k=1}^{5}\|a^{(k)}\|^2 = \beta^2,\end{aligned}$$

and hence,

$$K_2 = \beta. \qquad \text{(IV.33)}$$

- Third, the invariant K_3 writes:

$$K_3 = \frac{3\sqrt{6}}{(K_2)^3}\det(\operatorname{dev}\boldsymbol{Y}) = 3\sqrt{6}\det\sum_{k=1}^{5} a_k\boldsymbol{Y}^{(k)}. \qquad \text{(IV.34)}$$

Finally, the dense sampling method can be re-written in terms of the natural strain invariant as

$$\boldsymbol{U} = \exp\left(\frac{1}{3}K_1\boldsymbol{I} + K_2\sum_{k=1}^{5} a_k\boldsymbol{Y}^{(k)}\right). \qquad \text{(IV.35)}$$

In particular, note that the "deviatoric amplitude" β introduced by Kunc and Fritzen (2019a) is indeed a measure of magnitude. In addition, the invariant K_3 only depends on the direction vector a. A sampling of the sphere in $\mathbb{R}^5$ is then equivalent to a sampling of modes-of-distortion, measured by K_3. This correspondence between the two theories is, in fact, not surprising. Indeed, both were build upon the DDMS, either applied to the stretch tensor $\boldsymbol{U}$ (Kunc and Fritzen, 2019a), or directly to the Hencky strain tensor $\boldsymbol{Y}$ (Criscione et al., 2000). Our analysis thus bridges the gap between them and proves that the dense sampling method is split into three sampling of physically meaningful distinct quantities K_1, K_2 and K_3.

Adapted method for two-dimensional incompressible plane stress problems In the case of two-dimensional incompressible plane stress problems, we can deepen the analysis to extract a one-to-one relationship between a, K_3 and other parameters to be determined.

Let us again consider a BVP expressed in the canonical basis $\mathcal{B} = (\boldsymbol{e}_X, \boldsymbol{e}_Y, \boldsymbol{e}_Z)$. Let us apply Eq. (IV.35) to a three-dimensional stretch tensor $\boldsymbol{U}$, with determinant $J = 1$ (incompressibility assumption). In addition, let $\boldsymbol{e}_Z$ be the plane stress direction. As mentioned in Section 2.1.2, the tangent space of deviatoric stretch tensors

is only three-dimensional. Then, only the first three basis tensors $\boldsymbol{Y}^{(k)}$ are needed. Then, for plane stress incompressible problems Eq. (IV.35) becomes

$$\boldsymbol{Y} = \ln \boldsymbol{U} = K_2 \sum_{k=1}^{3} a_k \boldsymbol{Y}^{(k)}, \tag{IV.36}$$

and

$$\boldsymbol{U} = \exp\left(K_2 \sum_{k=1}^{3} a_k \boldsymbol{Y}^{(k)} \right). \tag{IV.37}$$

In the following, we relate coefficients a_1, a_2, a_3 to invariant K_3 using the matrix representations of tensor $\boldsymbol{Y}$ in the canonical and principal basis.

Let us first recall the matrix representation $Y^{(k)}$ of the basis tensors $\boldsymbol{Y}^{(k)}$ in the basis $\mathcal{B}$ for $k = 1$, 2 and 3:

$$\begin{aligned} Y^{(1)} &= \frac{\sqrt{6}}{6} \begin{pmatrix} 2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \; Y^{(2)} = \frac{\sqrt{2}}{2} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \\ Y^{(3)} &= \frac{\sqrt{2}}{2} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \end{aligned} \tag{IV.38}$$

Note that, based on a quick calculation and the remarkable value of the invariants given in Table IV.1, the basis matrix $Y^{(1)}$ corresponds to a uniaxial tension along $\boldsymbol{e}_X$, $Y^{(2)}$ to a pure shear test along $\boldsymbol{e}_Y$ and $Y^{(3)}$ to a pure shear test along the direction at 45° of $\boldsymbol{e}_X$. The matrix representation $Y_{\mathcal{B}}$ of the tensor $\boldsymbol{Y}$ in the basis $\mathcal{B}$ then writes:

$$Y_{\mathcal{B}} = K_2 \frac{\sqrt{2}}{2} \begin{pmatrix} \frac{2\sqrt{2}}{3} a_1 & a_3 & 0 \\ a_3 & -\frac{\sqrt{3}}{3} a_1 + a_2 & 0 \\ 0 & 0 & -\frac{\sqrt{3}}{3} a_1 - a_2 \end{pmatrix} \tag{IV.39}$$

To relate the coefficient a_1, a_2, a_3 to invariant K_3, we build upon the relationships between the eigenvalues y_I, y_{II}, y_{III} of $\boldsymbol{Y}$ and K_3 given by Criscione et al. (2000) as

$$y_I = \ln \lambda_I = K_2 \sqrt{\frac{2}{3}} \sin\left(\varphi + \frac{2\pi}{3} \right) \tag{IV.40a}$$

$$y_{II} = \ln \lambda_{II} = K_2 \sqrt{\frac{2}{3}} \sin\left(\varphi \right) \tag{IV.40b}$$

$$y_{III} = \ln \lambda_{III} = K_2 \sqrt{\frac{2}{3}} \sin\left(\varphi - \frac{2\pi}{3} \right), \tag{IV.40c}$$

with $\lambda_I \geq \lambda_{II} \geq \lambda_{III}$ the principal stretches, and $\varphi = -\arcsin(K_3)/3 \in [-\pi/6, \pi/6]$. We refer to $\boldsymbol{p}_I, \boldsymbol{p}_{II}, \boldsymbol{p}_{III}$ as the principal directions (eigenvectors) associated with the corresponding principal stretches. These eigenvalues are ordered in

decreasing order, but this order is not necessary in accordance with the plane stress condition: there is no reason for y_{III} to be the eigenvalue associated to the eigenvector $\boldsymbol{e}_Z$. As an example, an equi-biaxial compression in the plane (X, Y) would result in the highest eigenvalue y_I being in the $\boldsymbol{e}_Z$ direction. In addition, note that all eigenvalues are the product of K_2 with a function of K_3. Let us then define a generic form for the relationship between the eigenvalues of $\boldsymbol{Y}$ and the invariants K_2 and K_3 as

$$y(K_2, K_3, z) = K_2 \sqrt{\frac{2}{3}} \sin\left(\varphi + z\frac{2\pi}{3}\right), \tag{IV.41}$$

where $z \in \mathbb{Z}$ is an integer taking only three distinct values -1, 0 and 1 which defines the order of the eigenvalue $y_i = y(K_2, K_3, z_i)$ ($z = 1$ is the highest, $z = 0$ is the middle one, $z = -1$ is the lowest).

We denote $\mathcal{P} = (\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3)$ the principal basis, where $\boldsymbol{p}_3 = \boldsymbol{e}_Z$ is the principal direction of plane stress, and $\boldsymbol{p}_1$, $\boldsymbol{p}_2$ are the remaining principal directions such that $\boldsymbol{p}_1$ is associated with the highest eigenvalue y_1 in the (X, Y) plane: $y_1 = \ln\lambda_1 \geq y_2 = \ln\lambda_2$ and $y_3 = \ln\lambda_3$ is the principal Hencky strain along $\boldsymbol{e}_z$. The matrix representation $Y_{\mathcal{P}}$ of $\boldsymbol{Y}$ in the principal basis $\mathcal{P}$ then writes

$$Y_{\mathcal{P}} = y_1\,(\boldsymbol{p}_1 \otimes \boldsymbol{p}_1) + y_2\,(\boldsymbol{p}_2 \otimes \boldsymbol{p}_2) + y_3\,(\boldsymbol{p}_3 \otimes \boldsymbol{p}_3)\,. \tag{IV.42}$$

Note that $(\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3)$ is a permutation of $(\boldsymbol{p}_I, \boldsymbol{p}_{II}, \boldsymbol{p}_{III})$. The eigenvalue y_i with $i = 1, 2, 3$ can be obtained from Eq. (IV.41) with a specific value of $z = z_i \in \{-1, 0, 1\}$. In addition, we denote $\theta = (\boldsymbol{p}_1, \boldsymbol{e}_X)$ the angle between $\boldsymbol{p}_1$ and $\boldsymbol{e}_X$, *i.e.* the angle of principal stretch in the (X, Y) plane. From here on out, we refer to θ as the *orientation angle* of the in-plane principal stretch. Then, the matrix representation $Y_{\mathcal{B}}$ of $\boldsymbol{Y}$ in the canonical basis $\mathcal{B}$ is related to its counter part $Y_{\mathcal{P}}$ as

$$Y_{\mathcal{B}} = Q\,Y_{\mathcal{P}}\,Q^{-1} \qquad \text{with } Q = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 01 & \end{pmatrix}. \tag{IV.43}$$

From Eqs. (IV.39), (IV.41) and (IV.43) it is now possible to relate the coefficients a_k with the invariant K_3 and the indicator z_1, z_2 of the order of the eigenvalues y_1, y_2 as

$$a_1 = \cos^2(\theta)\sin(\varphi + z_1\tfrac{2\pi}{3}) + \sin^2(\theta)\sin(\varphi + z_2\tfrac{2\pi}{3}) \tag{IV.44a}$$

$$a_2 = \frac{\sqrt{3}}{3}\left((1 + \sin^2(\theta))\sin(\varphi + z_1\tfrac{2\pi}{3}) + (1 + \cos^2(\theta))\sin(\varphi + z_2\tfrac{2\pi}{3})\right) \tag{IV.44b}$$

$$a_3 = \frac{\sqrt{3}}{3}\left(\sin(\varphi + z_1\tfrac{2\pi}{3}) - \sin(\varphi + z_2\tfrac{2\pi}{3})\right)\sin(2\theta), \tag{IV.44c}$$

with

$$\varphi = -\arcsin(K_3)/3. \tag{IV.44d}$$

The different parameters of the sampling method for plane stress incompressible problems are then all mechanically meaningful: (i) the deviatoric amplitude β is the magnitude-of-distortion K_2, and (ii) the direction $a \in \mathbb{R}^3$ is related to the mode-of-distortion K_3, the orientation angle θ and the orders (z_1, z_2) of the in-plane principal stretches.

It is then also possible to sample the space of admissible stretch tensors directly from K_2, K_3, θ, z rather than from K_2, a_1, a_2, a_3. That way, the modes-of-distortion and the orientation angle of the largest in-plane principal stretch can be tailored to specific needs. In Fig. IV.5, we plot an example of such a set of stretch tensors, with $K_2 = 1$ fixed, K_3 linearly spaced between $[-1, 1]$ and $(z_1, z_2) \in \{(1, 0), (0, -1), (1, -1)\}$. We can see in Fig. IV.5(right) that the cone-like

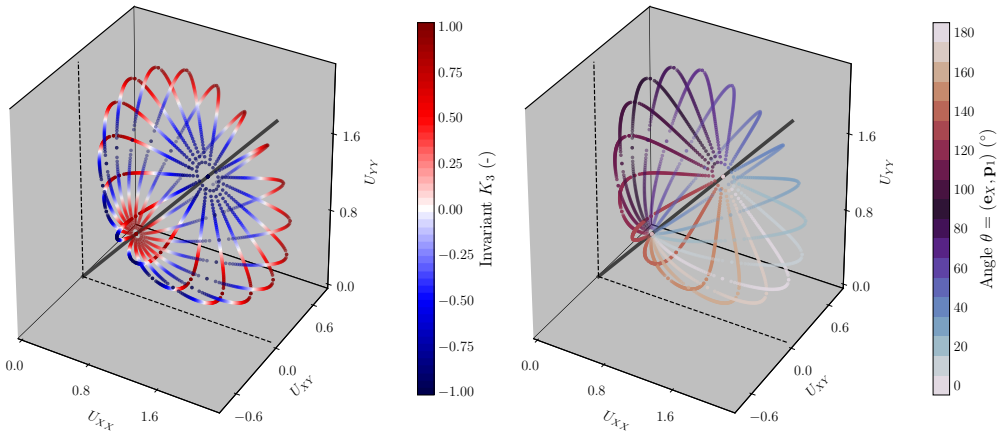


Figure IV.5 – Stretch tensors $\boldsymbol{U}$ in basis $\mathcal{B} = (\boldsymbol{e}_X, \boldsymbol{e}_Y, \boldsymbol{e}_Z)$ obtained from a regular sampling of modes-of-distortion K_3 and of orientation angle θ with steps of $\Delta K_3 = 0.04$ and $\Delta\theta = 10°$.

surface is engendered by a rotation of angle 2θ of a parametric three-dimensional curve along the axis $U_{YY} = U_{XX}$ (which corresponds to tensors $\boldsymbol{U} = \lambda \boldsymbol{I}$, with $\lambda > 0$).

Finally, following a similar rationale for the stretch tensor, one can relate the components U_{XX}, U_{XY}, U_{ZZ} to the parameters K_2, K_3, θ and (z_1, z_2). We make use of the properties of the matrix exponential for diagionalizable matrices. Hence, the matrix representation $U_{\mathcal{B}}$ of tensor $\boldsymbol{U} = \ln \boldsymbol{Y}$ in the canonical basis writes

$$U_{\mathcal{B}} = Q \exp Y_{\mathcal{P}} \, Q^{-1} \tag{IV.45}$$

with $Y_{\mathcal{P}}$ defined in Eq. (IV.42). The canonical components readily follow as

$$U_{XX} = \tfrac{1}{2} \left(\lambda_1 + \lambda_2 + (\lambda_1 - \lambda_2) \cos(2\theta)\right) \tag{IV.46a}$$
$$U_{YY} = \tfrac{1}{2} \left(\lambda_1 + \lambda_2 + (\lambda_2 - \lambda_1) \cos(2\theta)\right) \tag{IV.46b}$$
$$U_{XY} = \tfrac{1}{2} \left(\lambda_1 - \lambda_2\right) \sin(2\theta), \tag{IV.46c}$$

with

$$\lambda_1 = e^{y_1} = y\left(K_2, K_3, \theta, z_1\right) \tag{IV.47a}$$
$$\lambda_2 = e^{y_2} = y\left(K_2, K_3, \theta, z_2\right) \tag{IV.47b}$$

and the functional y as in Eq. (IV.41). Thanks to Eq. (IV.46), the parametric three-dimensional rotation surface spanned by the components of the stretch tensors $\boldsymbol{U}$ is fully determined with mechanically meaningful parameters K_3, θ, and (z_1, z_2). In Fig. IV.6 we illustrate the role of (z_1, z_2): the different parts of the cone-like shape are obtained from three combinations of $(z_1, z_2) \in \{(1,0),(0,-1),(1,-1)\}$. In Fig. IV.7, we plot the arrangement of the K_3 and θ isovalues on the cone-like shape. The K_3 isovalues draw sections of the cones, perpendicular to the axis $U_{XX} = U_{YY}$ while θ isovalues illustrate again the surface of revolution.

With this analysis, we now know that a direction vector a on the unit sphere of $\mathbb{R}^3$ corresponds to a unique mode-of-distortion K_3, occurring along an orientation angle θ, in tension or compression depending on the value of (z_1, z_2). Having a dense, regular placement of direction vectors is then equivalent to sampling various modes-of-distortion in different directions and of different "signs" (tension or compression). This variety of deformation is what we refer to as the *richness* of the database. It then appears to us that the dense sampling method is a convenient way of generating and designing *rich* material databases for DDCM.

2.2 Standard sampling

From the insight of the mechanical analysis conducted in the previous section, it is also possible to generate a material database which contains only selected modes-of-distortion. The hyperelastic constitutive models are generally calibrated with at-most three homogeneous experimental tests: a uniaxial tensile, a pure shear, and when possible an equi-biaxial test (*e.g.* Ogden (1972); Treloar (1944)). They correspond respectively to $K_3 = 1$, $K_3 = 0$ and $K_3 = -1$. With the method described above, it becomes very easy to generate a database containing these modes of deformation only, in every directions of the plane, for various amplitudes. We refer in the following to this kind of material databases as standard material database (DB-STD). With these databases, we aim at evaluating whether the data-driven computing paradigm could be used with the currently available experimental data. The procedure used to generate a given DB-STD database is listed in Algorithm IV.2.

2.3 Importance sampling

As aforementioned, the finite strain DDI method is not able to generate Lagrangian material data points yet. However, we wanted to conduct a similar analysis as the one presented in Stainier et al. (2019), where the authors compared a regular sampling of the strain space to an "importance sampling". The latter term was used to refer to the sampling generated by DDI. In this approach, the material

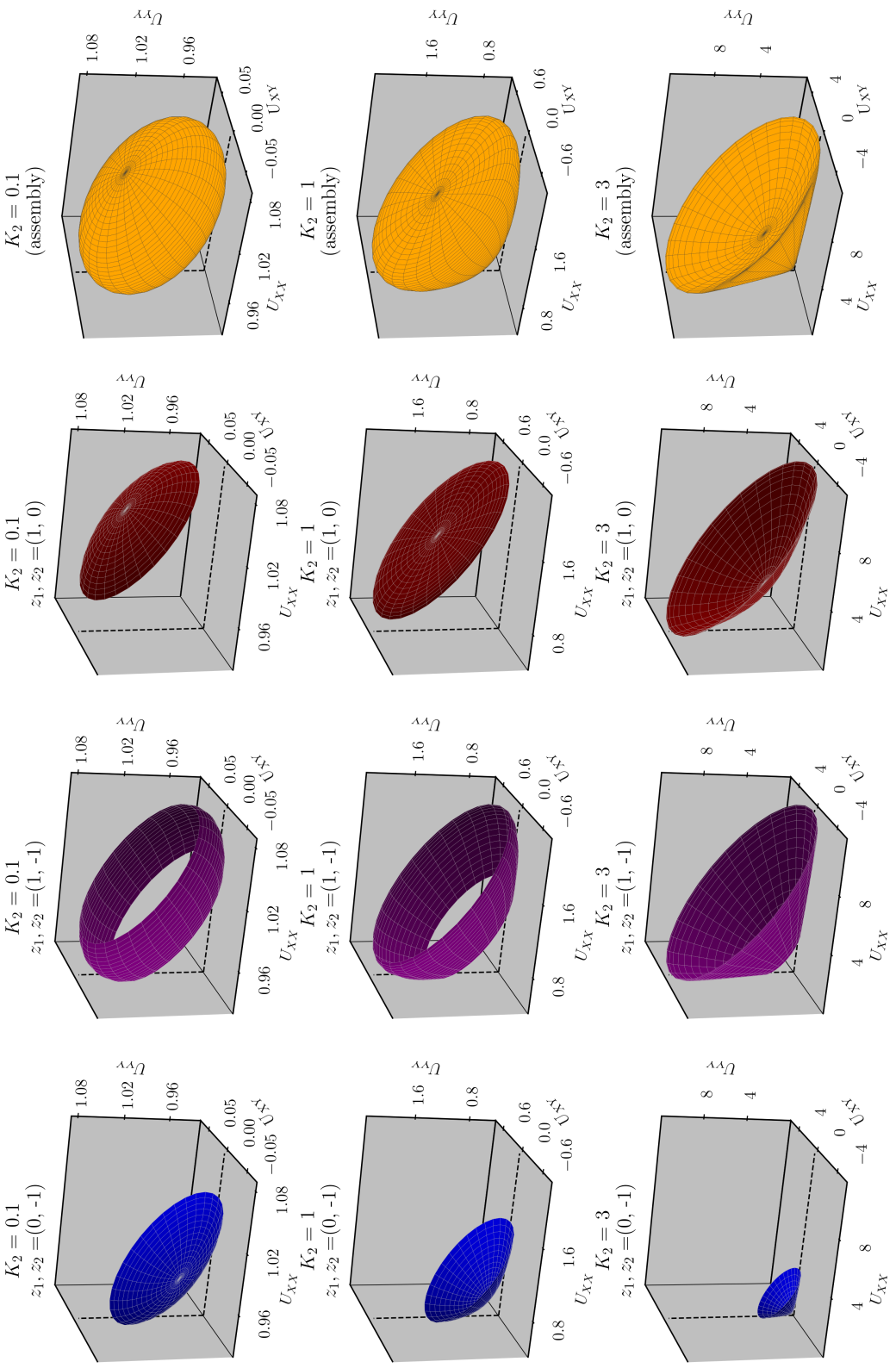


Figure IV.6 – Assembly of the cone-like shapes into a parametric surface from different orders of the in-plane principal stretches, parameterized by the integers z_1, z_2 (see Eqs. (IV.41) and (IV.46)).

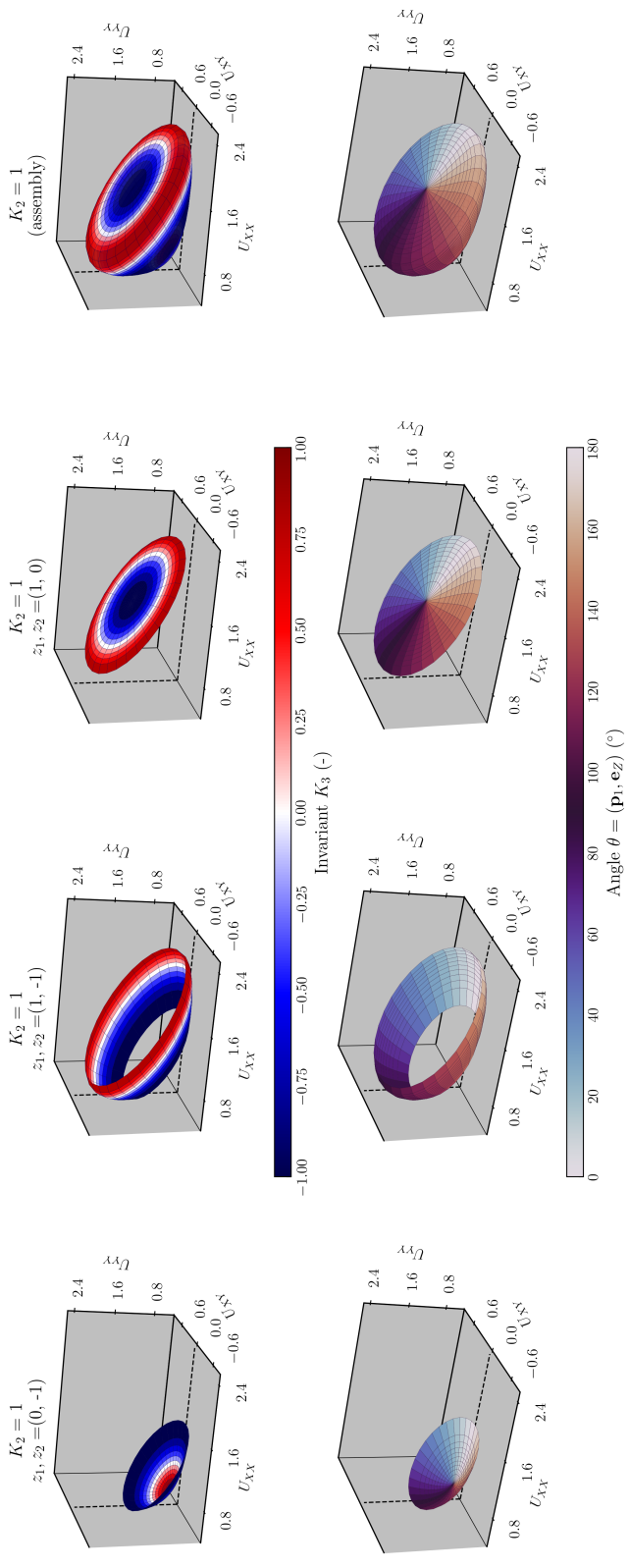


Figure IV.7 – Values of the mode-of-distortion K_3 and the principal stretch direction $\theta = (\boldsymbol{e}_X, \boldsymbol{p}_1)$ on the space spanned by the components of the stretch tensors (see Eqs. (IV.41) and (IV.46)). The set was generated with an amplitude-of-distortion $K_2 = 1$.

Algorithm IV.2 Generation of a DB-STD database.

Step 1. Generate a set of admissible three-dimensional deviatoric stretch tensors $\boldsymbol{U}$ ($\det(\boldsymbol{U}) = 1$), constrained by plane stress condition along $\boldsymbol{e}_Z$, with the standard sampling method.

1. Sample N_{amp} deviatoric amplitudes $\beta = K_2$ in $(0, \beta_{\max}]$.
2. Sample N_θ orientation angles θ in $[0, \pi)$.
3. Select $N_{K_3} = 3$ modes-of-distortion K_3 in $\{-1, 0, 1\}$.
4. Set the couples of order indicators (z_1, z_2) to $\{(1,0), (0,-1), (1,-1)\}$.
5. Compute the set of $N_{\text{dir}} = 3 \times N_{K_3} \times N_{theta} \times N_{\text{amp}}$ direction vectors $a = [a_1, a_2, a_3]^T \in \mathbb{R}^3$ with the relations $a_k(K_2, K_3, \theta, z_1, z_2)$ given by Eq. (IV.44).
6. Compute the corresponding set of linear combinations of Hencky strains according to Eq. (IV.23).
7. Return the set of samples

$$\bar{\mathcal{U}}_{N_{\text{amp}},\,N_{\text{dir}}} = \left\{ \beta^{(i)} \exp \bar{Y}^{(j)} \right\}_{i,\,j=1}^{i=N_{\text{amp}},\,j=N_{\text{dir}}} \subset \bar{\mathcal{U}}.$$

Step 2. Generate the set of corresponding two-dimensional Green-Lagrange strain tensors from

$$\boldsymbol{E} = \tfrac{1}{2}\left(\boldsymbol{U}^2 - \boldsymbol{I}\right),$$

with $\boldsymbol{U} \in \mathbb{R}^{2\times 2}$ obtained from the $\boldsymbol{e}_X \otimes \boldsymbol{e}_X$, $\boldsymbol{e}_X \otimes \boldsymbol{e}_Y$, $\boldsymbol{e}_Y \otimes \boldsymbol{e}_Y$ components of $\boldsymbol{U}_{(3)} \in \bar{\mathcal{U}}_{N_{\text{amp}},\,N_{\text{dir}}}$ ($\boldsymbol{U}_{(3)} = \boldsymbol{U}_{(3)}^T$).

Step 3. Generate the set of corresponding two-dimensional second Piola-Kirchhoff stress tensors $\boldsymbol{S} = \check{\boldsymbol{S}}(\boldsymbol{U}^2)$ from the incompressible neo-Hookean relation Eq. (IV.3).

Return the material database

$$\mathcal{D} = \left\{ \left(\boldsymbol{E}^{(i,j)}, \boldsymbol{S}^{(i,j)} \right) \mid i \in [1\,..\,N_{\text{amp}}],\, j \in [1,..\,N_{\text{dir}}] \right\} \subset \mathbb{R}^{2\times 2}_{\text{sym}} \times \mathbb{R}^{2\times 2}_{\text{sym}}$$

states are the centroids of clusters of mechanical states, obtained in that case from FE simulations of various loading conditions and loading amplitudes on a complex structure.

To emulate a database that would have been generated by the DDI, we performed a clustering analysis on the mechanical states obtained from various computations on a thin membrane with two holes. In the following, this kind of material databases is referred to as DDI material database (DB-DDI). The constitutive model used in

the FE simulations is the neo-Hookean incompressible model given in Eq. (IV.3). The mesh of the structures and some snapshots of the loading conditions are given in Fig. IV.8.

To perform the clustering, the $(\boldsymbol{E}, \boldsymbol{S})$ pairs are transformed by the Euclidean mapping mentioned in Chapter III and presented in Appendix B, into a single vector so that the standard Euclidean product is equivalent to the data-driven square distance. We recall that the data-driven square distance d^2 writes

$$d^2\left((\boldsymbol{E}, \boldsymbol{S}), (\boldsymbol{E}', \boldsymbol{S}')\right) = \frac{1}{2}\left(\Delta\boldsymbol{E} : \mathbb{C} : \Delta\boldsymbol{E} + \Delta\boldsymbol{S} : \mathbb{C}^{-1} : \Delta\boldsymbol{S}\right), \tag{IV.48}$$

with $\Delta\boldsymbol{E} = \boldsymbol{E} - \boldsymbol{E}'$ and $\Delta\boldsymbol{S} = \boldsymbol{S} - \boldsymbol{S}'$. Recasting the tensors into their Mandel form (Brannon, 2018), it also writes as the standard inner product:

$$d^2\left((\boldsymbol{E}, \boldsymbol{S}), (\boldsymbol{E}', \boldsymbol{S}')\right) = \left(\mathbf{z} - \mathbf{z}'\right) \cdot \left(\mathbf{z} - \mathbf{z}'\right), \tag{IV.49}$$

with

$$\mathbf{z} = \begin{pmatrix} \frac{1}{\sqrt{2}}\mathbf{C}^{1/2}\mathbf{E} \\ \frac{1}{\sqrt{2}}\mathbf{C}^{-1/2}\mathbf{S} \end{pmatrix}, \tag{IV.50}$$

with $\mathbf{E}$ (respectively $\mathbf{S}$) the Mandel vector representation of the strain (respectively stress) tensor $\boldsymbol{E}$ (respectively $\boldsymbol{S}$) and $\mathbf{C}$ the Mandel matrix representation of the fourth-order tensor $\mathbb{C}$. The clustering analysis is then performed on the collection of the *transformed* strain-stress state pairs $\mathbf{z}_e^X$ of all the integration points e from all the snapshots X of the FE simulations represented in Fig. IV.8, using standard a k-means algorithm.

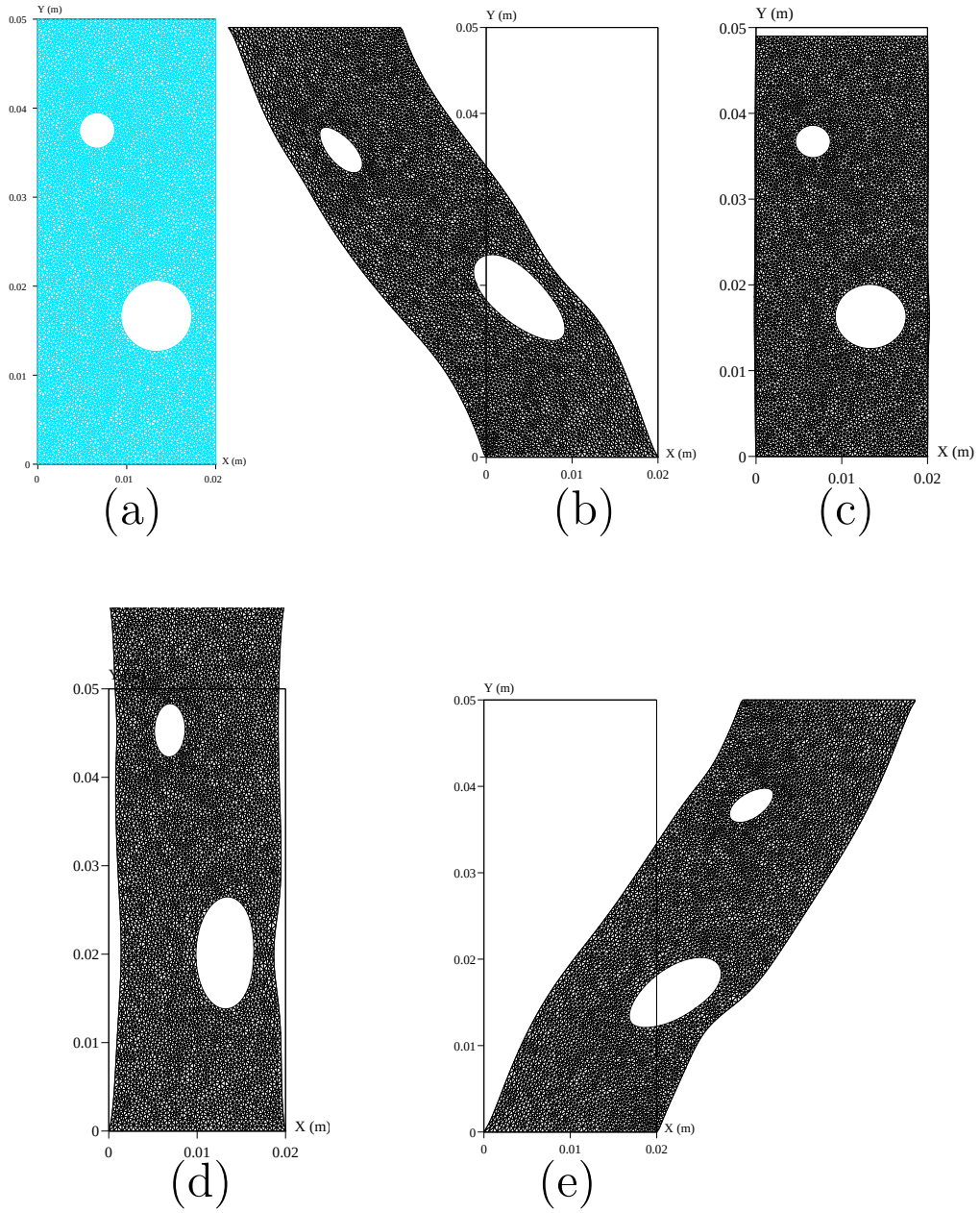


Figure IV.8 – Example of snapshots used to generate the DB-DDI database.
(a) Mesh of 10 738 linear triangular elements.
(b) Shear test $\boldsymbol{u}(X, Y = h) = -0.03\boldsymbol{e}_X$.
(c) Compression test $\boldsymbol{u}(X, Y = h) = -0.001\boldsymbol{e}_Y$.
(d) Tensile test $\boldsymbol{u}(X, Y = h) = 0.01\boldsymbol{e}_Y$.
(e) Shear test $\boldsymbol{u}(X, Y = h) = 0.03\boldsymbol{e}_X$.

3 Numerical application

3.1 Test case and material databases

Structure and mesh. We study the simultaneous tension and shearing of the two-dimensional thin membrane with a hole, presented in Chapter III. We assume both incompressibility of the material and plane stress conditions. To evaluate the convergence of the data-driven method with respect to mesh size, we consider two meshes for the samples. The two discretizations are represented in Fig. IV.9, together with the geometry and the boundary conditions. The first (respectively second) mesh, referred to as "coarse" (respectively "fine") contains $N = 1092$ (respectively 2495) nodes and 1015 (respectively 2375) bi-linear quadrangular elements such that the number of integration points amounts to $M = 4060$ (respectively $M = 9500$). The meshes were generated with Gmsh (Geuzaine and Remacle, 2009).

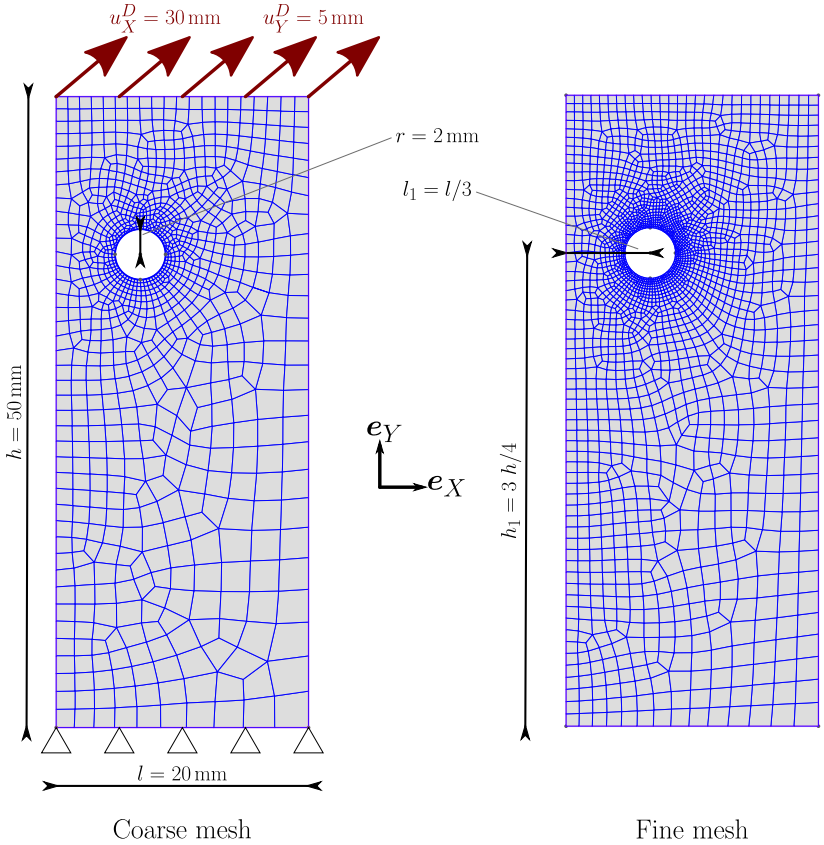


Figure IV.9 – Geometry, mesh and boundary conditions of the problem.

Material databases. We generate three families of material databases: DB-DENSE, DB-STD, DB-DDI according to the methods presented in Section 2. The

constitutive model used to generate the data is the neo-Hookean incompressible model, reduced to its two-dimensional expression in plane stress conditions recalled in Eq. (IV.3) from Appendix C. The material parameter is set to $\mu = 1.2\,\text{MPa}$.

To evaluate the convergence of the data-driven results with respect to the number of material data points, we vary the sampling parameters to generate DB-DENSE and DB-STD databases of different sizes:

- we generate 12 DB-DENSE databases, each comprising $m = N_{dir} \times N_{amp}$ points with $N_{dir} \in \{25, 256, 2500, 25\,600\}$ and $N_{amp} \in \{10, 1000, 1000\}$.
- we generate 3 DB-STD databases, each comprising $m = 3N_{K_3} \times N_\theta \times N_{amp}$ points with $N_{K_3} = 3$ and $N_\theta = 60$ (corresponding to $\Delta\theta = 3°$) fixed, and $N_{\text{amp}} =\in \{10, 1000, 1000\}$.

The DB-DDI database was generated with a total of 429 520 strain-stress states and comprises $m = 2200$ points. The $\mathbb{C}$ parameter used in the Euclidean mapping Eq. (IV.50) was set to the plane stress linear elastic tensor, with a Young's modulus of $E = 3\mu = 3.6\,\text{MPa}$ and a Poisson's ratio of $\nu = 0.5$. Note that this material database then contains less points than there are integration points in the mesh. Note also that the DB-DDI *does not* contain the reference solution exactly, since it was generated with the strain-stress states of a different structure and went through a clustering algorithm with no constraint on equilibrium or compatibility.

3.2 Comparison with the finite element reference solution

3.2.1 Results of the minimization

The data-driven computing is a distance-minimizing scheme. A first indicator of the success of the minimization is then the final value $d^2(z, \mathcal{D}) = d^2(z, z^*)$ of the square distance between the mechanical states $z = \{z_e = (\boldsymbol{E}_e, \boldsymbol{S}_e)\}_{e=1}^M$ and the associated material states $z^* = \{z^*_{ie} = (\boldsymbol{E}^*_{ie}, \boldsymbol{S}^*_{ie})\}_{e=1}^M$. This value can be compared to the minimal distance $d^2(z^{\text{ref}}, \mathcal{D}) = \min_{z' \in \mathcal{D}} d^2(z^{\text{ref}}, z')$ between the reference states $z^{\text{ref}} = \left\{z^{\text{ref}}_e = \left(\boldsymbol{E}^{\text{ref}}_e, \boldsymbol{S}^{\text{ref}}_e\right)\right\}_{e=1}^M$ (obtained from the classical FE simulation) and the material data set $\mathcal{D} = \{z \in \mathcal{Z} \mid z_e \in \mathcal{D}_{\text{loc}}\}$ with the material database $\mathcal{D}_{\text{loc}} = \{(\boldsymbol{E}_i, \boldsymbol{S}_i)\}_{i=1}^m$. We recall that the two square distances are respectively defined as

$$d^2(z, z^*) = \sum_{e=1}^{M} \frac{w_e}{2} \left(\Delta\boldsymbol{E}_e : \mathbb{C} : \Delta\boldsymbol{E}_e + \Delta\boldsymbol{S}_e : \mathbb{C}^{-1} : \Delta\boldsymbol{S}_e\right), \tag{IV.51}$$

$$d^2(z, \mathcal{D}) = \sum_{e=1}^{M} \min_{(\boldsymbol{E}_i, \boldsymbol{S}_i) \in \mathcal{D}_{\text{loc}}} \frac{w_e}{2} \left(\Delta\boldsymbol{E}^{\text{ref}}_{e,i} : \mathbb{C} : \Delta\boldsymbol{E}^{\text{ref}}_{e,i} + \Delta\boldsymbol{S}^{\text{ref}}_{e,i} : \mathbb{C}^{-1} : \Delta\boldsymbol{S}^{\text{ref}}_{e,i}\right), \tag{IV.52}$$

with w_e the integration weight, $\Delta\boldsymbol{E}_e = \boldsymbol{E}_e - \boldsymbol{E}^*_{ie}$ and $\Delta\boldsymbol{E}^{\text{ref}}_{e,i} = \boldsymbol{E}^{\text{ref}}_e - \boldsymbol{E}_i$ (and likewise for stresses). The values of both distances for each material database and each mesh size are given in Table IV.2.

Table IV.2 – Comparison of the value of the data-driven objective function $d^2(z, \mathcal{D})$ with the minimal square distance $d^2(z^{\text{ref}}, \mathcal{D})$ between the reference solution and the material data set $\mathcal{D}$ for each type of database and each mesh. The values are given in $\text{Pa}\,\text{m}^2$ (*cf.* Eqs. (IV.51) and (IV.52)).

			coarse		fine	
		Database parameters	$d^2(z, \mathcal{D})$	$d^2(z^{\text{ref}}, \mathcal{D})$	$d^2(z, \mathcal{D})$	$d^2(z^{\text{ref}}, \mathcal{D})$
Database type	DB-DENSE	$N_{\text{dir}} = 25$, $N_{\text{amp}} = 10$	9.278	17.65	8.698	17.79
		$N_{\text{dir}} = 25$, $N_{\text{amp}} = 100$	12.02	15.32	11.79	15.32
		$N_{\text{dir}} = 25$, $N_{\text{amp}} = 1000$	12.21	15.24	12.42	15.24
		$N_{\text{dir}} = 256$, $N_{\text{amp}} = 10$	3.676	4.659	3.57	4.68
		$N_{\text{dir}} = 256$, $N_{\text{amp}} = 100$	2.831	2.219	2.676	2.209
		$N_{\text{dir}} = 256$, $N_{\text{amp}} = 1000$	2.768	2.128	2.209	2.128
		$N_{\text{dir}} = 2500$, $N_{\text{amp}} = 10$	2.612	2.754	2.114	2.772
		$N_{\text{dir}} = 2500$, $N_{\text{amp}} = 100$	0.415	0.298	0.447	0.293
		$N_{\text{dir}} = 2500$, $N_{\text{amp}} = 1000$	0.407	0.21	0.418	0.21
		$N_{\text{dir}} = 25600$, $N_{\text{amp}} = 10$	2.48	2.57	1.9	2.58
		$N_{\text{dir}} = 25600$, $N_{\text{amp}} = 100$	0.114	0.111	0.116	0.105
		$N_{\text{dir}} = 25600$, $N_{\text{amp}} = 1000$	0.045	0.024	0.045	0.023
	DB-STD	$N_\theta = 61$, $N_{\text{amp}} = 10$	4.029	4.755		
		$N_\theta = 61$, $N_{\text{amp}} = 100$	4.722	2.329		
		$N_\theta = 61$, $N_{\text{amp}} = 1000$	5.126	2.243		
	DB-DDI	$m = 2200$	0.941	1.43	0.94	1.39

Let us draw attention on several results given by Table IV.2:

- First, note that the values do not change much from the coarse mesh to the fine mesh. We can then consider that the data-driven solution is converged in terms of spatial discretization, with respect to this indicator.
- Second, the closest databases to the reference solution are, as expected, the largest DB-DENSE (and notably the one where $N_{\text{dir}} = 25\,600$ and $N_{\text{amp}} = 1000$) and the DB-DDI. For the former, this demonstrates that the sampling strategy efficiently densifies the data set so as to increase its capacity to represent the mechanical problem. For the latter, this indicates that a well-designed material database (*i.e.* containing states that are representative of what is expected in the structure) performs already very well with a few material data points.
- Third, the value of the objective function $d^2(z, \mathcal{D})$ is smaller than the square distance $d^2(z^{\text{ref}}, \mathcal{D})$ between the reference solution and the material database for the smallest DB-DENSE databases, and conversely for the largest DB-DENSE. This demonstrates the convergence of the method with the density of the material database: the mechanical states are, by definition, designed to minimize the distance between the constraint set $\mathcal{E}$ to the material data set $\mathcal{D}$. By contrast, the reference solution z^{ref} lies at the exact intersection $\check{\mathcal{D}} \cap \mathcal{E}$, with $\check{\mathcal{D}}$ the continuous material data set which is sampled by the discrete material data set $\mathcal{D}$. The more material data points in $\mathcal{D}$, the finer the approximation of $\check{\mathcal{D}}$, and then, the closer the reference solution gets to $\mathcal{D}$. We add that this could reveal the limitations of the alternated minimization as well: the more points in the largest DB-DENSE database, the more complex the combinatorial optimization, and conversely, the sparsity of the smallest DB-DENSE may increase the amount of local minimizers.

3.2.2 Relative error to the reference solution

To explore further the convergence of the data-driven results to the reference solution, let us examine the RMS percent errors in strain and stress. Recalling the definitions given in Eq. (III.115), the RMS percent errors write

$$E_{(\%\text{RMS})} = \left(\frac{\sum_{e=1}^{M} w_e \, W\left(\boldsymbol{E}_e - \boldsymbol{E}_e^{\text{ref}}\right)}{\sum_{e=1}^{M} w_e \, W\left(\boldsymbol{E}_e^{\text{ref}}\right)} \right)^{\frac{1}{2}}$$

$$S_{(\%\text{RMS})} = \left(\frac{\sum_{e=1}^{M} w_e \, W^*\left(\boldsymbol{S}_e - \boldsymbol{S}_e^{\text{ref}}\right)}{\sum_{e=1}^{M} w_e \, W^*\left(\boldsymbol{S}_e^{\text{ref}}\right)} \right)^{\frac{1}{2}}$$

where $W(\boldsymbol{A}) = (\mathbb{C} : \boldsymbol{A}) : \boldsymbol{A}$ and $W^*(\boldsymbol{A}) = \left(\mathbb{C}^{-1} : \boldsymbol{A}\right) : \boldsymbol{A}$.

Let us first evaluate the convergence with the DB-DENSE databases alone. As can be seen from Fig. IV.10, both RMS percent errors decrease with increasing number of sampled directions N_{dir}, for each data set of fixed amplitudes N_{amp} =constant. In addition, we see almost no difference between the two mesh discretizations. Finally,

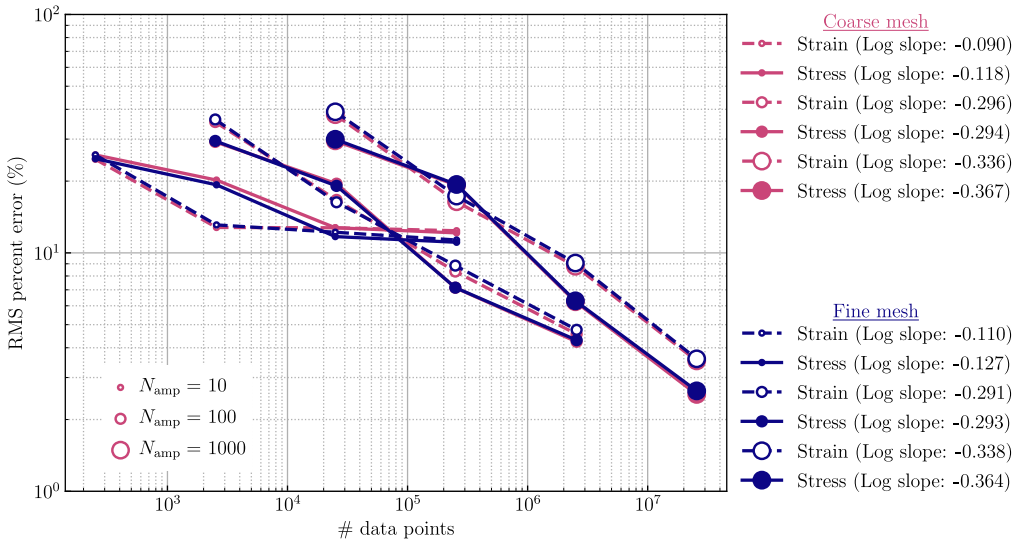


Figure IV.10 – Convergence of the RMS percent errors in strain and stress with respect to the size of the DB-DENSE database. The color indicates the mesh discretization: light pink for coarse and purple for fine. The size of the circles increases with the number of samples amplitudes $N_{amp} \in \{10, 100, 1000\}$. Each dotted line (resp. solid line) then represents the RMS error in strain (resp. in stress) for databases of fixed N_{amp}.

the convergence rate significantly increases from $N_{amp} = 10$ to $N_{amp} = 100$ but seems to stabilize around -0.324 for $N_{amp} = 100$ and $N_{amp} = 1000$.

Otherwise, the number of sampled amplitudes has little influence on the convergence, as illustrated in Fig. IV.11. Then, the RMS percent errors are not much improved from the addition of amplitudes to the database. Together with the results of the DB-DENSE databases (circles), we added in this plot the results for the DB-STD (triangles) and DB-DDI (diamond) databases. Note that all of them present an error above 5 % and that, again, mesh size has no influence. Also, the DB-STD databases behave like the DB-DENSE databases. This is expected as the number of directions were set to $N_{dir} = 3 \times N_{K_3} \times N_\theta$ when we generated the DB-STD databases. Finally, we can see that the only discretization of the DB-DDI performs quite well with few material data points. The lowest RMS errors are obtained for the largest DB-DENSE database (see the large circles in the bottom right corner of Fig. IV.11): $\left(E_{(\%\mathrm{RMS})}, S_{(\%\mathrm{RMS})}\right) = (3.52\,\%, 2.55\,\%)$.This is achieved at high computational cost: the DB-DENSE database with $N_{dir} = 25\,600$ and $N_{amp} = 1000$ comprises 25 600 000 points. As a comparison, a regular grid of this size would correspond to a sample of only 295 points for each component of the strain tensor.

The RMS percent errors analyzed above are global results in the structure and could be influenced by a small amount of very badly predicted mechanical states. To evaluate this claim, we analyze the local strain and stress relative errors defined

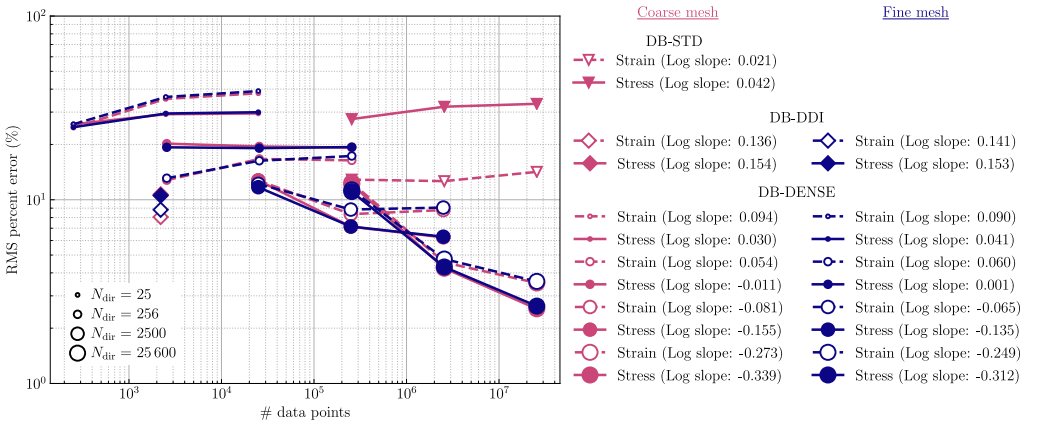


Figure IV.11 – Convergence of the RMS percent errors in strain and stress with the size of all kinds of databases. The color indicates the mesh discretization: light pink for coarse and purple for fine. The DB-DENSE results are depicted by circles, the DB-STD results by triangles and the DB-DDI results by diamonds. For the DB-DENSE databases, the size of the circles increases with the number of sampled directions $N_{\text{dir}} = \{25, 256, 2500, 25\,600\}$. Each dotted line (resp. solid line) then represents the RMS error in strain (resp. in stress) for databases of fixed N_{dir}.

as

$$E^e_{(\%\text{err})} = \frac{W\left(\boldsymbol{E}_e - \boldsymbol{E}_e^{\text{ref}}\right)}{W\left(\boldsymbol{F}_e^{\text{ref}}\right)} \text{ and } S^e_{(\%\text{err})} = \frac{W^*\left(\boldsymbol{S}_e - \boldsymbol{S}_e^{\text{ref}}\right)}{W^*\left(\boldsymbol{S}_e^{\text{ref}}\right)} \tag{IV.53}$$

for all integration point $e \in [1\,..\,M]$. The histograms of the local errors for each kind of databases are shown in Figs. IV.12 to IV.14. The candidates for the DB-DENSE and DB-STD were chosen as the one giving the best results for the global RMS errors.

Fig. IV.12 shows that more than 90 % of the integration points present a relative local error below 0.1 %, both in strain and stress, for the results obtained with the DB-DENSE database. These are the lowest error that have been obtained: for the DB-DDI, only about 20 % of the integration points are below 0.1 % errors; this value falls down below 10 % of the integration points in the DB-STD case (see Figs. IV.13 and IV.14). The DB-DENSE was expected to out-perform the other databases from this point of view: it is clear that the denser the database, the better the discrete material database approximates the constitutive response used for the FE reference solution.

3.2.3 Global response of the structure

We close this section with a discussion of a more global indicator. In Figs. IV.15 to IV.17, we show the global response of the structure for each family of material databases. The DB-DDI and DB-DENSE database provide very satisfactory results. It is notable for the DB-DENSE that the local strain and stress errors revealed in the

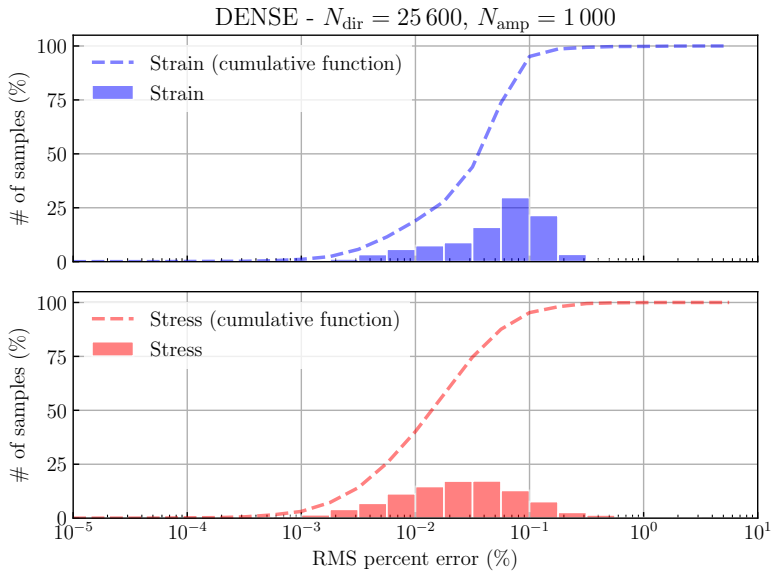


Figure IV.12 – Local strain and stress errors for the DB-DENSE database generated with $N_{dir} = 256\,000$ and $N_{\text{amp}} = 1000$.

previous sections have no impact on the total force at the grip. This implies that the data-driven results can be reliable for global analysis such as the force-displacement curve.

3.3 Analysis of the data-driven results

So far, we have evaluated the performance of the data-driven results only with respect to their ability to reproduce the reference solution. This exercise is only relevant to show that the data-driven computing paradigm encompasses the constitutive modeling-based framework: as the material database approximates increasingly closely the constitutive manifold, the data-driven solution converges to the classical solution. However, the approach was introduced to address problems which can not fit in the classical formulation, when a constitutive relation is not tractable for instance. In that case, the data-driven results must be analyzed for what they are: a mechanically admissible response, which is as close as possible to what has been measured of the material response, through discrete points only. In this section, we thus compare the different data-driven results *with each other*, without the reference solution.

3.3.1 Objective function *vs.* database size

First, we examine the evolution of the objective function, *i.e.* the converged value of the distance between the mechanical and material states, with respect to the size of the database. Similarly to Figs. IV.10 and IV.11, Figs. IV.18 and IV.19 present the results for the DB-DENSE databases of fixed N_{amp}, and the results for the DB-DENSE

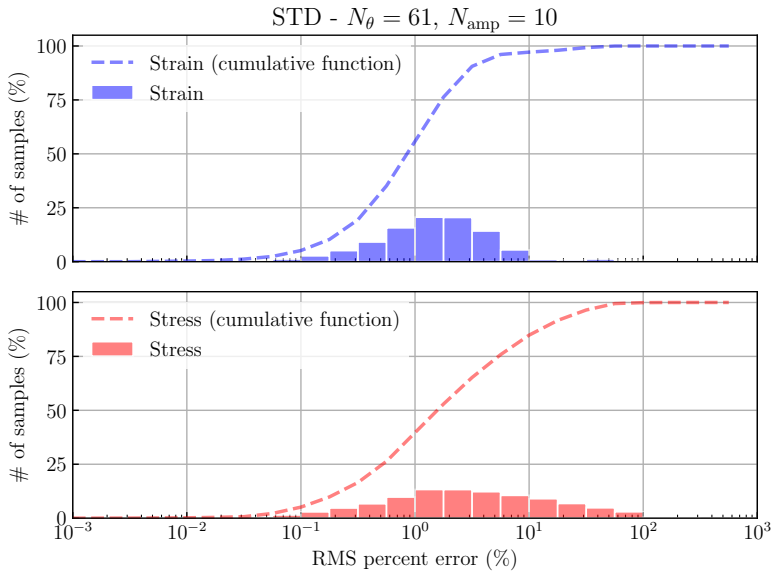


Figure IV.13 – Local strain and stress errors for the DB-STD database generated with $N_\theta = 61$ and $N_{\text{amp}} = 10$.

databases of fixed N_{dir} together with the DB-STD and DB-DDI results respectively. In both figures, the mesh discretization again proves to have little to no influence. This indicates that the data-driven solution is already converged with respect to mesh size for the coarse mesh. In addition, the best performing database with respect to this indicator is the largest DB-DENSE database with ($N_{\text{amp}} = 1000, N_{\text{dir}} = 25\,600$): as expected, the more points in the database, the closer it can get to the mechanical solution. For here on out then, we only show the results for the coarse mesh and restrict our discussion on the DB-DENSE databases to the largest one, unless otherwise necessary.

Fig. IV.18 shows that the databases comprising the lowest number $N_{\text{amp}} = 10$ of sampled amplitudes have the worst convergence rate. For the two other sampling, the addition of directions to the database is shown to greatly improve the convergence of the method. On the contrary, as can be seen from Fig. IV.19, the addition of amplitudes to a set of fixed N_{dir} directions does not clearly improve the final distance between mechanical and material states. This indicates as aforementioned that the richness of the database corresponds better to the mode-of-distortion and the orientation angle than to the amplitude. Indeed, thanks to the analysis performed on the dense sampling method in Section 2.1.3, we know that the sampling of the direction vector a is equivalent to a sampling of both the modes-of-distortion represented by invariant K_3 and the orientation angles θ. This statement is confirmed with the results obtained from the DB-STD database: the addition of amplitudes even slightly increased the final value of the objective function. Moreover, in this family of databases, the number $N_\theta = 61$ of orientation angles is much higher than the number of sampled modes-of-distortion ($N_{K_3} = 3$). Then, the DDCM clearly

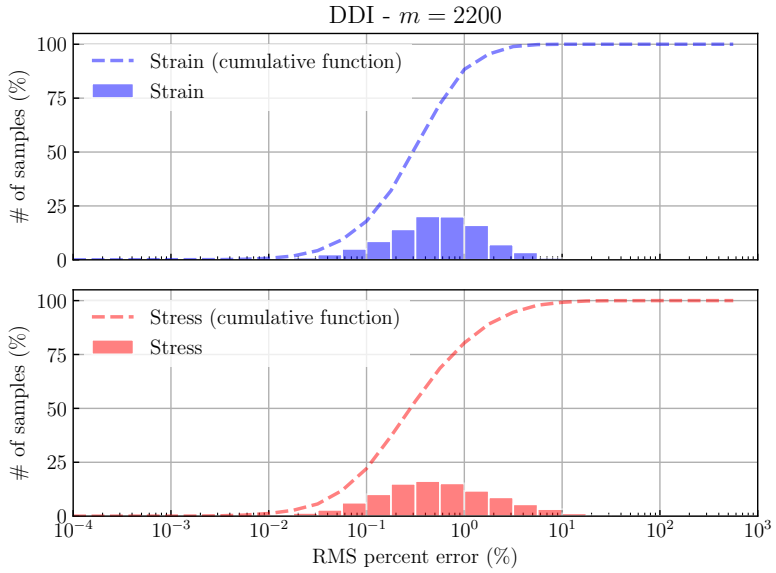


Figure IV.14 – Local strain and stress errors for the DB-DDI database generated with $m = 2200$.

needs the databases to comprise a variety of modes-of-distortion.

3.3.2 Stress results

A finite element simulation is often performed in order to predict the stress distribution in the structure. The performance of the finite strain DDCM is now evaluated according to the local stress states. We consider the Cauchy stress field $\boldsymbol{\sigma}$ as it mixes the strain and stress outputs of the simulation. In Fig. IV.20, the maps of the three components $\sigma_{xx}, \sigma_{xy}, \sigma_{yy}$ of the stress tensor, for the largest DB-DENSE database are depicted. Figs. IV.21 and IV.22 show the same maps for the DB-STD and DB-DDI database respectively. As can be seen from Figs. IV.20 and IV.22, the results for the largest DB-DENSE and the DB-DDI databases are very similar while their numbers of material data points are very different. This could be explained by the fact that, actually, a large part of the data points in DB-DENSE is not used during the data-driven simulation: only 131 149 (respectively 259 257) distinct material data points are selected at least once, throughout all load steps and iterations, by the coarse mesh (respectively the fine mesh), out of the 256 000 available. We can assume that these actually selected points correspond more or less to the one contained in the DB-DDI database. From Fig. IV.21, the limitations of the DB-STD database are obvious: the stress fields are not smooth and we know from the above results that is it not due to the mesh discretization. This is an indicator of the sparsity of the material database: large regions of the phase space $\mathcal{Z}$ are not covered by the data set $\mathcal{D} \subset \mathcal{Z}$ with this sampling method; this results in a poor projection on the constraint set of mechanical admissibility $\mathcal{E} \subset \mathcal{Z}$.

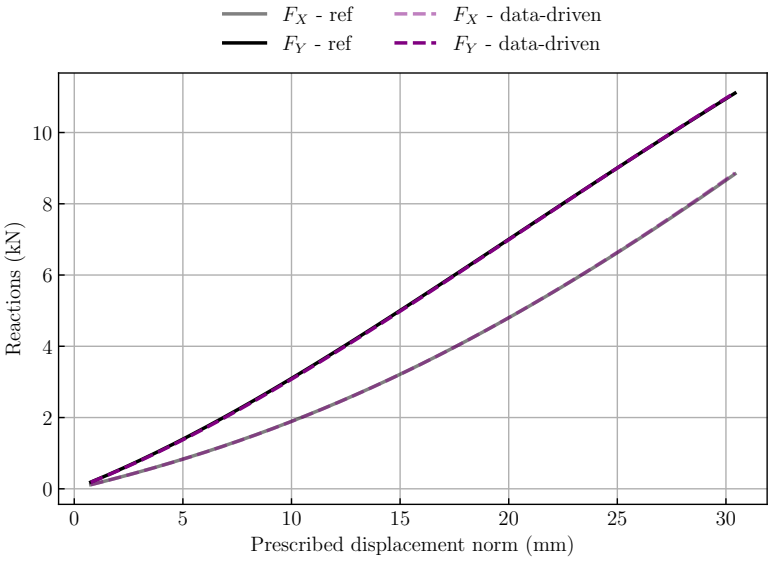


Figure IV.15 – Global response of the structure with the DB-DENSE database generated with $N_\theta = 256\,000$ and $N_{\text{amp}} = 1000$.

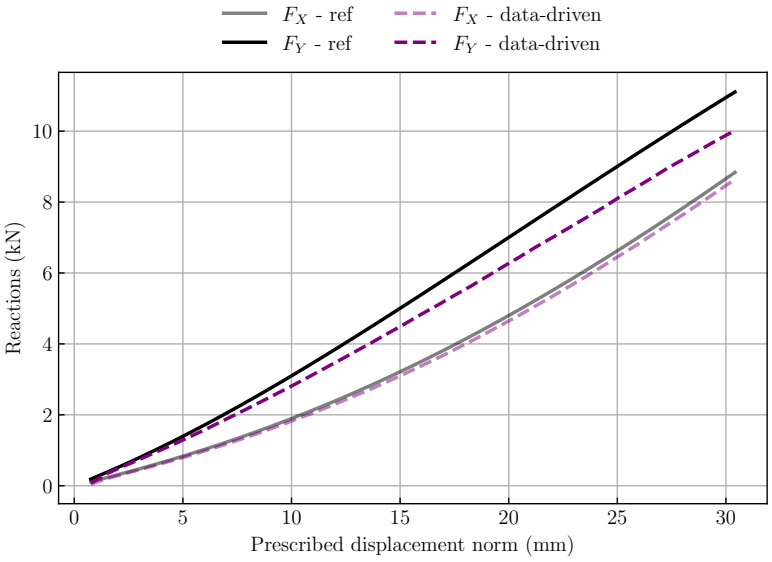


Figure IV.16 – Global response of the structure with the DB-STD database generated with $N_\theta = 61$ and $N_{\text{amp}} = 100$.

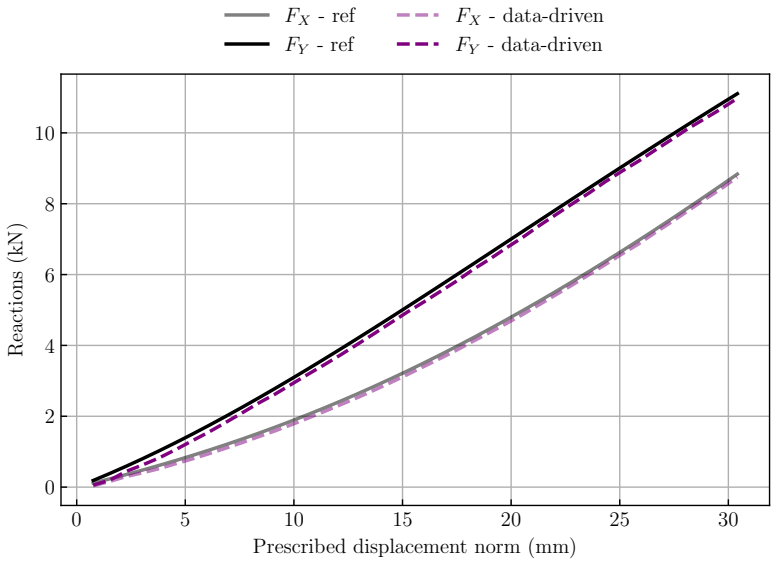


Figure IV.17 – Global response of the structure obtained with the DB-DDI database generated with $m = 2200$.

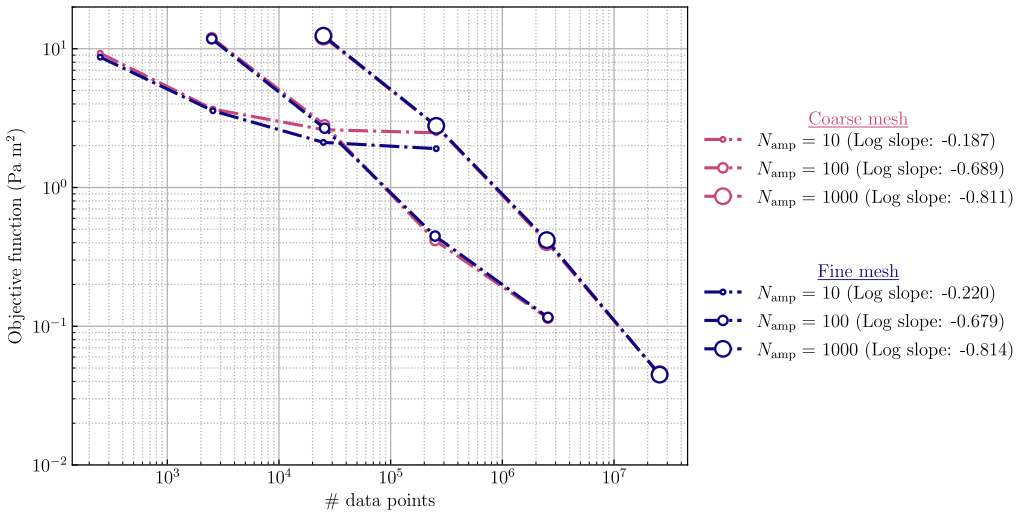


Figure IV.18 – Convergence of the objective function respect to the size of the DB-DENSE database. The color indicates the mesh discretization: light pink for coarse and purple for fine. The size of the circles increases with the number of samples amplitudes $N_{\text{amp}} = \{10, 100, 1000\}$. Each dotted line then represents the value of the objective function for databases of fixed number of sampled amplitudes N_{amp}.

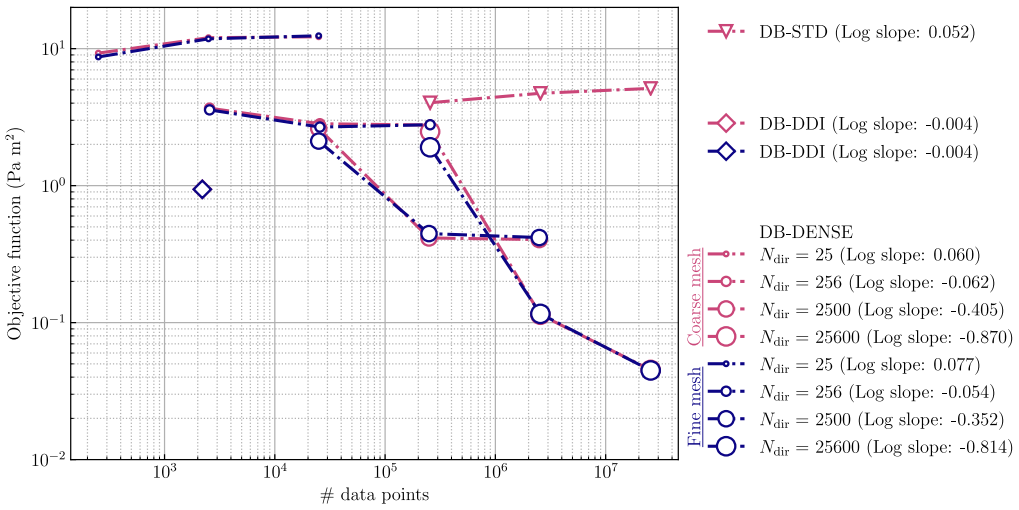


Figure IV.19 – Convergence of the objective function respect to the size of all kinds of databases. The color indicates the mesh discretization: light pink for coarse and purple for fine. The DB-DENSE results are depicted by circles, the DB-STD results by triangles and the DB-DDI results by diamonds. For the DB-DENSE databases, the size of the circles increases with the number of sampled directions $N_{\text{dir}} = \{25, 256, 2500, 25600\}$. Each dotted line then represents the value of the objective function for databases of fixed number of sampled amplitudes N_{dir}.

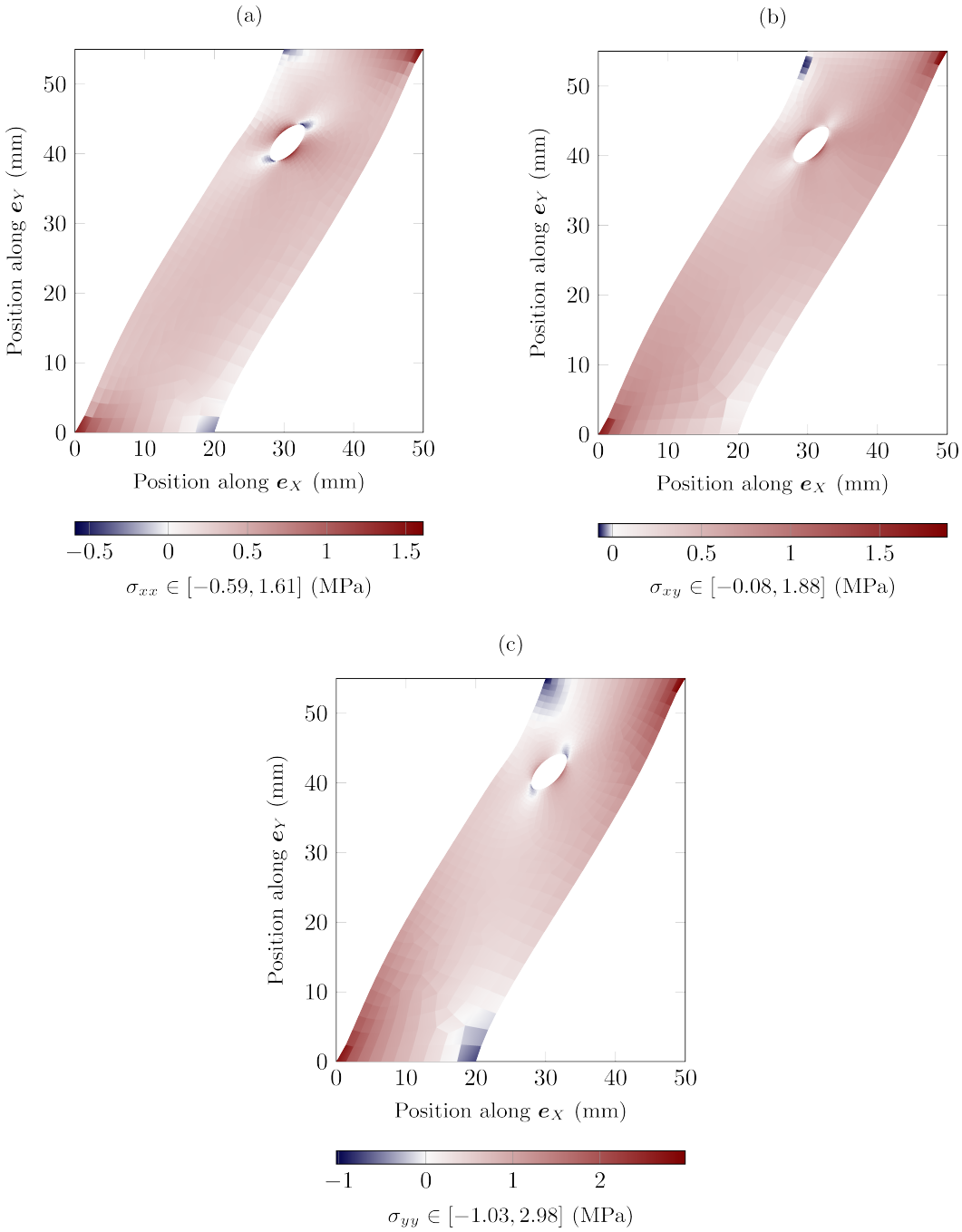


Figure IV.20 – Cauchy stress field obtained with the coarse mesh and the largest DB-DENSE database.

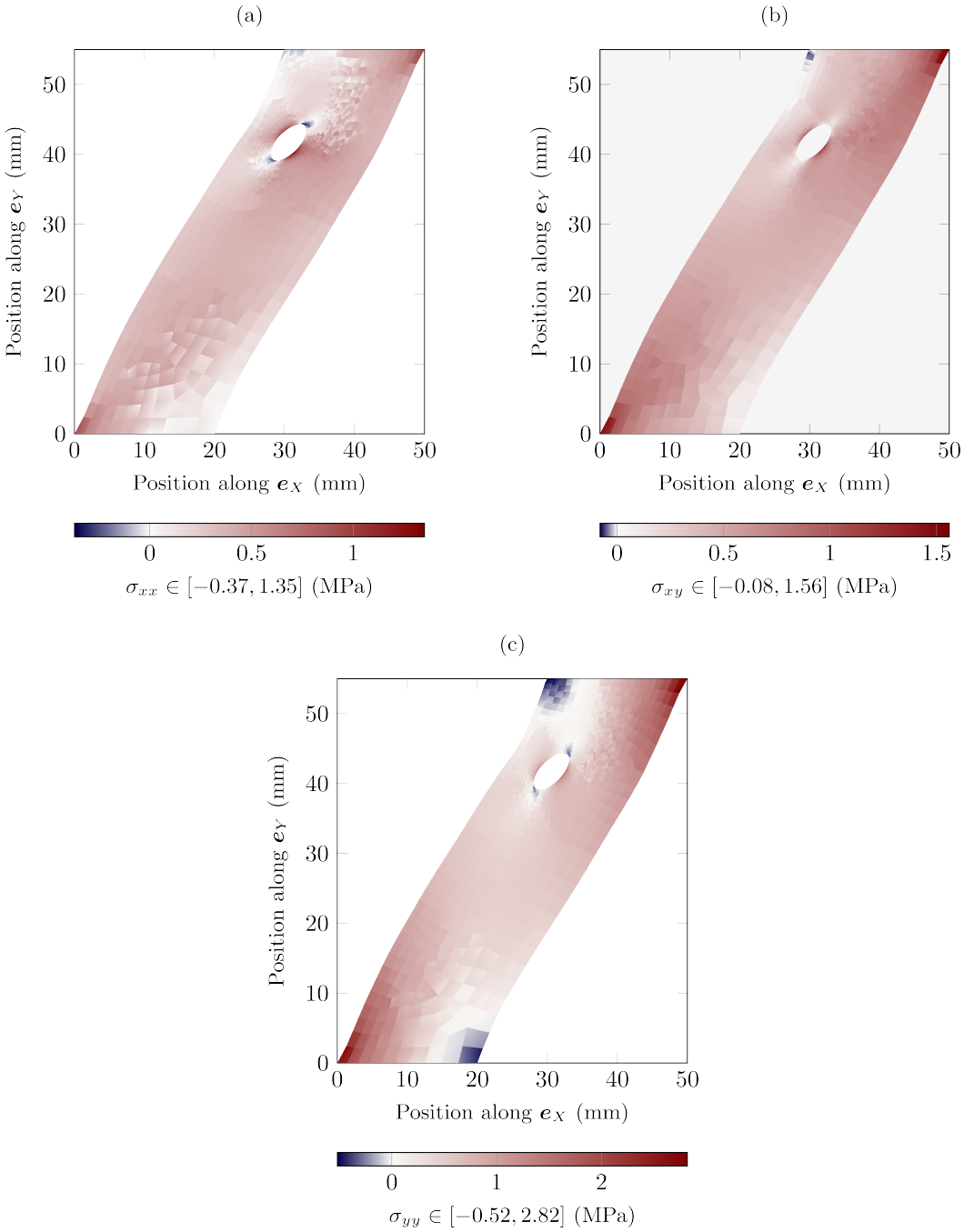


Figure IV.21 – Cauchy stress field obtained with the coarse mesh and the smallest DB-STD database ($N_{\text{amp}} = 10$).

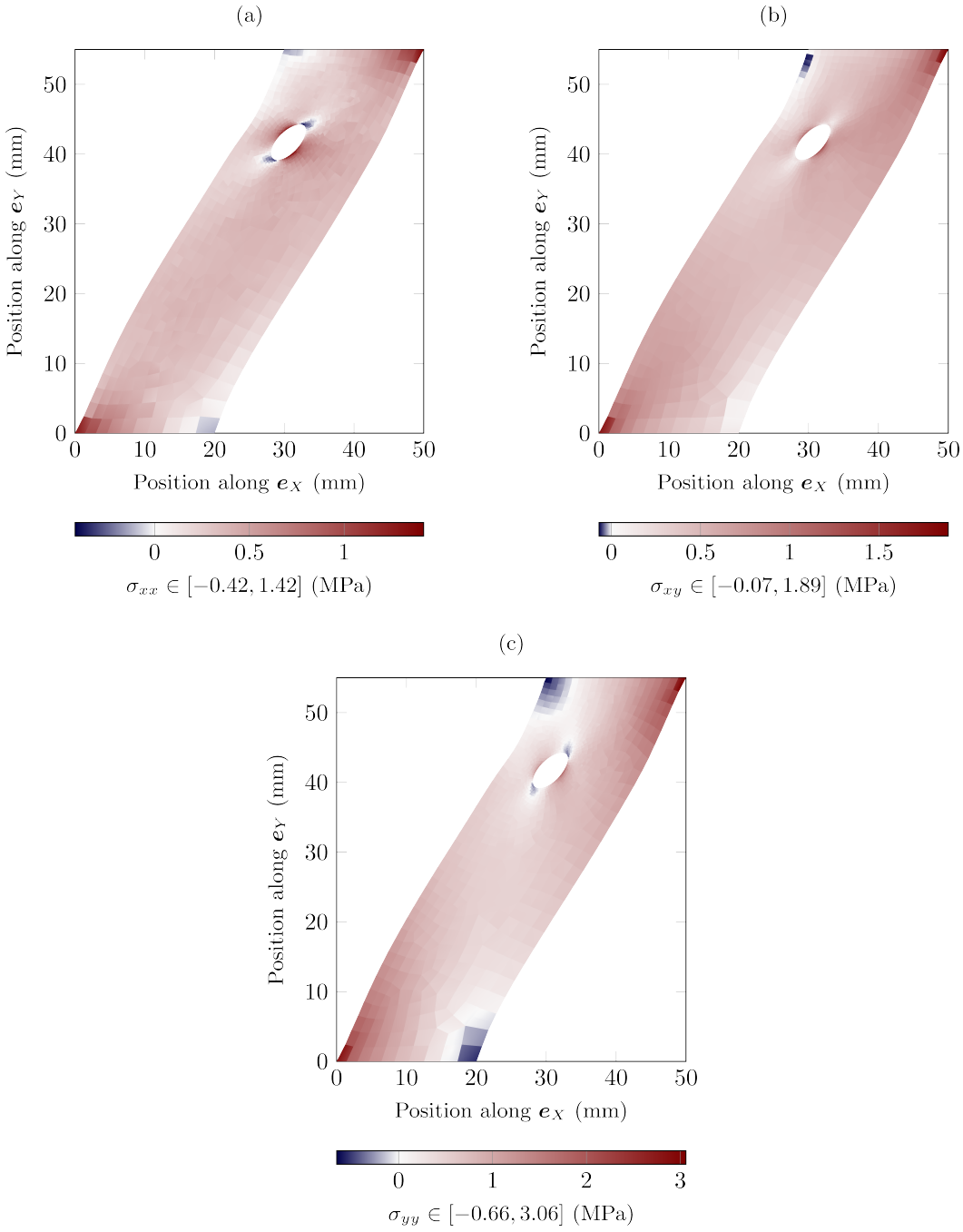


Figure IV.22 – Cauchy stress field obtained with the coarse mesh and the DB-DDI database.

3.3.3 Strain results

From the analysis conducted in Section 2.1.3, we can now represent the strain fields with the aid of three indicators: invariants K_2 and K_3 of the Lagrangian Hencky strain tensor which correspond respectively to the magnitude and the mode of distortion, and the orientation angle θ which indicates the direction of the largest in-plane principal stretch. To this end, we represent the mechanical states of all integration points of the coarse mesh on a polar plot, such as the example provided in Fig. IV.23. The angular position of a scatter point on the polar plot corresponds to its orientation angle $\theta \in [0°, 180°)$, while its radial position gives the value of $K_2 \in (0, 1]$. Finally, the color of the point corresponds to its K_3 value. On this type of plots, one can then readily read the mode, direction and amplitude of deformation of the mechanical states.

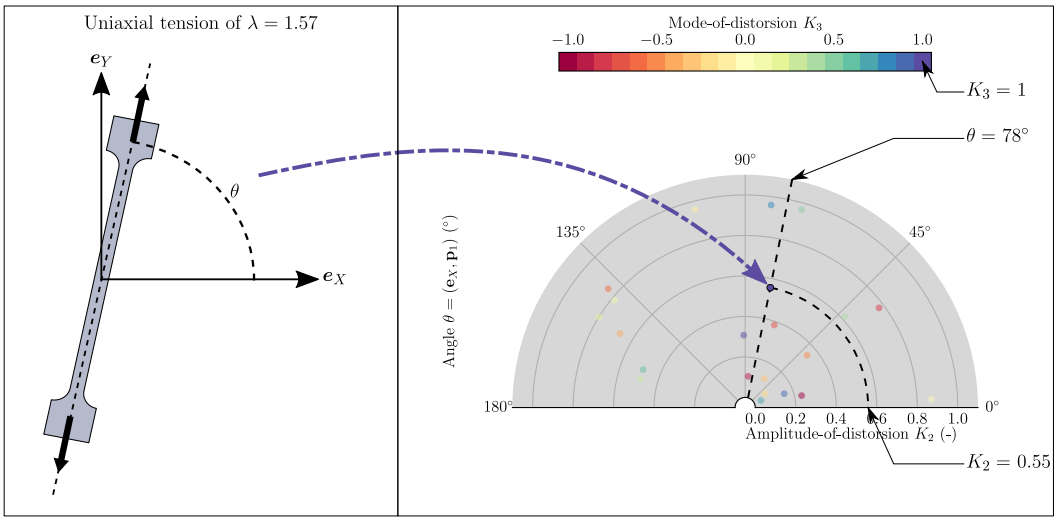


Figure IV.23 – Description of the scatter polar plot of the mechanical strain states in the structure.

In Fig. IV.24, the strain analysis of the mechanical states obtained with the largest DB-DENSE database is depicted. In addition to the information presented in Fig. IV.23, we added a histogram of the values of K_3 underneath the colorbar: the length of the grays bars indicates the occurrence of the corresponding color in the scatter plot. The black rectangle at the furthest right shows the length of a bar corresponding to 10 % of occurrence of the integration points. This indicator proves that most elements of the membrane are subject to uniaxial tension ($K_3 = 1$). This is also visible through the amount of points of violet color. The spreading of the scatter points in the polar plane demonstrates that a large part of the integration points are stretched along a direction close to $\boldsymbol{e}_Y$. The remaining elements yield an orientation angle comprised between 45° and 90°. This is consistent with the loading conditions: the top of the membrane is pulled in both positive $\boldsymbol{e}_X$ and $\boldsymbol{e}_Y$ directions. Finally, a certain part of the integration points in the membrane reads

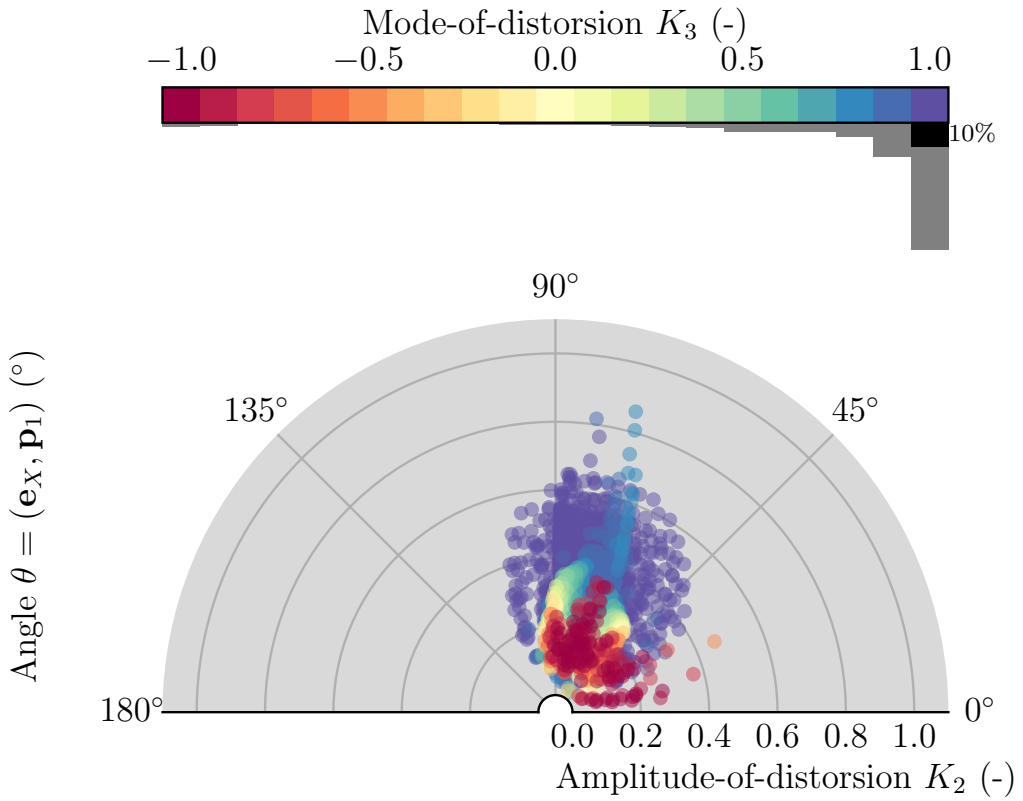


Figure IV.24 – Scatter polar plot representing the strain analysis of the mechanical states obtained with the largest DB-DENSE database ($N_{\text{amp}} = 1000$, $N_{\text{dir}} = 256\,000$). The gray bars underneath the colorbar of K_3 correspond to a histogram. All points are semi-transparent to make superimposed points visible.

$K_3 = -1$, which correspond to equi-biaxial tension. However, we recall that K_2 and K_3 are not able to make the difference between equi-biaxial tension and uniaxial compression. Only the values of the in-plane principal stretches λ_1, λ_2 can indicate if the point is subject to equi-biaxial tension ($\lambda_1 = \lambda_2$) or uniaxial compression ($\lambda_1 > \lambda_2$ with $\lambda_1 > 1$). Here, all points are in fact in uniaxial compression (data not shown).

In Fig. IV.25, we represent the strain analysis of the mechanical states obtained with the smallest DB-STD database. First, note that, as previously, the most represented mode-of-distortion is $K_3 = 1$. Second, note that the spreading of the points in the polar plane are a bit different than in Fig. IV.24. Third, it is remarkable that the mechanical states yield other values of K_3 than -1, 0 and 1. Indeed, only this three modes-of-distortion were used to generate the database. This indicates that the projection of the material database on the constraint set of mechanical admissibility is able to change the mode-of-distortion and therefore compensate the sparsity of the database on that level. However, as demonstrated earlier on the

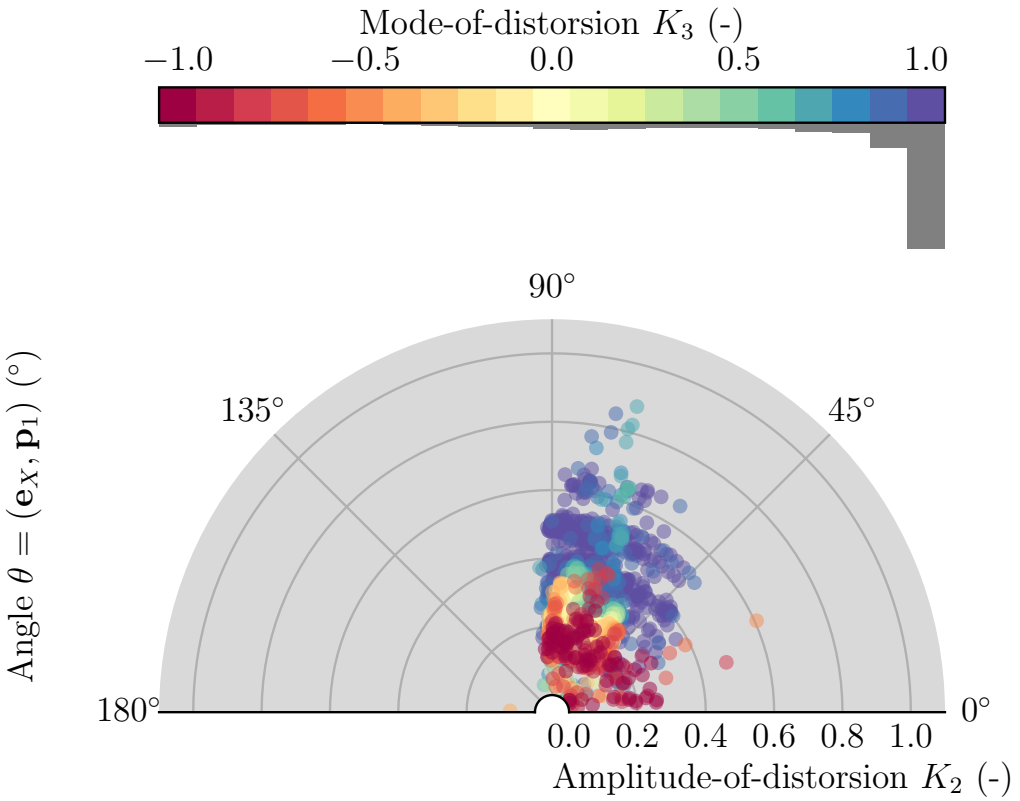


Figure IV.25 – Scatter polar plot representing the strain analysis of the mechanical states obtained with the smallest DB-STD database ($N_{\text{amp}} = 10$). The gray bars underneath the colorbar of K_3 correspond to a histogram. All points are semi-transparent to make superimposed points visible.

Cauchy stress field represented in Fig. IV.21, the sparsity of the database is too high to be fully compensated by the compatibility and equilibrium constraints.

Finally, we show the results obtained from the DB-DDI database in Fig. IV.26. The spreading of the points is similar to the one of the DB-DENSE database, although the number of points in the database is much smaller. Again, the uniaxial tension is highly represented.

3.3.4 Recovering isotropy

As proposed by Leygue et al. (2018); Dalémat et al. (2019), we now measure the ability of the data-driven solver to recover the isotropy of the material. Indeed, the material database has been generated with an isotropic constitutive model. However, no specific assumption on material symmetry is made in the data-driven solver. To verify if the mechanical states recover the isotropy of the material, we compute the mismatch angle α between the eigenvectors corresponding to the respective highest eigenvalues of the two-dimensional strain $\boldsymbol{E}$ and stress $\boldsymbol{S}$ tensors. It is defined

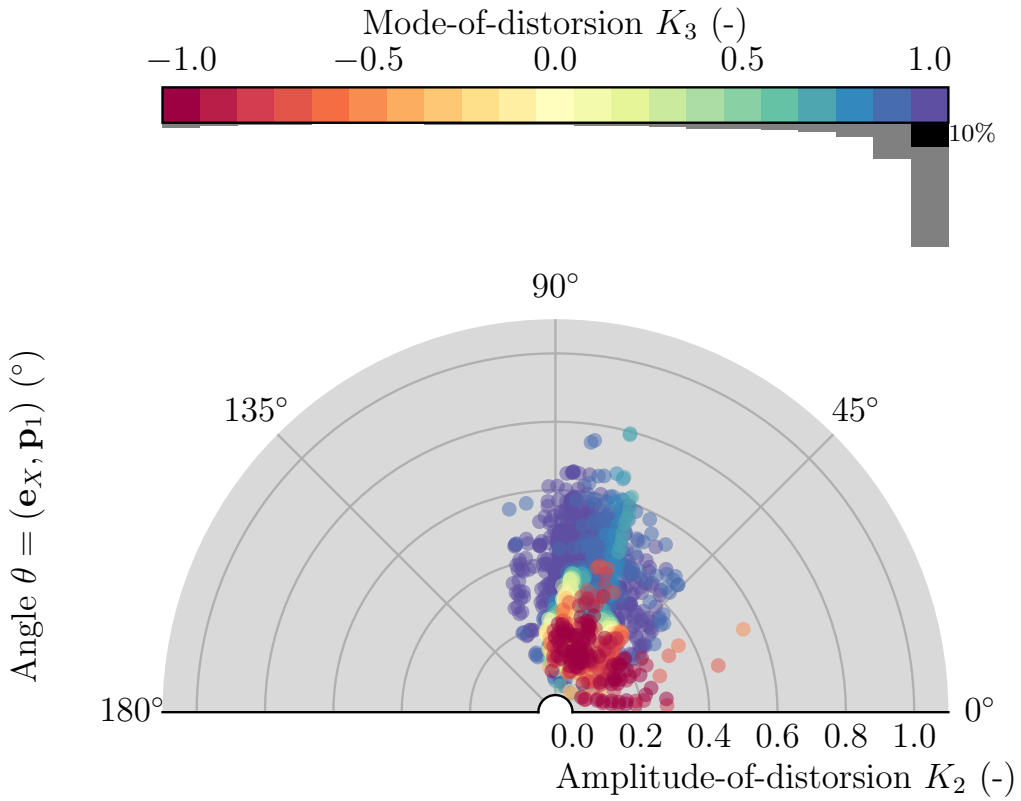


Figure IV.26 – Scatter polar plot representing the strain analysis of the mechanical states obtained with the DB-DDI database ($m = 2200$). The gray bars underneath the colorbar of K_3 correspond to a histogram. All points are semi-transparent to make superimposed points visible.

as

$$\alpha = \arctan\left(\frac{\boldsymbol{p}_2(\boldsymbol{E}) \cdot \boldsymbol{p}_1(\boldsymbol{S})}{\boldsymbol{p}_1(\boldsymbol{E}) \cdot \boldsymbol{p}_1(\boldsymbol{S})}\right) \quad \in [-90°, 90°], \tag{IV.54}$$

with $p_1(\boldsymbol{A})$ (respectively $p_2(\boldsymbol{A})$) the eigenvector corresponding to the highest (respectively lowest) eigenvalue of the two-dimensional tensor $\boldsymbol{A} \in \mathbb{R}^{2\times2}_{\text{sym}}$. With the definition Eq. (IV.54), the angle α is either positive or negative: we took the $(\boldsymbol{p}_1(\boldsymbol{E}), \boldsymbol{p}_2(\boldsymbol{E}))$ as the reference frame. We illustrate the positive and negative cases in Fig. IV.27.

In Fig. IV.28, we show histograms of the results for the DB-DENSE and DB-STD databases. In concordance with what we observed previously, the addition of sampled directions in DB-DENSE databases improves the results: in Fig. IV.28(a), we see that the histogram gets narrower as the number of sampled directions increases. The richer the database in directions, the closer the mechanical states to isotropic results. Conversely, increasing number of amplitudes in the DB-STD database tends to worsen the isotropy prediction (Fig. IV.28(b)). A possible explanation for this is

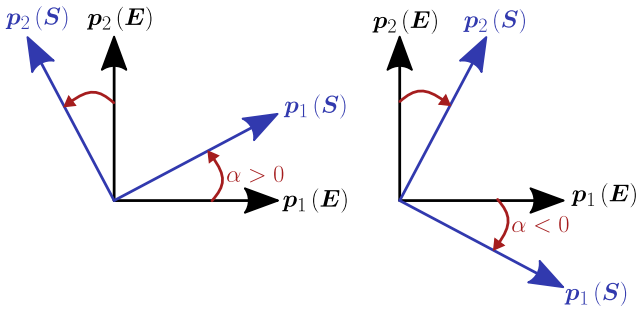
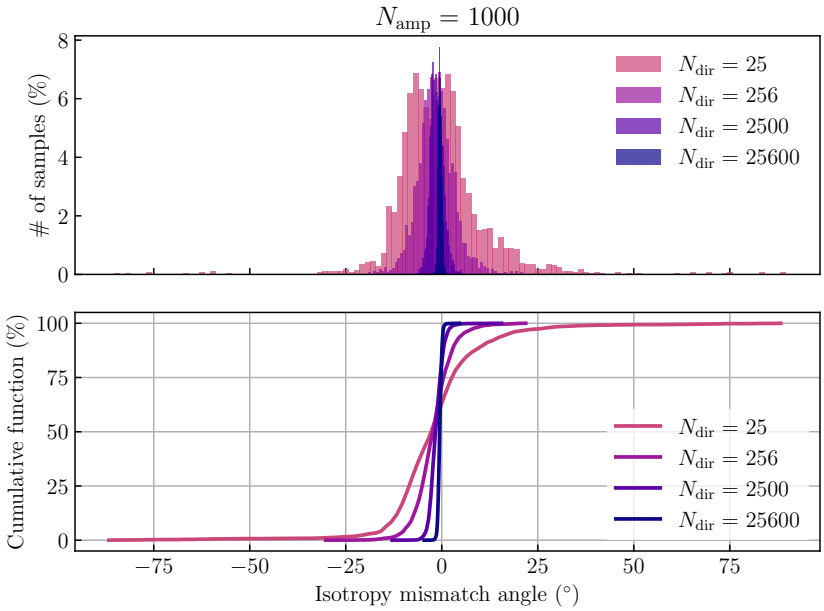


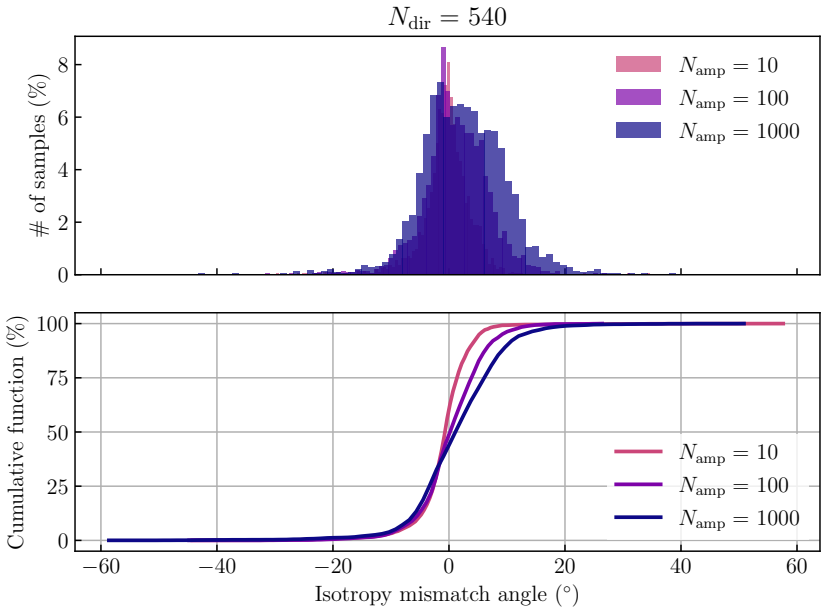
Figure IV.27 – Definition of the isotropy mismatch angle α, as the angle between the eigenvectors of the strain and stress tensors $\boldsymbol{E}$ and $\boldsymbol{S}$. Illustration of the positive and negative cases.

that increasing the number of amplitudes may densify distinct regions of the phase space, which would increase the amount of local minimizers. In addition, we recall that the direction vector a used to generate the DB-DENSE sets depends on both the invariant K_3 and the orientation angle θ. The number of sampled K_3 is fixed to only 3 while the number of sampled orientation angles θ is relatively large (61) in the DB-STD database. Hence, it is most likely that when adding directions to the DB-DENSE, the results are improved because this actually densified the sampling of K_3.

Finally, we compare the largest DB-DENSE results with the DB-DDI results in Fig. IV.29. The DB-DDI is clearly out-performed but given the respective number of points of each database, its performance can be considered as very satisfactory.



(a) DB-DENSE results.



(b) DB-STD results.

Figure IV.28 – Misalignment angle between the principal strain and stress of the mechanical states, for different kinds of database.

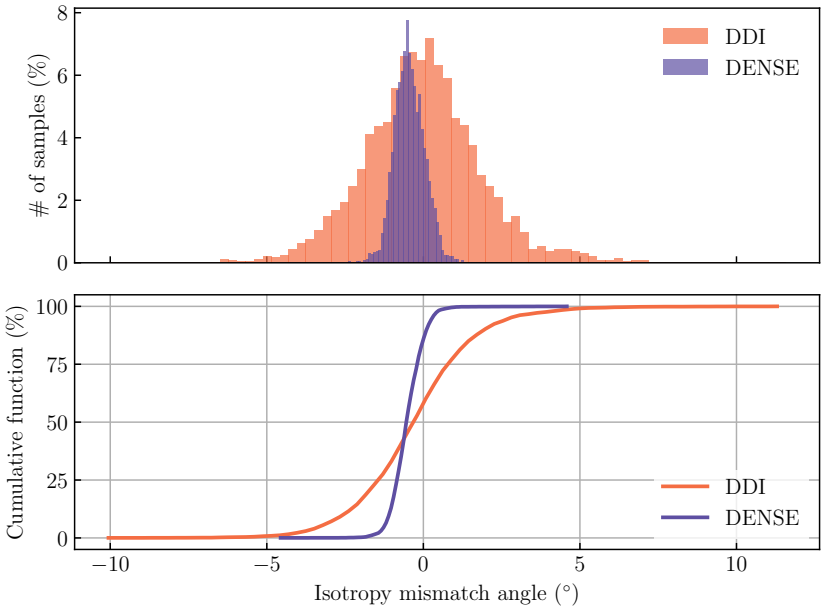


Figure IV.29 – Histograms of the misalignment angle between the principal strain and stress of the mechanical states, obtained with DB-DENSE, and DB-DDI databases.

4 Conclusion

As a conclusion, we summarize the main contributions of this chapter and highlight some of the key findings of Section 3. This chapter was devoted to the analysis of the content of material databases and its impact on the data-driven solution. We focused on the Lagrangian solver and a two-dimensional example for practical reasons, but a similar analysis could be conducted with the nominal solver and applied to three-dimensional problems. To this end, we generated synthetic material databases from a constitutive model that was later forgotten in the data-driven computations. This allowed for comparing the data-driven solution with the reference one, obtained from a classical FE simulation using the same constitutive model. The material databases were generated by sampling the strain space, with the corresponding stress obtained from a constitutive model. We analyzed and performed three different methods for the sampling of the strain space, which we briefly recall in the following.

1. First, we adapted the dense sampling method of the space of Lagrangian stretch tensors $\boldsymbol{U}$, first introduced by Kunc and Fritzen (2019a), to the specific case of two-dimensional plane stress and incompressible materials. This method relies on the DDMS of the stretch tensors and the exponential mapping: (i) the space of the Lagrangian Hencky strains are sampled on a orthonormal basis with an amplitude and a direction parameters, (ii) the matrix exponential yields a set of deviatoric stretch tensors which is multiplied by a sampling of the determinants to obtain the product set of the complete stretch tensors. Such databases we referred to as DB-DENSE. Together with adapting the approach, we built upon the invariant basis for natural strain theory of Criscione et al. (2000) to provide a mechanical analysis of the thus-generated databases. We are then able to characterize a set of strain tensors with three physically meaningful parameters, namely K_2 the magnitude-of-distortion, K_3 the mode-of-distortion and θ the orientation angle of the largest in-plane principal stretch with the canonical basis.

2. Second, we generated so-called DB-STD databases, which only contains the three standards tests that are used to calibrate the hyperelastic models, namely uniaxial tension, pure shear and equi-biaxial tension.

3. Third, we emulated a Lagrangian formulation of the DDI method with a clustering algorithm to generate importance sampling database, referred to as DB-DDI.

The performance of the three families of material databases were first evaluated with respect to their accordance with the FE reference solution. In particular, we studied the convergence of the data-driven solution to the reference one, with respect to the size of the data sets as well as the mesh discretization. From this investigation, we showed that the largest DB-DENSE database out-performed, as expected, the other types of sampling. In particular, the DB-STD proved to be insufficiently rich: the DDCM can not extrapolate from missing data. We confirmed with the two other

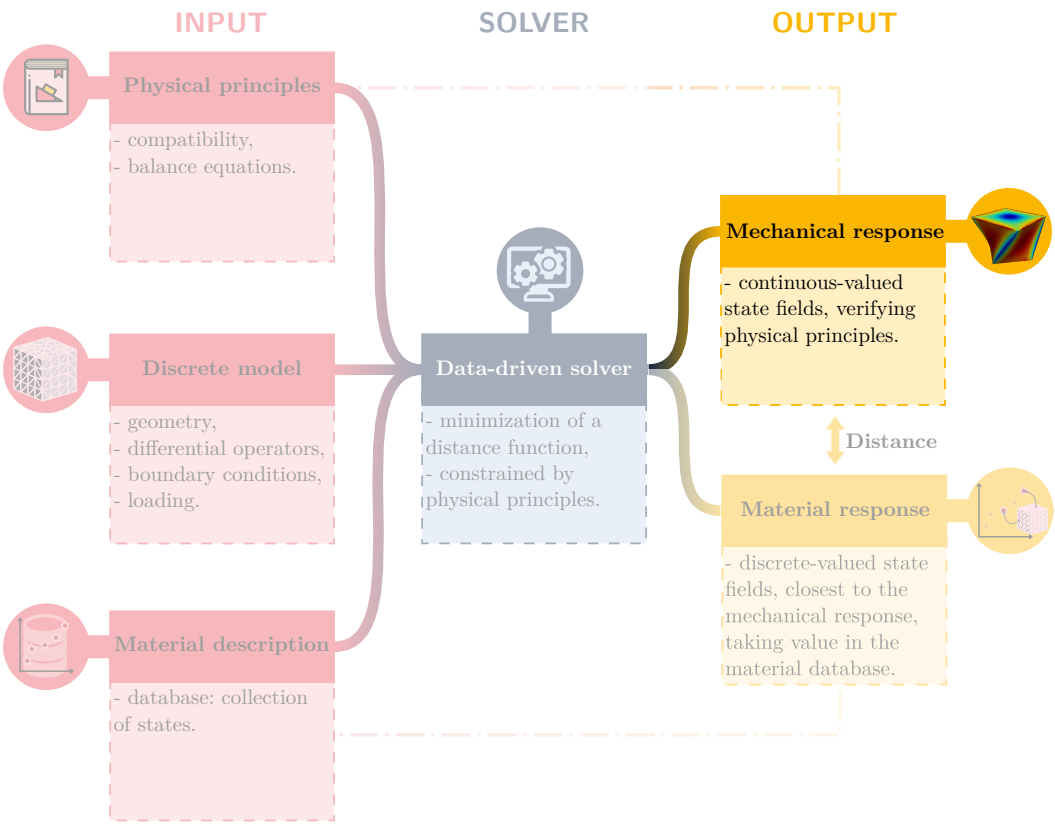
database that "richness" is mainly needed in terms of modes-of-distortion. Finally, the DB-DDI provided satisfactory results with fewer material data points than there are integration points in the structure. In particular, the reference loading curve were well-captured by the data-driven solution.

In the last part of this chapter, we analyzed the data-driven results for themselves, without anymore reference to the classical solution. When we are able to use raw material data, there will not be any reference solution to compare with. We proposed tools to analyze and compare the data-driven results alone. In particular, four indicators were investigated: the convergence of the objective function with the size of the material database, the local Cauchy stress states in the structure, the mechanical analysis of the local strain states and the discrepancy to isotropy. With respect to these four indicators, the DB-DENSE material database again out-performs the other ones, especially thanks to its very large number of points. However, we found that only a portion of these data points were used in the simulation: this indicates that the mechanical states of a given BVP only cover a limited region of the phase space. Therefore, the number of material data points is not the most relevant quantity: their arrangement in phase space and their diversity in terms of modes-of-distortion and orientation angles was proved to have more impact on the quality of the results than the mere number of samples in the data set. That is why the DB-DDI offered satisfying outcome: it consisted in our case of an importance sampling of the relevant strain-stress states that was expected to take place in the structure.

With the work presented in this chapter, we are now able to generate as dense and as rich as possible material databases and to analyze the data-driven results with mechanically meaningful tools. The next chapter is then focused on the application of these contributions to a given three-dimensional problem.

Chapter V

Application to a three-dimensional problem



Contents

1 Introduction

This short chapter is dedicated to the application of the DDCM solvers developed in Chapter III to a large strain three-dimensional problem. Moreover, the methods presented in Chapter IV are used to generate synthetic material databases.

The chapter is organized as follows: the mechanical problem is described in Section 2.1 whereas the generation of the material databases are detailed in Section 2.2. The results for the Lagrangian and nominal formulations are presented in Section 3 and discussed in Section 4.

2 Methods

2.1 Mechanical problem

We study the torsion of a T-shaped column, inspired from the example of Bonet et al. (2016, Chapter 10, page 310). Contrary to their exercise, the simulation is here displacement-driven: the handling of pressure follower load by DDCM solvers requires extra derivation that we did not perform. The geometry and the mesh of the problem are given in Fig. V.1. The mesh consists of N =925 nodes and 576 hexahedral elements of average size 0.25^3 (without units). Each of them containing 8 integration points, the total number of integration points in the mesh is $M = 4608$. The boundary conditions are applied to the surfaces highlighted in gray in Fig. V.1. The bottom surface is fixed ($\boldsymbol{u}_D = \boldsymbol{0}$) while the lateral surfaces on the top part are subject to displacements which engender a rotation of angle $\theta = 360°$ around the Z-axis ($X = Y = 0$):

$$\boldsymbol{u}_D\left(\boldsymbol{X}, X_0, Y_0, \theta\right) = \begin{pmatrix} (X - X_0)\cos\theta - (Y - Y_0)\sin\theta - (X - X_0) \\ (X - X_0)\sin\theta + (Y - Y_0)\cos\theta - (Y - Y_0) \\ 0 \end{pmatrix}, \qquad \text{(V.1)}$$

with $(X_0, Y_0) = (0, 0)$ the center of rotation in the (X, Y) plane.

In the simulations, the nodal displacements are prescribed incrementally: at each pseudo-time step $t \in (0, 1]$, $\boldsymbol{u}_D(t) = \boldsymbol{u}_D(\boldsymbol{X}, X_0, Y_0, t \times \theta)$. We used 45 steps, which corresponds to an increment of $\Delta\theta = 3°$.

2.2 Material databases

2.2.1 Challenges in three dimensions

Until now, we only performed two-dimensional DDCM simulations. The local phase space was then of dimension $3+3 = 6$ for the Lagrangian formulation and $4+4 = 8$ for the nominal formulation. When considering three-dimensional problems, the dimensionality of the local phase space is doubled: it is now $6+6 = 12$ and $8+8 = 19$ for each formulation respectively. The difficulty in increasing the dimensionality of the local phase space is double: (i) definining distances and searching in high-dimensional spaces are complex problems, and (ii) to generate a synthetic material

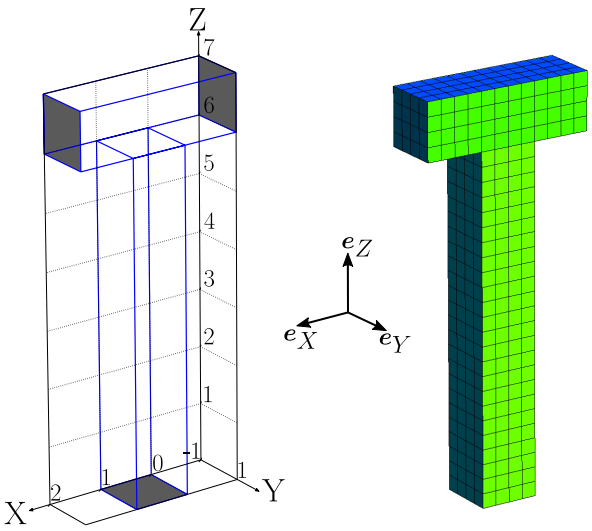


Figure V.1 – Geometry and meshes of the problem. Lengths have no dimension.

database, it becomes necessary to sample a high dimensional strain space. DDCM in 3D is thus exposed to the curse of dimensionality: all data points appear to be sparse, which particularly complicates the nearest neighbor search for the optimal material states (Marimont and Shapiro, 1979).

In addition, we would like to compare the data-driven solutions achieved from both finite strain formulations. The solutions must then be computed with comparable material databases: both strain-stress couples $(\boldsymbol{E}, \boldsymbol{S})$ and $(\boldsymbol{F}, \boldsymbol{P})$ must represent the same material response, *i.e.* sample comparable regions of their respective local phase space.

We capitalize on previous work to *first* generate a Lagrangian material database

$$\mathcal{D}_{\text{loc}}^{\text{Lag}} = \left\{ (\boldsymbol{E}_i, \boldsymbol{S}_i) \in \mathbb{R}_{\text{sym}}^{3\times3} \times \mathbb{R}_{\text{sym}}^{3\times3} \mid i \in [1\,..\,m] \right\}. \tag{V.2}$$

We use the dense sampling method in three dimensions, as proposed by Kunc and Fritzen (2019a) and presented in Chapter IV.

The nominal material database

$$\mathcal{D}_{\text{loc}}^{\text{nom}} = \left\{ (\boldsymbol{F}_i, \boldsymbol{P}_i) \in \mathbb{R}^{3\times3} \times \mathbb{R}^{3\times3} \mid i \in [1\,..\,m] \right\} \tag{V.3}$$

is then obtained from its Lagrangian counterpart $\mathcal{D}_{\text{loc}}^{\text{Lag}}$ and the relationship between the strain-stress pairs:

$$\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{F}^T\boldsymbol{F} - \boldsymbol{I}\right) \tag{V.4}$$

$$\boldsymbol{S} = \boldsymbol{F}^{-1}\boldsymbol{P}. \tag{V.5}$$

2.2.2 Lagrangian formulation

We apply the original dense sampling method of Kunc and Fritzen (2019a) presented in Chapter IV. As a reminder, it consists in sampling the space of Lagrangian stretch tensors $\boldsymbol{U}$ with a strategy based on the DDMS of the stretch tensor and an amplitude-direction split of the coordinates of the deviatoric Hencky strain with respect to the basis of traceless 3×3 matrices:

$$\boldsymbol{U} \in \left\{ \left(J^{(i)}\right)^{1/3} \exp\left(\beta^{(j)} \sum_{k=1}^{5} \left[a^{(l)}\right]_k Y^{(k)}\right) \right\}_{i,j,l=1}^{i=N_{\text{det}},j=N_{\text{amp}},l=N_{\text{dir}}} \subset \mathcal{U},$$

with N_{det} the number of sampled Jacobians J, N_{amp} the number of sampled deviatoric amplitudes β (or magnitudes-of-distortion $K_2 = \beta$), and N_{dir} the number of directions $a \in \mathbb{R}^5$ sampled on the unit sphere of $\mathbb{R}^5$. Each quantity is sampled regularly with a specific metric, that was motivated by the analysis of the FE reference solution:

- the Jacobians J are obtained from a regular discretization of a Laplace probability density function (PDF) defined as

$$\text{pdf}_L(x, x_0, s) = \frac{1}{2} \exp\left(-\left|\frac{x - x_0}{s}\right|\right). \tag{V.6}$$

  That way, the samples are concentrated around the central value of $x_0 = 1$ rather than evenly spaced between J_{min} and J_{max} (see Fig. V.2(a)).

- following a similar procedure, the deviatoric amplitudes β are obtained from a regular sampling of the upper half of a generalized normal PDF, defined for $x \geq 0$ as

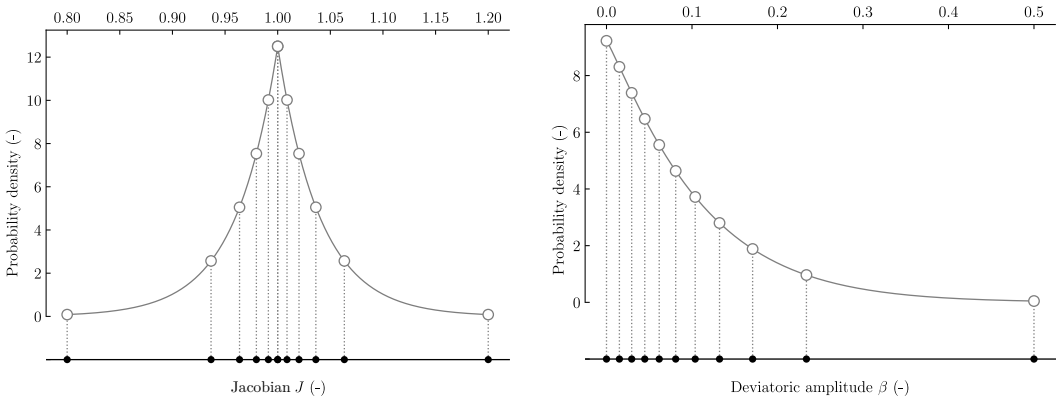
$$\text{pdf}_N(x, s) = \frac{\alpha}{\Gamma(1/\alpha)} \exp\left(-\left(\frac{x}{s}\right)^{\alpha}\right), \tag{V.7}$$

  with Γ the gamma function, $\alpha = 1.12$ and $s = 0.113$ the shape and scale parameters. The resulting points are shown in Fig. V.2(b) for $N_{\text{amp}} = 10$ and $\beta_{\text{max}} = 0.5$.

- the directions a are sampled on the unit sphere of $\mathbb{R}^5$ with the method used by Kunc and Fritzen (2019a). It has been developed by the authors in Kunc and Fritzen (2019b) and provided as an open source Matlab code, available at Fritzen and Kunc (2018a). It consists of "distributed points on the hypersphere such that user-defined energy is minimized". A computed example is provided in the GitHub repository for $N_{\text{dir}} = 512$. To generate larger samples, we used the default parameters of the GUI, *i.e.* the unconstrained minimization of the energy associated to the LOG-kernel, defined as

$$k_{\text{LOG}}(x, y) = \|x - y\| \left(\ln \frac{\|x - y\|}{2} - 1\right) + 2, \tag{V.8}$$

  where ln denotes the natural logarithm. For more details on this approach, the reader is referred to Kunc and Fritzen (2019b).



(a) (b)

Figure V.2 – Sampling procedure for the Jacobians J (a) and the deviatoric amplitudes β (b). The points are obtained from the projection of equi-distant probability values on the horizontal axis.

Once we generated $m = N_{\text{det}} \times N_{\text{amp}} \times N_{\text{dir}}$ stretch tensors $\boldsymbol{U}$, the Lagrangian material databases $\mathcal{D}_{\text{Lag}}$ is obtained from (i) the relation between $\boldsymbol{U}$ and the Green-Lagrange strain tensor $\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{U}^2 - \boldsymbol{I}\right)$, and (ii) a constitutive model $\boldsymbol{S} = \check{\boldsymbol{S}}(\boldsymbol{C} = \boldsymbol{U}^2)$, as

$$\mathcal{D}_{\text{loc}}^{\text{Lag}} = \left\{ \left(\boldsymbol{E}^{(i,j,l)}, \boldsymbol{S}^{(i,j,l)} \right) \mid i \in [1\,..\,N_{\text{det}}],\, j \in [1\,..\,N_{\text{amp}}],\, l \in [1\,..\,N_{\text{dir}}] \right\}. \quad \text{(V.9)}$$

As in Bonet et al. (2016), we consider a three-dimensional compressible neo-Hookean model, defined as

$$\check{\boldsymbol{S}}(\boldsymbol{C}) = \mu(\boldsymbol{I} - \boldsymbol{C}^{-1}) + \lambda \ln(J)\boldsymbol{C}^{-1}, \quad \text{(V.10)}$$

where λ, μ are material parameters. The parameters are set to $\lambda = \mu = 100$ (without units), corresponding to a Young's modulus of 250 and a Poisson's ratio of 0.25.

The material database used in the Lagrangian computations presented in this chapter was generated with the following parameters:

- $N_{\text{det}} = 49$
- $N_{\text{amp}} = 100$
- $N_{\text{dir}} = 2048$

which yielded $m = 10\,035\,200$ material data points.

2.2.3 Nominal formulation

The database for the nominal formulation is obtained from its Lagrangian counterpart as follows. We first re-write the relations Eq. (V.4) as

$$\boldsymbol{F} = \boldsymbol{R}\boldsymbol{U} \quad \text{with } \boldsymbol{U}^2 = 2\boldsymbol{E} + \boldsymbol{I} \quad \text{(V.11)}$$

$$\boldsymbol{P} = \boldsymbol{F}\boldsymbol{S} = \boldsymbol{R}\boldsymbol{U}\boldsymbol{S}, \quad \text{(V.12)}$$

with $\boldsymbol{R} \in SO(3)$ a rotation tensor in the Special Orthogonal group in three dimensions $SO(3) = \left\{R \in \mathbb{R}^{3\times3} \mid R^T R = RR^T = I, \det R = 1\right\}$. We must then additionally sample $SO(3)$ to obtain a nominal material database:

$$\mathcal{D}_{\text{loc}}^{\text{nom}} = \mathcal{R} \times \mathcal{D}_{\text{loc}}^{\text{Lag}}, \tag{V.13}$$

with $\mathcal{R} \in SO(3)$ a sampling set of rotation tensors, and $\mathcal{D}_{\text{loc}}^{\text{Lag}}$ as in Eq. (V.9). We denote N_R the number of elements in $\mathcal{R}$. In order to fully represent any rotation $\boldsymbol{R}$, N_R must be large. The number of points in the nominal material database $N_R \times m$ can then rapidly become huge. However, it is unlikely that all of the Lagrangian data points in $\mathcal{D}_{\text{loc}}^{\text{Lag}}$ are relevant to the computational problem presented in Section 2.1, since the dense sampling method spans (as uniformly as possible) *all* possible magnitudes and modes-of-distortion. Indeed, we showed in the previous chapter that few data points were actually selected as material states during the simulation for the two-dimensional problem, and a similar behavior is expected here. Hence, we chose for this first attempt at performing three-dimensional nominal DDCM to reduce the size of the database with *a priori* knowledge of the solution. We use a first Lagrangian computation as a way to select the relevant data points: all strain-stress pairs that are selected at least once as a material state, within any local data-assignment iteration k of any load step t are collected into a reduced database[1]

$$\mathcal{D}_{\text{loc}}^{\text{Lagred}} = \left\{ \left\{ (\boldsymbol{E}_{\text{ie}}^*, \boldsymbol{S}_{\text{ie}}^*)^{(t,k)} \right\}_{e=1}^{M} \mid t \in [1\,..\,N_{\text{steps}}],\, k \in [1\,..\,k_{\max}^t] \right\}, \tag{V.14}$$

where M is the number of integration points in the mesh, N_{steps} is the number of pseudo-time steps, $k_{\max}^t$ is the number of iterations conducted in step t. Then, we generate the nominal database as the product set

$$\mathcal{D}_{\text{loc}}^{\text{nom}} = \mathcal{R} \times \mathcal{D}_{\text{loc}}^{\text{Lagred}}. \tag{V.15}$$

It thus remains to generate a set $\mathcal{R}$ of 3×3 rotation matrices $R \in SO(3)$, which are the matrix representations of the rotation tensors $\boldsymbol{R}$ in the canonical basis $\mathcal{B} = (\boldsymbol{e}_X, \boldsymbol{e}_Y, \boldsymbol{e}_Z)$. Several parametrization of the special orthogonal group $SO(3)$ are available (Euler angles, quaternions, *etc.*). The most suitable representation is the axis-angle parametrization as it provides an (almost) one-to-one relationship between an axis-angle pair $(\hat{\boldsymbol{e}}, \phi) \in \mathbb{R}^3 \times \mathbb{R}$ and a rotation matrix $R \in SO(3)$. Hence, the sampling strategy is pretty straightforward. This relationship is given by the Rodrigue's formula as

$$R(\hat{\boldsymbol{e}}, \phi) = \cos(\phi) I + \sin(\phi) W_{\hat{\boldsymbol{e}}} + (1 - \cos(\phi))(\hat{\boldsymbol{e}} \otimes \hat{\boldsymbol{e}}), \tag{V.16}$$

where the rotation vector $\hat{\boldsymbol{e}} \in \mathbb{R}^3$ is a unit vector ($\|\hat{\boldsymbol{e}}\| = 1$) giving the axis of rotation, ϕ is the amount of rotation (in radians) around this axis, and $W_{\hat{\boldsymbol{e}}}$ is the

[1] This step may be regarded as another emulation of what would result from a three-dimensional DDI computation, *i.e.* as an importance sampling of the material database.

cross product matrix of $\hat{\boldsymbol{e}}$ such that $W_{\hat{\boldsymbol{e}}}\,\boldsymbol{x} = \hat{\boldsymbol{e}} \times \boldsymbol{x}$ for any vector $\boldsymbol{x} \in \mathbb{R}^3$. Therefore, a product sampling of three-dimensional unit vectors and angles readily gives a set $\mathcal{R}$ of rotation matrices as

$$\mathcal{R} = \left\{ \boldsymbol{R}^{(i,j)} = R(\hat{\boldsymbol{e}}^{(i)}, \phi^{(j)}) \mid i \in [1 \,..\, N_{\text{rot}},\, j \in [1 \,..\, N_{\text{ang}}]] \right\}. \tag{V.17}$$

The set $\left\{\hat{\boldsymbol{e}}^{(i)}\right\}_i^{N_{\text{rot}}}$ of unit vectors is obtained from a sampling of the unit sphere of $\mathbb{R}^3$ using the method of Sloan and Womersley (2004), where N_{rot} is the number of points distributed on the sphere. The set $\left\{\phi^{(j)}\right\}_j^{N_{\text{ang}}}$ of angles consists of N_{ang} evenly spaced values in the interval $(0, \pi]$ such that the distance between two samples is $\Delta\phi = \pi / N_{\text{ang}}$. The case $\phi = 0$ is excluded since the corresponding axis is then not defined: $R(x, \phi = 0) = I$, $\forall x \in \mathbb{R}^3$. To avoid redundant samples, the identity rotation I is added to the set of rotations defined in Eq. (V.17) after the sampling.

Finally, the nominal material database is obtained from the product set

$$\mathcal{D}_{\text{loc}}^{\text{nom}} = \left\{ (\boldsymbol{F} = \boldsymbol{R}\boldsymbol{U},\, \boldsymbol{P} = \boldsymbol{F}\boldsymbol{S}) \mid \boldsymbol{U}^2 = 2\boldsymbol{E} + \boldsymbol{I},\, (\boldsymbol{E}, \boldsymbol{S}) \in \mathcal{D}_{\text{loc}}^{\text{Lagred}},\, \boldsymbol{R} \in \mathcal{R} \right\}. \tag{V.18}$$

2.2.4 Summary

The adopted strategy for generating comparable Lagrangian and nominal material databases is given in Fig. V.3.

2.3 Computation parameters

The data-driven solutions were obtained with the following parameters:

C-parameter the amplitude of the scaling parameter is set to the Young's modulus of the model $E = 250$, for both formulations: $C = E$ in the nominal solver and $\mathbb{C} = C\,\mathbb{I}^{\text{S}}$ in the Lagrangian solver, with $\mathbb{I}^{\text{S}}$ the symmetric fourth-order tensor ($I^{\text{S}}_{ijkl} = \frac{1}{2}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk})$). Note that in that case, $\mathbb{C}$ does not take into account the compressibility of the material (the Poisson's ratio is not involved in the definition).

Initialization The material states are initialized with the "zero" method, in both formulations (see Line 3 in Algorithm III.1 and Line 3 in Algorithm III.3, page 102):all material states are initialized to the zero-deformation state at the first pseudo-time step: $(\boldsymbol{F}^*_{\text{ie}}, \boldsymbol{P}^*_{\text{ie}}) = (\boldsymbol{I}, \boldsymbol{0})$, $\forall e$ and $(\boldsymbol{E}^*_{\text{ie}}, \boldsymbol{S}^*_{\text{ie}}) = (\boldsymbol{0}, \boldsymbol{0})$, $\forall e$.

Augmented Lagrangian parameters the initial penalty parameters c_0^D, c_0^t, c_0^m are respectively set to $10^3 C$, $10^3 C$, C. The initial Lagrange multipliers λ_0, η_0, μ_0 are all set to zero (see Lines 3 and 4 in Algorithm III.2, page 101).

Computer all simulations are performed on a laptop with the following specifications: processor Intel Core i5-6200U CPU @ 2.30 GHz × 4, with 15.5 Gio of RAM.

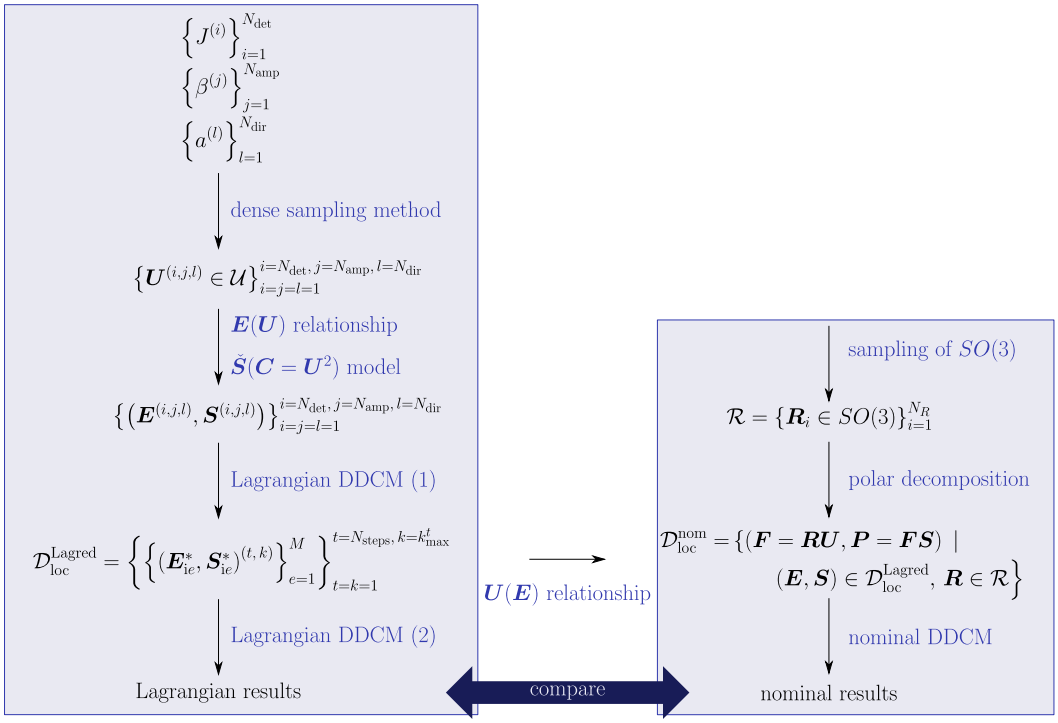


Figure V.3 – Procedure to generate the Lagrangian (left) and nominal (right) material databases.

3 Results

3.1 Nominal results

We did not success in obtaining converged results with the nominal formulation and the material database generated with the procedure listed in Fig. V.3. More investigation should be conducted to understand why. In particular, more care should be taken in defining the initial penalty parameters in order to avoid ill-conditioning as well as to ensure good convergence properties. However, we believe that the dimensionality of the phase space is a probable cause for this failure. To illustrate this claim, we compare the deformed mesh together with the Von Mises norm of the Cauchy stress obtained with both formulations in Fig. V.4. The average element size is twice as large as for the mesh presented in Fig. V.1. The material database is constituted with the FE reference solution only, corresponding to both formulations. The material states are initialized with the "arange" method: the assignments for the first pseudo-time step (only) are initialized to point towards the corresponding FE solution (*c.f.* Chapter III for more details on initialization methods). As can be seen from Fig. V.4, the deformed mesh is not well predicted by the nominal formulation. The ranges of stresses are however quite similar. This example is degenerate: the mesh is too coarse for the problem and the material database is very sparse, however perfectly located in phase space. Nevertheless,

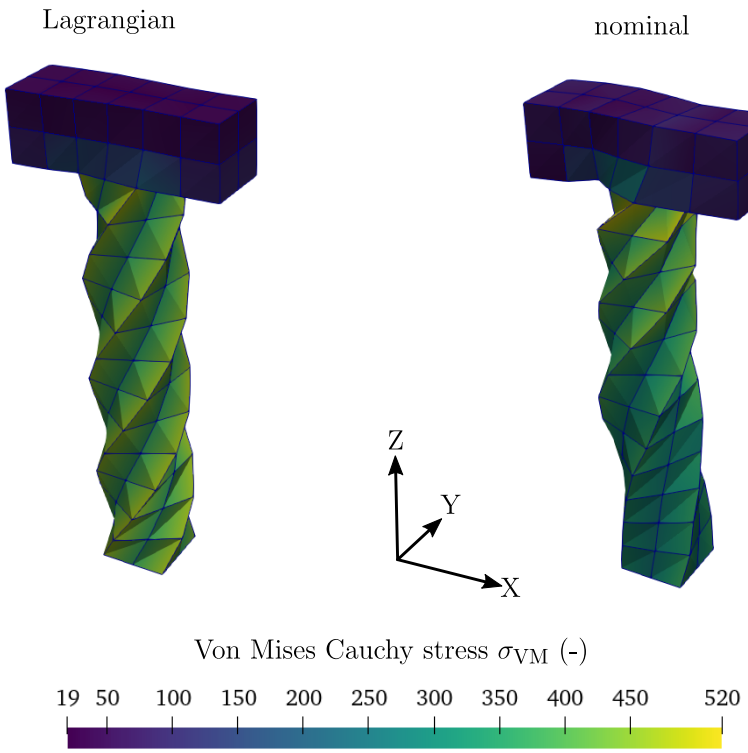


Figure V.4 – Deformed mesh of the Lagrangian (left) and nominal (right) formulations. The material database is the concatenation of the FE reference solution at all integration points on all pseudo-time steps.

the Lagrangian formulation is able to find a good qualitative solution, working in a 2×6-dimensional space. The nominal formulation is not able to find the same solution, with *a priori* the same input. This could be due to larger dimensionality (2×9) of the phase space, as the distance-minimization problem is more complex to solver. Others possible causes for failure include:

- the scaling parameter C in the nominal distance is a scalar, and hence does not take into account the Poisson effect. However, $\mathbb{C} = C\,\mathbb{I}^{\mathrm{SYM}}$ in the Lagrangian distance, which should be equivalent.

- the material data points in the reference nominal database (constituted with the reference solution only) may lack an orbit representation, which would ease the search for optimal material states.

Finally, we can not exclude the possibility of a bug: the solver was developed in the last months of the thesis and has not been fully tested yet. In lack of hindsight on this method, we are not able to comment further the behavior of the nominal formulation. In the following, we then restrict ourselves to presenting and discussing the Lagrangian results only.

3.2 Lagrangian formulation

We now examine the results of the DDCM in the Lagrangian formulation. Unless specified, we analyze the results achieved from the second Lagrangian computation using the reduced database $\mathcal{D}_{\text{loc}}^{\text{Lagred}}$, which consists of only the relevant material data points selected from the first Lagrangian computation (see Fig. V.3). The number of material data points has then reduced from $m = 10\,035\,200$ to $m = 75\,077$. After the second Lagrangian computation, the number of data points that have been used as material states is again reduced, to $m = 22\,170$.

3.2.1 Deformed mesh and resultant

Let us first illustrate the DDCM results with global (structural) indicators. In Fig. V.5, the deformed mesh and the value of the shear strain F_{XY} for different pseudo-time steps are shown. Note that the first snapshot at $\theta = 152°$ in Fig. V.5

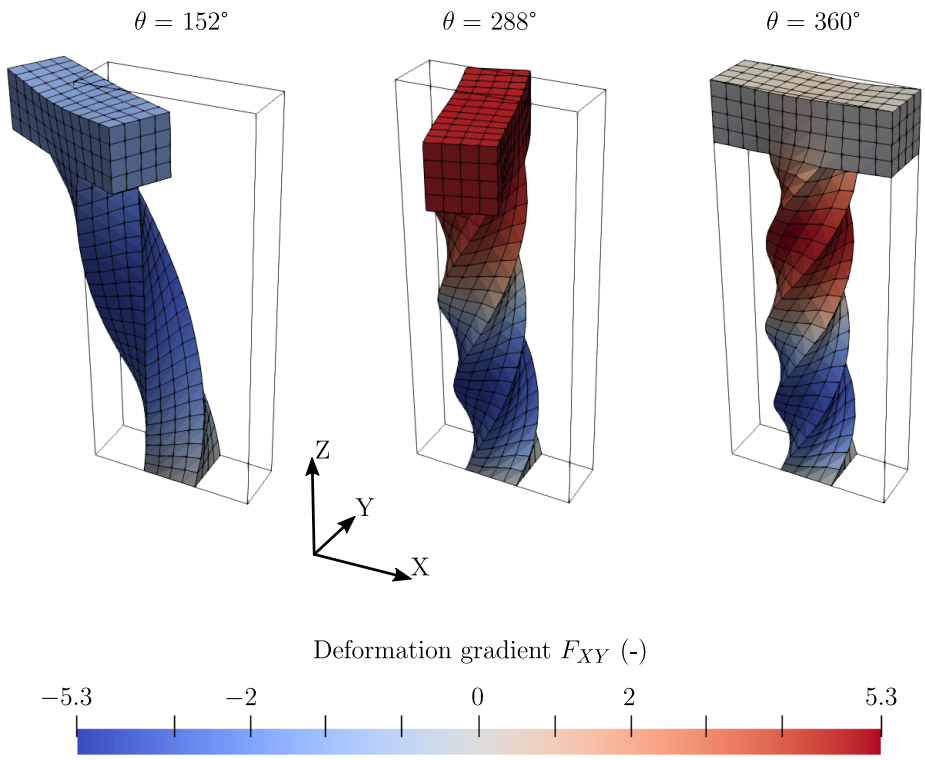


Figure V.5 – Lagrangian results. Deformed mesh and shear F_{XY} for different loading steps.

illustrates that the center of rotation is not the center of the column, as prescribed by the Dirichlet boundary conditions. In addition, the torsion of the bottom part of the column at $\theta = 360°$ is, as expected, independent of the loading path. The

deformed mesh is then qualitatively satisfactory.

Since the bottom surface $Z = 0$ is fixed, we compute the reaction forces and moments on this surface as a post-processing. We refer back to the equilibrium residual $\boldsymbol{R}^\eta$ defined by Eq. (III.39a) in Chapter III. The nodal reaction force $\boldsymbol{t}_a$ at node a of the bottom surface $Z = 0$ is given by

$$\boldsymbol{t}_a = \boldsymbol{R}_a^\eta = \sum_e w_e \, \boldsymbol{F}_e \left(\mathbb{C} : \left(\boldsymbol{F}_e^T \boldsymbol{\nabla}_0 \boldsymbol{\eta} \right) \right) \boldsymbol{B}_{ea} + \sum_e w_e \, \boldsymbol{F}_e \boldsymbol{S}_{\text{ie}}^* \boldsymbol{B}_{ea}, \tag{V.19}$$

(note that $\boldsymbol{T}_a^{\text{ext}}$ from Eq. (III.39a) is $\boldsymbol{0}$ since no external force is applied). Then, the resultant force $\boldsymbol{T}$ is the summation of nodal forces on the $N_{Z=0}$ nodes of surface $Z = 0$:

$$\boldsymbol{T} = \sum_{a=1}^{N_{Z=0}} \boldsymbol{t}_a. \tag{V.20}$$

In addition, the resultant moment $\boldsymbol{M}$ with respect to the center of the column $\boldsymbol{X}_c = [0.5, 0.5, 0]^T$ is computed as the summation of nodal reaction moments:

$$\boldsymbol{M} = \sum_{a=1}^{N_{Z=0}} (\boldsymbol{X}_a - \boldsymbol{X}_c) \times \boldsymbol{t}_a, \tag{V.21}$$

where $\times$ denotes the standard cross product. Evolution of the resultant force and moment with the applied rotation angle θ is shown in Fig. V.6. The results for the first (resp. second) Lagrangian simulation, labeled "DDCM" (resp. "DDCM red."), are represented in solid (resp. dotted-dashed) line. The standard finite element reference solution is also shown in dotted line, for comparison. We can see that the Z-component of the resultant moment consistently increases with the applied rotation. In addition, the resultant forces are maximized when the center of the column is above the center of rotation ($\theta \simeq 180°$), *i.e.* when the structure is also sheared. Note however that the Z-component T_Z of the resultant force is underestimated by DDCM. Finally, the results obtained from the first and second Lagrangian computations are very similar. This indicates that the m =75 077 data points were indeed sufficient to predict the global response of the structure.

These first results demonstrate that it is possible to perform qualitatively consistent three-dimensional data-driven simulations with the Lagrangian formulation.

3.2.2 Results of the optimization

Let us now examine how well the distance-minimization scheme performed. Table V.1 reports the computation times together with the final value of the objective function for the two Lagrangian simulations and the FE reference solution, as a comparison. We can see that the final value of the objective function is twice as large for the second simulation than for the first one. However, the computational times are very similar.

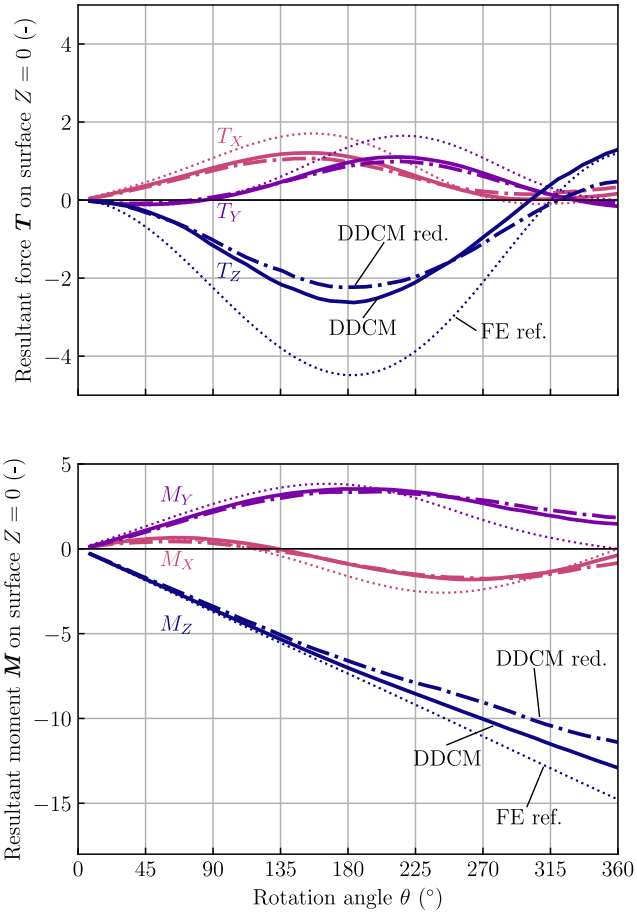


Figure V.6 – Resultant force and moment on the bottom surface $Z = 0$.

Table V.1 – Computation times for the two Lagrangian simulation and the FE reference solution.

Simulation	Computation time	Objective function	Iterations
DDCM (1)	2 h 54 min	3.31 MPa mm^2	340
DDCM (2)	3 h 5 min	6.42 MPa mm^2	390
FE	30 min		

In Fig. V.7, the convergence of the objective function throughout all pseudo-time steps is depicted, together with the number of local data assignments changed between two data-driven iterations. This indicates the number of integration points which get their optimal material states updated from one iteration k to the next (it is the number of elements in the set $\left\{e \in [1 \,..\, M] \mid \; z_{\mathsf{ie}}^{*,(k)} \neq z_{\mathsf{ie}}^{*,(k+1)}\right\}$). The optimization is considered converged either when no changes occur, or when the number of changes falls below 0.1 % of the total number of integration points M. This additional criterion greatly improves computation time as it prevents iterations

from running without much improvement of the objective function. It can be seen

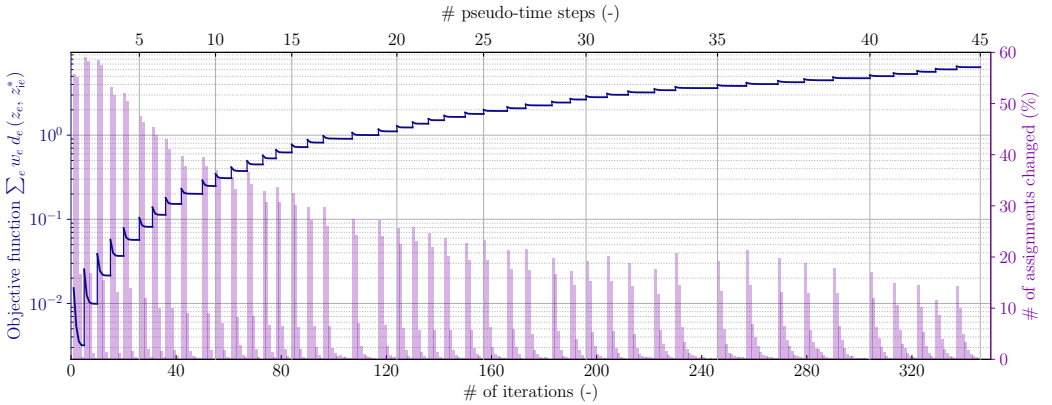


Figure V.7 – Convergence of the objective function during the simulation. The bars indicates the number of material state updates as a percentage of M, the number of integration points. The solid line represents the convergence path throughout all pseudo-time steps (indicated at the top) and all iterations (indicated at the bottom).

from Fig. V.7 that the minimum value of the objective function increases with the number of pseudo-time steps. In addition, the objective function is significantly lowered within a few data-driven iterations for the first pseudo-time steps. Conversely, it is very little improved in the last pseudo-time steps. As the step number increases, the prescribed rotation of the column induces larger strain in the structure (*cf.* Fig. V.5). We can conclude that the optimization problem is harder to solve for large deformations. We recall that the initialization for the displacements and the material states of one given pseudo-time step corresponds to the results obtained at the last (converged) data-driven iteration of the previous step.

4 Discussion

4.1 Recovering isotropy

As in Chapter IV, the data-driven results are discussed based on their ability to recover the isotropy of the material. Note that the discrete material database is indeed generated by an isotropic constitutive model. However, this assumption is not enforced in the data-driven computations. In other words, there can exist a mechanical solution to the data-driven problem which is not perfectly isotropic, while satisfying equilibrium and kinematics, and being close enough to the material database. To measure how far the DDCM (mechanical) solution lies from a perfectly isotropic solution, we consider again the angle α between the principal directions of the Lagrangian strain and stress tensors $\boldsymbol{E}$ and $\boldsymbol{S}$. In the three-dimensional case, it is unsigned and defined as

$$\alpha = \arccos\left(\boldsymbol{p}_I(\boldsymbol{E}) \cdot \boldsymbol{p}_I(\boldsymbol{S})\right), \tag{V.22}$$

where $\boldsymbol{p}_I(\boldsymbol{A})$ is the eigenvector associated with the largest eigenvalue of tensor $\boldsymbol{A} \in \mathbb{R}^{3\times3}$. We refer to α as the isotropy mismatch angle. Note that we restrict α to the interval $[0, \pi]$ by rotating the eigenvectors of $\boldsymbol{S}$ such that $\boldsymbol{p}_I(\boldsymbol{E})$ and $\boldsymbol{p}_I(\boldsymbol{S})$ are in the same quarter of the unit 3D sphere[2]. Two examples are provided in Fig. V.8, where we represent the three eigenvectors of both tensors. The angle α can also be seen as the amount of rotation around the axis defined by $\boldsymbol{p}_I(\boldsymbol{E}) \times \boldsymbol{p}_I(\boldsymbol{S})$, using the counter-clockwise convention.

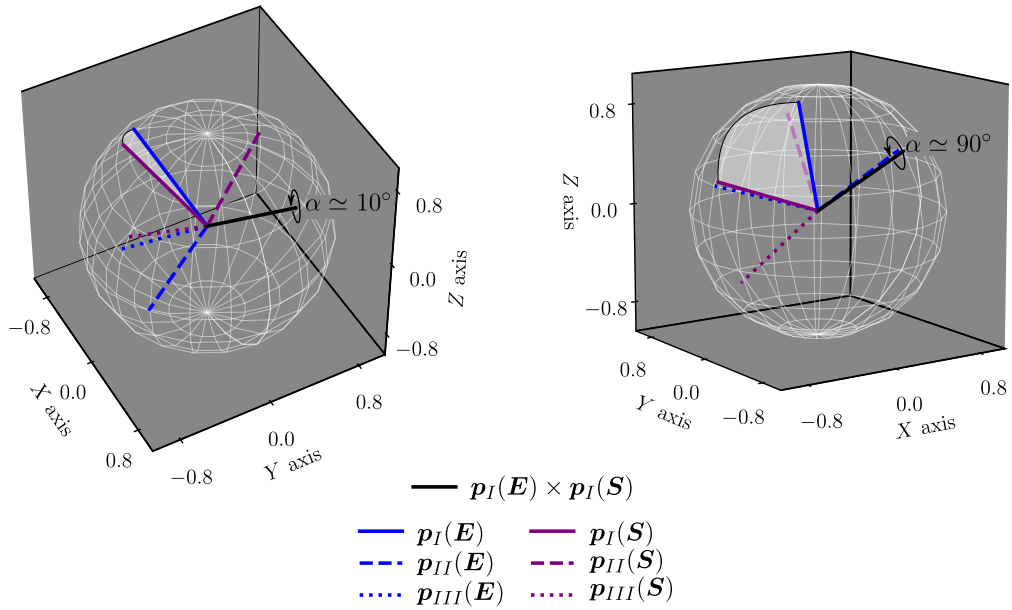


Figure V.8 – Isotropy mismatch angle, defined as the angle of rotation between the first principal directions of the strain and stress tensors $\boldsymbol{p}_I(\boldsymbol{E})$ and $\boldsymbol{p}_I(\boldsymbol{S})$. Left: situation where the angle is about 10°. Right: situation where the angle is about 90°.

In Fig. V.9, histograms of the results achieved from the first and second Lagrangian DDCM computations are presented. Most of the integration points in the mesh have a relatively low isotropy mismatch angle: 60 % of the mechanical states are below 12°. We can see that the second Lagrangian computation behaves slightly worse than the first one. The selection of the material states in the reduced Lagrangian database (see Eq. (V.14)) merely discarded *a priori* useless points, thus delimiting the region of phase space spanned by the problem. However, the density of material data points in this region of interest has not been improved. We can surmise that the isotropy mismatch angle would be lower with a richer database. Nevertheless, the method to enrich the reduced database with appropriate points is an open subject.

[2] $(\boldsymbol{p}_I(\boldsymbol{S}), \boldsymbol{p}_{II}(\boldsymbol{S}), \boldsymbol{p}_{III}(\boldsymbol{S}))$ is transformed into $(-\boldsymbol{p}_I(\boldsymbol{S}), -\boldsymbol{p}_{II}(\boldsymbol{S}), \boldsymbol{p}_{III}(\boldsymbol{S}))$ whenever $\alpha > 90°$, so as to remain a right-handed system.

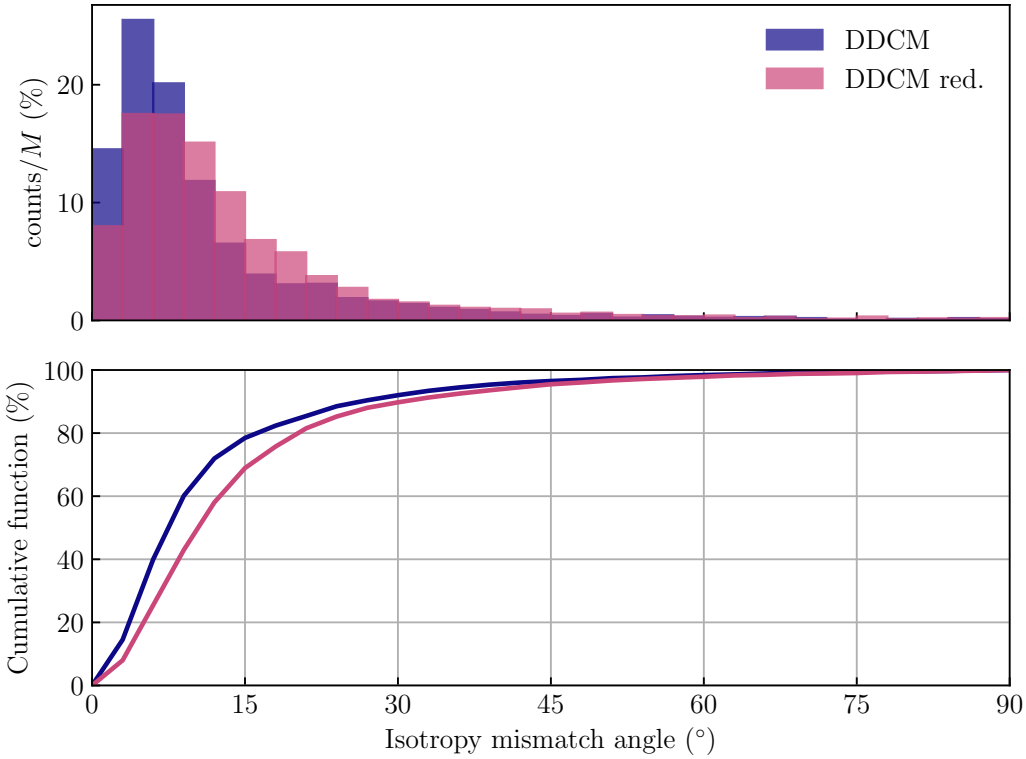


Figure V.9 – Isotropy mismatch angle: misalignment angle between the principal strain and the principal stress of all mechanical states, obtained from the first ("DDCM") and second ("DDCM red.") Lagrangian computations.

As can be seen in Fig. V.9, there is certain amount of integration points where α is close to 90°. We recall that the eigenvectors of any tensor form an orthonormal basis. Hence, the angle between $p_I(\boldsymbol{E})$ and $p_{II}(\boldsymbol{E})$ is 90°. An isotropy mismatch angle of 90° could then indicate that the order of the eigenvectors is not well determined, *i.e.* the first and second eigenvalues are very close to one another. To investigate this assumption, we examine each isotropy mismatch angle in regard with the corresponding principal strain (stress) gap, defined as $\Delta E = E_I - E_{II}$ ($\Delta S = S_I - S_{II}$), where E_I and E_{II} (S_I and S_{II}) are the largest and second largest strain (stress) eigenvalues, as determined by the `numpy.linalg.eigh()` function of the Numpy package in Python. Fig. V.10 confirms that the largest isotropy mismatch angles are found for the lowest principal strain or stress gaps.

To conclude, the isotropy is quite well recovered by the data-driven solution, provided that the material database only consists of 75 007 strain-stress pairs. The few local errors correspond either to small strain or almost equi-biaxial states for which the isotropy mismatch angle may not be well numerically computed.

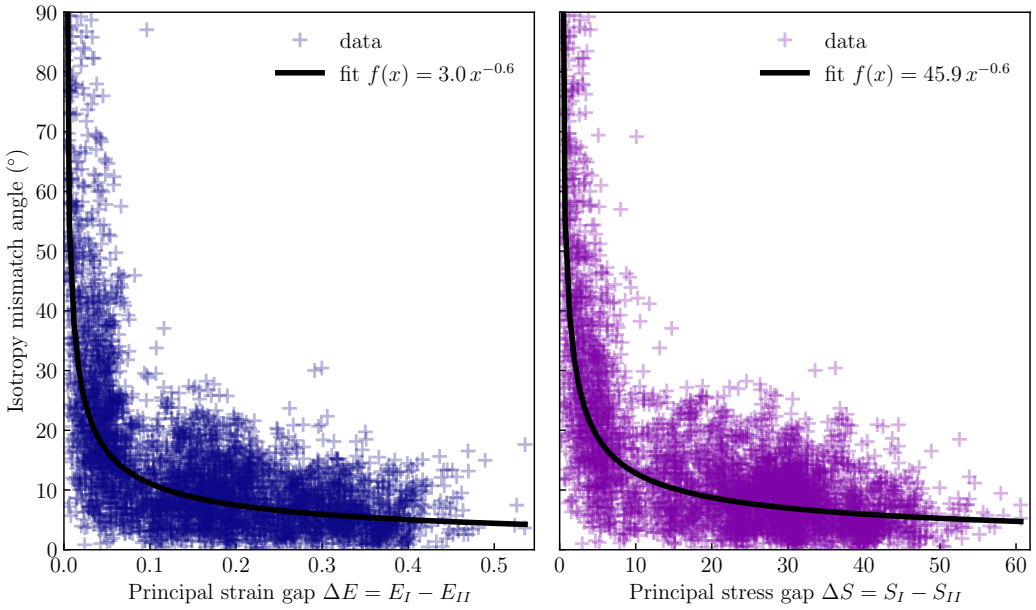


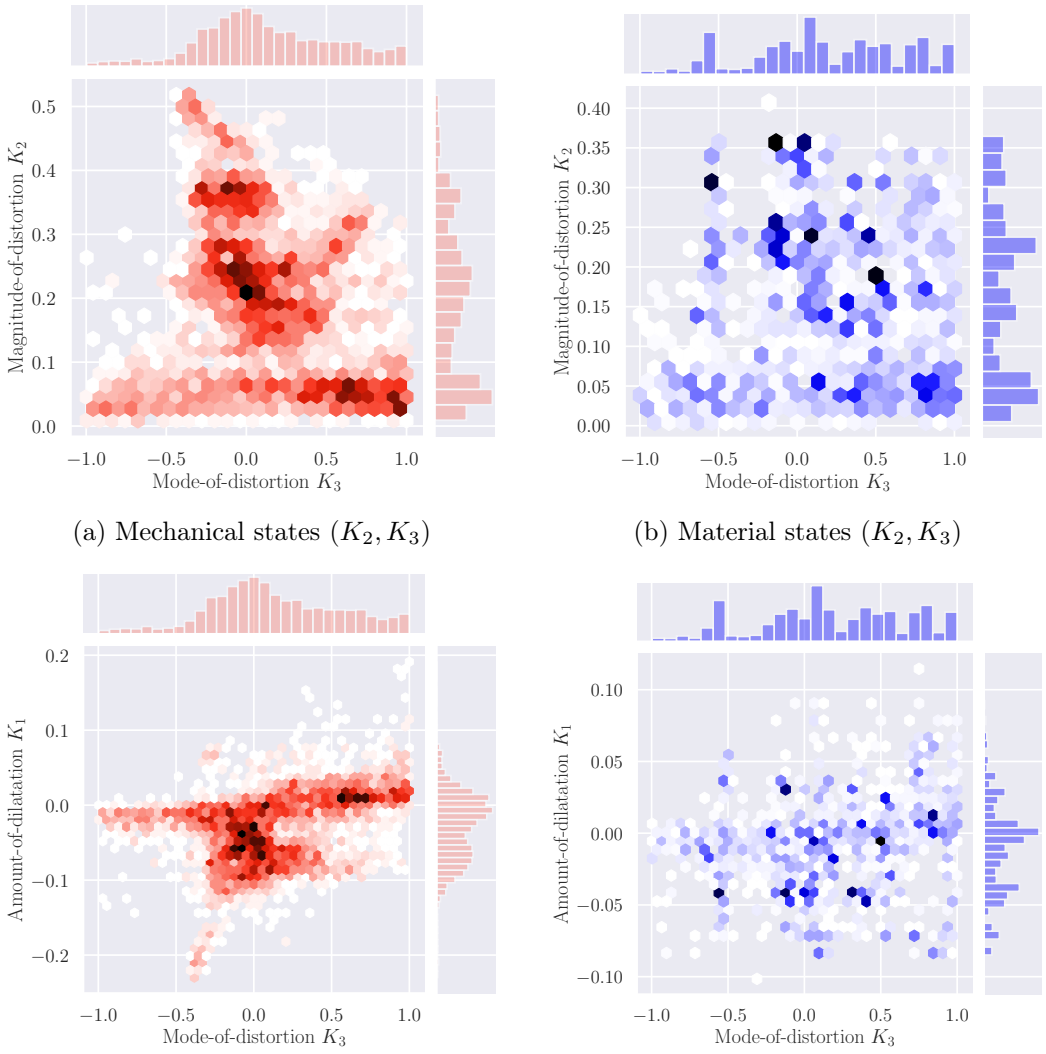
Figure V.10 – Isotropy mismatch angle *vs.* principal strain (left) and stress (right) gap.

4.2 Modes of deformation

As mentioned in Chapter IV, the natural strain invariant basis introduced by Criscione et al. (2000) is an interesting tool to analyze the modes of deformation occurring in a structure. Let us briefly recall the meaning of these invariants. The first invariant K_1 of the Hencky strain tensor $\boldsymbol{Y} = \ln \boldsymbol{U}$ is a measure of the amount-of-dilatation as $K_1 = \ln J$ with $J = \det \boldsymbol{F}$; a negative values stands for contraction while a positive value accounts for dilatation. An isochoric deformation is measured by $K_1 = 0$. The second and third invariants $K_2 \geq 0$ and $K_3 \in [-1, 1]$ respectively accounts for the magnitude- and mode-of-distortion, *e.g.* tension or shear. In Fig. V.11, we represent bi-variate histograms of the (K_1, K_2, K_3) values for the mechanical and material states of the second Lagrangian computation, together with the marginal distributions. The counts in the bins are intentionally not shown to simplify the plots; darker bins indicates concentration of points.

First, note that the distributions of the mechanical and material states, in both the (K_3, K_2) and (K_3, K_1) planes are very different. More precisely, comparing Fig. V.11(a) and Fig. V.11(b), the mechanical states exhibit a larger diversity of magnitudes-of-distortion K_2 than the material states. As for the K_3 invariants, it also seems that more modes-of-distortion are covered by the mechanical states. In addition, it can be seen from Fig. V.11(a) that integration points with rather small strain (lowest values of K_2) undergo almost all possible modes-of-distortion, from equi-biaxial tension ($K_3 = -1$) to uniaxial tension ($K_3 = 1$) through pure shear ($K_3 = 0$).

From the (K_1, K_3) distributions represented in Fig. V.11(c), a majority of the mechanical states are in contraction ($K_1 < 0$), which is expected by the loading conditions and the compressibility of the material. Likewise, it can be seen from the marginal distribution of K_3 of the mechanical states in Fig. V.11(c) that there is a concentration of points around the pure shear state $K_3 = 0$. Comparing Figs. V.11(c) and V.11(d), it is remarkable that some material states lie in an area where no mechanical states sit ($\{K_3 > 0.5\} \cup \{K_1 < -0.1\}$).



(a) Mechanical states (K_2, K_3)

(b) Material states (K_2, K_3)

(c) Mechanical states (K_1, K_3)

(d) Material states (K_1, K_3)

Figure V.11 – Bi-variate distributions of (a)-(b) (K_1, K_3) invariants and (c)-(d) (K_1, K_3) invariants of the mechanical states (in red) and the material states (in blue) respectively. Darker colors indicate concentration of points in this area. Marginal histograms are shown for each variable.

5 Conclusion

This chapter was intended to assess the data-driven finite strain solvers in a complex three-dimensional problem. To this end, we presented two methods to generate synthetic material databases which can be used in both the Lagrangian and nominal formulations, while providing comparable results. The former merely consisted in using the dense sampling method extensively presented in Chapter IV. The latter relied on the product set of rotation-free material data points $(\boldsymbol{U}, \boldsymbol{S})$ with rotation matrices $\boldsymbol{R}$. In both cases, mechanical problems in three-dimensions are revealed more challenging that the two-dimensional examples presented so far: the corresponding local phase space of the data-driven BVP is of higher dimension, which considerably increases the number of points needed to generate dense databases. This difficulty is even more acute in the nominal formulation as the local phase space consists of non-symmetrical tensors.

The database for the Lagrangian formulation then comprised $m = 10\,035\,200$ points. It was successfully used in a first Lagrangian computation, which yielded satisfactory results after a quick qualitative analysis. More interestingly, it turned out that only $m = 75\,077$ of the material data points were actually selected in the computation. This was expected: the dense sampling method spans all possible deformation modes in all possibles directions while the studied mechanical problem is necessarily restricted to certain deformation paths. Hence, we used these data points directly as the material database in a second Lagrangian simulation, and as a basis to generate the nominal material database. The second Lagrangian simulation also yielded satisfactory results. Unfortunately, we could not reach convergence with the nominal solver. A sanity check with a coarser mesh and the FE reference solution as the database showed that the three-dimensional case is probably more difficult to handle with the nominal formulation. The reasons why the simulation did not succeed are still unclear.

On the opposite, the Lagrangian solution compared well enough with the FE reference solution. In particular, we showed that the deformed mesh and the resultant force and moment were qualitatively very well captured. Investigating the behavior of the solver, we showed that the optimization problem is harder to solve at large strain, as the objective function decreases more slowly with the iterations. Finally, we discussed the ability of the data-driven solver to recover the isotropy of the material, while not being enforced in the simulation. If the results are not as satisfying as the one presented in the previous chapter, we argue that they are encouraging: one should bear in mind that the density of the material database was much lower herein and the distance-minimizing problem is more complex in three dimensions. In addition, the study of the amplitudes and modes of deformation with the natural strain invariant basis offered an understanding of the material and mechanical strain states. Note that further work is needed to develop a dual representation of the stress tensors to analyze in like manner the stress results. This part probably brought about more questions than answers. For example, how are mechanical and material states related in terms of mode and amplitude of deforma-

tion (dilatation and distortion-wise)? Nevertheless, it showed that the data-driven results were consistent with the material properties (compressibility in the broad sense) and the loading conditions (a rotation which induces shear).

This study is an encouraging first step towards realistic large strain three-dimensional data-driven computations, at least for the Lagrangian formulation. To the best of our knowledge, this work is the first truly three-dimensional data-driven simulation performed with DDCM (finite strain) solvers: the linear elastic example presented in Kirchdoerfer and Ortiz (2016) actually used a limited material database, encoding plane-stress conditions. Future work should include a systematic study of the influence of the different parameters, *e.g.* the number of samples N_{dir} or N_{det}, or the $\mathbb{C}$ parameter. In addition, the adaptation of the nominal formulation to three-dimensional problems should be elucidated.

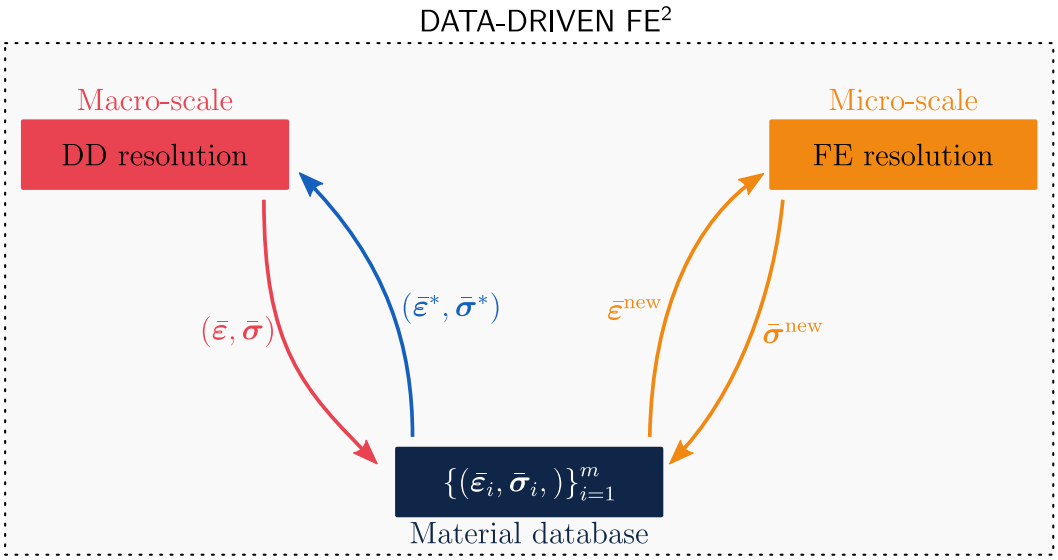
To conclude, let us emphasize that one crucial question remains unsolved in the two and three-dimensional examples we presented in this second part of the thesis. The DDCM paradigm provides a mechanically admissible response which is as close as possible to what is known of the material behavior. This knowledge can be, as we have seen in Chapter IV and herein, insufficient for the BVP under consideration. Hence, one would want to *complete* this knowledge with additional data. That is what we tried to do when adding more directions or more amplitudes to the material database. But it would be preferable to find a systematic way of enriching the material database. We try and address this question in the last part of the thesis, in Chapter VI, for the particular case of heterogeneous materials.

Part C

Towards data-driven multiscale numerical schemes

Chapter VI

A first attempt to apply the data-driven approach to multiscale simulation



Contents

1 Introduction

So far, the material databases we used were artificial: they were generated by sampling a constitutive model, later discarded in the data-driven computations. In this manner, we assessed the data-driven solvers by comparing their results with the classical FE reference solution. However, the DDCM paradigm aims at using *raw* material data, thus by-passing constitutive models. As aforementioned, material data points can be produced in two manners. On the one hand, collecting raw *experimental* data is a true challenge. We refer back to Chapters II and IV for the achievements and difficulties in producing experimental material databases suitable for DDCM. On the other hand, numerical simulations at finer scales can provide the overall behavior of heterogeneous materials. There, the constitutive models of the microscopic phases (fine scale) are assumed simple enough and reliable. In this chapter, we explore the latter approach and investigate how the DDCM can be successfully applied to the simulation of structures made of heterogeneous materials. In this first attempt, we restrict our attention to small strain elasticity and use the original solver of Kirchdoerfer and Ortiz (2016).

1.1 Computational homogenization: principles and challenges

We first briefly recall the basic principles of the homogenization of heterogeneous materials and the challenges of computational homogenization. Multiscale material modeling has been a vivid area of research for several decades, which we do not pretend to exhaustively report herein. Instead, we refer to some recent reviews as the one of Zaoui (2002); Kanouté et al. (2009); Geers et al. (2010); Matouš et al. (2017) or standard books such as Bornert et al. (2001); Zohdi and Wriggers (2008).

Heterogeneous or composite materials refer to materials mixing distinct microscopic phases, with different mechanical properties. We consider composites in which the size of the heterogeneities is much smaller than the size of the engineering structure, such as reinforced polymers. Then, two scales are distinguished: at the macroscale, the material is virtually considered as homogeneous; at the microscale, the microscopic phases are modeled by the constitutive model of their constituent. The effective or overall mechanical response then depends on the mechanical properties of the constituents, their geometry, volume fraction and spatial distribution in the microstructure. The effective response is derived by solving a specific BVP on a RVE of the microstructure, associated to a material point of the macrostructure, as illustrated in Fig. VI.1. As defined by Hill (1963), the RVE is, at the same time, "sufficiently large" to be statistically representative of the microstructure and "sufficiently small" with respect to the characteristic length of the macrostructure (separation of scales). The macroscopic strain and stress responses are given by volume averages on the RVE Ω, of volume V:

$$\bar{\boldsymbol{\varepsilon}} = \langle \boldsymbol{\varepsilon} \rangle \,; \quad \bar{\boldsymbol{\sigma}} = \langle \boldsymbol{\sigma} \rangle \,, \tag{VI.1}$$

with $\langle \cdot \rangle = 1/V \int_\Omega (\cdot)\, \mathrm{d}\boldsymbol{x}$. In Eq. (VI.1), the strain and stress fields $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma}$ at the

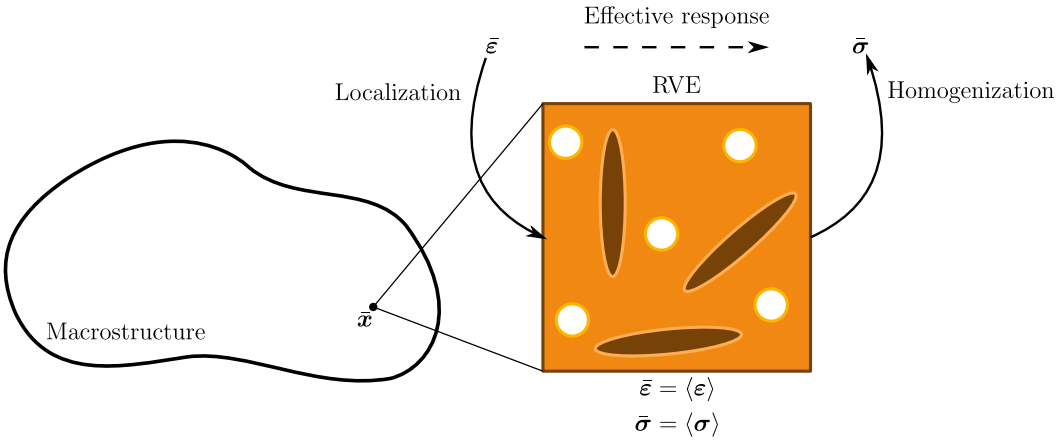


Figure VI.1 – Effective constitutive relation of a composite, obtained from the resolution of a BVP on the RVE of the microstructure. Inspired from Brassart (2011).

microscale are solution of a local BVP on the RVE:

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(\boldsymbol{\nabla} \boldsymbol{u} + \boldsymbol{\nabla}^T \boldsymbol{u}\right), \qquad \forall \boldsymbol{x} \in \Omega \tag{VI.2a}$$

$$\operatorname{div} \boldsymbol{\sigma}(\boldsymbol{x}) = \boldsymbol{0}, \qquad \forall \boldsymbol{x} \in \Omega, \tag{VI.2b}$$

$$\boldsymbol{\sigma}(\boldsymbol{x}) = \sum_{r=1}^{N_r} \chi^{(r)} \check{\boldsymbol{\sigma}}^{(r)}(\boldsymbol{\varepsilon}(\boldsymbol{x})), \qquad \forall \boldsymbol{x} \in \Omega, \tag{VI.2c}$$

where $\check{\boldsymbol{\sigma}}^{(r)}(\bar{\boldsymbol{\varepsilon}})$ is the local constitutive relation of microscopic phase r, N_r is the number of distinct microscopic phases in the composite, and the characteristic function $\chi^{(r)}$ is defined as

$$\chi^{(r)} = \begin{cases} 1 & \text{if } \boldsymbol{x} \in \Omega^{(r)} \\ 0 & \text{otherwise,} \end{cases} \tag{VI.3}$$

with $\Omega^{(r)}$ the domain occupied by phase r. The associated boundary conditions are usually chosen among three kinds, namely:

- linear displacement boundary conditions

$$\boldsymbol{u}(\boldsymbol{x}) = \bar{\boldsymbol{\varepsilon}} \cdot \boldsymbol{x}, \ \forall \boldsymbol{x} \in \partial\Omega \tag{VI.4}$$

  which ensure that $\bar{\boldsymbol{\varepsilon}} = \langle \boldsymbol{\varepsilon} \rangle$;

- uniform traction boundary conditions

$$\boldsymbol{t}(\boldsymbol{x}) = \bar{\boldsymbol{\sigma}} \cdot \boldsymbol{n}(\boldsymbol{x}), \ \forall \boldsymbol{x} \in \partial\Omega \tag{VI.5}$$

  which ensure that $\bar{\boldsymbol{\sigma}} = \langle \boldsymbol{\sigma} \rangle$, with $\boldsymbol{n}(\boldsymbol{x})$ the outward normal to the RVE boundary;

- periodic boundary conditions

$$\boldsymbol{u}(\boldsymbol{x}^+) - \boldsymbol{u}(\boldsymbol{x}^-) = \bar{\boldsymbol{\varepsilon}} \cdot (\boldsymbol{x}^+ - \boldsymbol{x}^-), \qquad \forall \boldsymbol{x} \in \partial\Omega, \tag{VI.6a}$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n}(\boldsymbol{x}^+) = -\boldsymbol{\sigma} \cdot \boldsymbol{n}(\boldsymbol{x}^-), \qquad \forall \boldsymbol{x} \in \partial\Omega, \tag{VI.6b}$$

  which ensure that $\bar{\boldsymbol{\varepsilon}} = \langle \boldsymbol{\varepsilon} \rangle$, where $\boldsymbol{x}^+$ and $\boldsymbol{x}^-$ denotes opposite points on the RVE boundary $\partial\Omega$.

In the following, we use the linear displacement boundary conditions, as they are the easiest ones to implement in a FE code. The effective constitutive relation between the macroscopic stress $\bar{\boldsymbol{\sigma}}$ and the macroscopic strain $\bar{\boldsymbol{\varepsilon}}$ is no longer explicit but results from a localization step (application of the boundary conditions) and a homogenization step (volume average), as illustrated in Fig. VI.1. Finally, a key result of homogenization theory is the Hill-Mandel lemma, which states that the macroscopic work is equal to the average microscopic work (Hill, 1967). Considering an equilibrated stress field $\boldsymbol{\sigma}$ and a compatible strain field $\boldsymbol{\varepsilon}$, if $\boldsymbol{\sigma}$ satisfies the uniform boundary conditions Eq. (VI.5) or if $\boldsymbol{\varepsilon}$ satisfies the linear displacement boundary conditions Eq. (VI.4), then

$$\langle \boldsymbol{\sigma} : \boldsymbol{\varepsilon} \rangle = \langle \boldsymbol{\sigma} \rangle : \langle \boldsymbol{\varepsilon} \rangle = \bar{\boldsymbol{\sigma}} : \bar{\boldsymbol{\varepsilon}}. \tag{VI.7}$$

Roughly speaking, the research concerned with homogenization of composite materials in solid mechanics can be divided into two main fields:

1. On the one hand, continuum micromechanics aims at extracting macroscopic constitutive equations based on continuum mechanics at the level of microstructural heterogeneities. In particular, mean fields approaches determine the effective behavior of the composite from the statistical moments of micromechanical fields. In linear elasticity, most models build upon the problem of an isolated inclusion embedded in an infinite medium and subject to a far-field strain, for which Eshelby and Peierls (1957) provided an analytical solution. The extension to nonlinear elasticity is made either through the linearization of the strain-stress relation (*e.g.* Hill (1965)) or by means of variational approaches which provide rigorous bounds and estimates for hyperelastic materials (see Zaoui (2002) for an extensive overview). In this field, the microstructure is simplified and specific assumptions describe the interactions between constituents.

2. On the other hand, in numerical approaches, the effective properties are computed thanks to equilibrium simulations on a RVE of the microstructure. Then, fewer assumptions are made on the spatial arrangement of the heterogeneities or the relationship between them. The "real" microstructure can even be directly used as input to the computation, such as in image-based FE methods (Terada et al., 1997) or using Fast Fourrier Transform (FFT)-based solvers (Moulinec and Suquet, 1998). In particular, the effective properties of a linear elastic composite are obtained from a limited number of computations, with suitably chosen boundary conditions (Bornert et al., 2001, Chapitre 2).

The above-mentioned approaches provide a constitutive model for the effective or macroscopic response of the heterogeneous material. The associated material parameters are either analytically derived or obtained from numerical experiments on representative microstructures. Such model can be later used in standard FE simulations of a structure made of this heterogeneous material, referred to as the macrostructure. This methodology is particularly suited for elastic materials, for which the macroscopic stiffness tensor can be easily determined beforehand. However, constitutive relations of nonlinear composite materials are not easily derived from the behavior of their constituents. The mechanical response of a nonlinear microstructure subject to arbitrary macroscopic loading conditions can then not be determined beforehand. Full-field numerical simulations consisting in meshing with a high resolution all heterogeneities in the structure are still way too expensive. Instead, multiscale approaches search to solve two nested BVP defined on each of the micro- and macroscale. Such methods are commonly referred to as computational homogenization (Matouš et al., 2017). In particular, hierarchical methods popularized by Feyel (1999) under the name of FE^2 consist in solving a BVP at the microscale for each integration point of the macrostructure, with loading conditions based on the macroscopic strain-stress state. With this approach, the microstructure can be complex and evolving, and no assumption is made on the effective behavior of the composite material. Yet, depending on the complexity of the loading conditions applied to the macrostructure and the mesh resolution needed to fully represent the heterogeneities in the RVE, the approach can be computationally demanding. In addition, the material nonlinearity is usually handled using an iterative scheme at the macroscale which requires to compute the effective elastic tangent tensor of the RVE. This is typically done by numerical perturbation, thus requiring additional computations of the microstructure response: for a two-dimensional problem ($n = 2$), $n(n-1) = 3$ additional evaluations are needed, corresponding to the number of independent macroscopic strain components to vary; $n(n-1) = 6$ are required for a three-dimensional problem. The number of microscopic evaluations performed in a typical multilevel finite element approach (FE^2) simulation is then

$$N_{\text{micro. eval.}} = N_{\text{it.}} \times M \times (1 + n(n-1)), \tag{VI.8}$$

with $N_{\text{it.}}$ the number of iterations needed to reach macroscopic equilibrium, and M the number of integration points in the macrostructure. However, some of these evaluations may be redundant: it is most likely that, throughout the macroscopic iterations, two macroscopic integration points share a similar strain state. Hence, the same microscopic evaluation is performed twice, at different moment or different location of the simulation. To avoid this unnecessary computational cost, one can store the results of microscopic evaluations during the macroscopic simulation and re-use them when necessary, instead of re-computing them. This approach has been adopted by Klusemann and Ortiz (2015), where the average response $(\bar{\boldsymbol{\varepsilon}}, \bar{\boldsymbol{\sigma}})$ is stored in phase space $\mathbb{R}^{n\times n}_{\text{sym}} \times \mathbb{R}^{n\times n}_{\text{sym}}$. Then, an interpolation scheme is used, based on simplicial subdivision of the phase space, to approximate the effective response

from (already computed) nearby points in phase space. As demonstrated by the authors, the computational cost of FE^2 is significantly reduced.

1.2 Towards data-driven multiscale homogenization

An another approach in computational homogenization consists in building a database from numerous microscopic computations, and using it to feed a surrogate model of the macroscopic response. A vast number of techniques are used to derive the macroscopic response, from proper orthogonal decomposition to artificial neural networks (see Matouš et al. (2017) for an extensive list of recent "reduced order models, data mining and acceleration of nonlinear multiscale methods"). With these approaches, some kind of material modeling is still conducted: the effective response of the material is either interpolated, reduced to a manifold of lower dimension or approximated by an artificial neural network. In contrast, Xu et al. (2020) recently proposed to incorporate multiscale material modeling in the DDCM paradigm, following a similar but model-free methodology: the material database of effective strain and stress states is constituted beforehand and used to feed a data-driven solver. Mora-Macías et al. (2020) also proposed to apply their data completion technique presented in Ayensa-Jiménez et al. (2019) to a database obtained from experiments on bone tissue. In both contributions, the material database is generated (and completed) *a priori* in an off-line stage, *via* microscopic simulations or up-sampling technique. It requires to define beforehand the suitable loading conditions to apply to the microstructure, notably the range of solicitations and the density at which they must be sampled.

Instead, we propose in this work a first step to link DDCM and computational homogenization techniques such as FE^2: the resolution of the BVP at the macroscale is performed by DDCM while the resolution of the BVP at the microscale uses standard FE method, and is performed on-line. Between the two scales, the material database of macroscopic strain-stress states is completed *on-the-fly*. Then, no *a priori* knowledge of the effective material response nor the macroscopic loading conditions is required. In addition, note that the DDCM does not need the effective tangent operator. Our methodology thus aims at reducing the number of microscopic RVE resolutions: thanks to the structure of the data-driven problem, only relevant points are computed and (re-)used.

This last chapter is organized as follows. The data-driven FE^2 solver is presented in Section 2. In Section 3, we illustrate the method with the simulation of a two-dimensional structure made of a two-phases nonlinear composite. We conclude and give perspective for this particular application of DDCM in Section 4.

2 Data-driven FE^2 solver

We consider a two-stage FE simulation. The macrostructure is made of a nonlinear heterogeneous material. The microstructure is the same everywhere in the macrostructure, such that the same, unique, RVE is associated to all macroscopic

integration points. At the macro-level, we use the standard data-driven solver in small strain, as introduced by Kirchdoerfer and Ortiz (2016) and recalled in Chapter II. The material database is constituted with macroscopic strain-stress tensor pairs $\bar{\boldsymbol{\varepsilon}}, \bar{\boldsymbol{\sigma}}$, obtained from the FE resolution of the local BVP at the microscale, with boundary conditions defined by the mechanical state of the associated macroscopic integration point. The flowchart of the method is compared with that of the standard FE^2 scheme in Fig. VI.2.

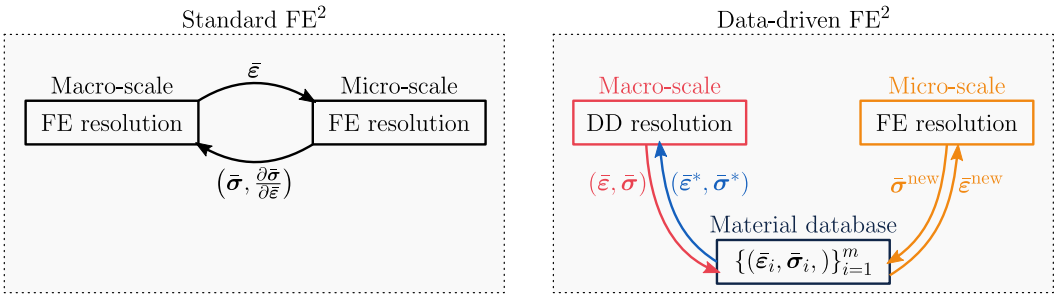


Figure VI.2 – Flowcharts of standard and data-driven FE^2 methods.

Our approach consists in conducting successive data-driven macroscopic simulations, with a material database that is enriched on-the-fly between two simulations. The data-driven solver is here used as a black-box, and we develop a strategy to *adaptively* enrich the material database with suitably chosen and non-redundant microscopic evaluations. First, we briefly recall from Chapter II the layout of the data-driven solver in Section 2.1 and insist on input and output data. Second, we expose in Section 2.2 the update strategy of the database and the subsequent two-scale data-driven solver.

2.1 Macroscopic data-driven solver

Let us consider a macroscopic FE mesh of N nodes, labeled $a \in [1\,..\,N]$, and M integration or quadrature or material points, labeled $e \in [1\,..\,M]$, in dimension $n = 2, 3$. Let $\alpha \in [1\,..\,N \times n]$ label the degrees of freedom; note that $\alpha = \alpha(a, i) = (a-1)n + i$, with $i \in [1\,..\,n]$ the direction. The system is subject to Dirichlet boundary conditions and applied external forces that are independent of the motion. The former are encoded in an array of n_{D} prescribed displacements $\left\{\bar{u}^{\text{D}}_\alpha\right\}_{\alpha \in D}$ with $D = \{\alpha_1, \cdots, \alpha_{n_{\text{D}}}\}$, such that $\alpha_k \in [1\,..\,Nn], \forall k \in [1\,..\,n_{\text{D}}]$. The latter are encoded in nodal force vectors $\left\{\bar{\boldsymbol{f}}_a \in \mathbb{R}^n\right\}_{a=1}^N$. The macroscopic local state of integration point e is denoted $z_e = (\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)$, and sits in the local phase space $\mathcal{Z}^{(e)}_{\text{loc}} = \mathbb{R}^{n\times n}_{\text{sym}} \times \mathbb{R}^{n\times n}_{\text{sym}}$. Note that using Voigt or Mandel vector notation, $\mathcal{Z}^{(e)}_{\text{loc}}$ is equivalently identified with $\mathbb{R}^{n(n-1)} \times \mathbb{R}^{n(n-1)}$. The global phase space is then the product set $\mathcal{Z} = \mathcal{Z}^{(1)}_{\text{loc}} \times \cdots \times \mathcal{Z}^{(M)}_{\text{loc}}$.

The material data set $\mathcal{D}$ is the product set of local material data sets $\mathcal{D}^{(e)}_{\text{loc}}$: $\mathcal{D} = \mathcal{D}^{(1)}_{\text{loc}} \times \cdots \times \mathcal{D}^{(M)}_{\text{loc}}$. In the following, as the microstructure is supposed to be the

same at each integration point, we consider only one local material data set, shared by all macroscopic points, denoted $\mathcal{D}_{\text{loc}}$ and referred to as the material database. For the same reason, the same local phase $\mathcal{Z}_{\text{loc}}$ is shared by all integration points. The material database is constituted with averaged strain and stress states, obtained from m microscopic computations:

$$\mathcal{D}_{\text{loc}} = \{(\bar{\boldsymbol{\varepsilon}}_i, \bar{\boldsymbol{\sigma}}_i) \mid i \in [1 \,..\, m]\}\,. \tag{VI.9}$$

On the one hand, the material strain-stress field $(\bar{\boldsymbol{\varepsilon}}', \bar{\boldsymbol{\sigma}}')$ consists of a selection in the local material database. On the other hand, the mechanical strain-stress field $(\bar{\boldsymbol{\varepsilon}}, \bar{\boldsymbol{\sigma}})$ must verify both macroscopic kinematics constraints and stress equilibrium, written as

$$\bar{\boldsymbol{\varepsilon}}_e = \sum_{a=1} \boldsymbol{B}_{ea} \bar{\boldsymbol{u}}_a, \qquad \forall e \in [1 \,..\, M], \tag{VI.10a}$$

$$\bar{\boldsymbol{u}}_a \cdot \boldsymbol{e}_i = \bar{u}^{\text{D}}_\alpha, \qquad \forall (a,i), \alpha(a,i) \in D, \tag{VI.10b}$$

$$\left(\sum_{e=1}^{M} w_e \boldsymbol{B}^T_{ea} \bar{\boldsymbol{\sigma}}_e - \bar{\boldsymbol{f}}_a \right) \cdot \boldsymbol{e}_i = 0, \qquad \forall (a,i), \alpha(a,i) \notin D, \tag{VI.10c}$$

where $\{\bar{\boldsymbol{u}}_a\}_{a=1}^N$ are the macroscopic nodal displacement vectors, the standard matrix $\boldsymbol{B}_{ea}$ encodes the connectivity and geometry of the FE mesh, and w_e is the integration weight of quadrature point e. The set of mechanical state fields is referred to as the constraint set and writes

$$\mathcal{E} = \{(\bar{\boldsymbol{\varepsilon}}, \bar{\boldsymbol{\sigma}}) \in \mathcal{Z} \mid \text{Eq. (VI.10)}\}\,. \tag{VI.11}$$

The data-driven solution $\mathcal{S}$ is defined as the mechanical state field $(\bar{\boldsymbol{\varepsilon}}, \bar{\boldsymbol{\sigma}}) \in \mathcal{E}$, closest to the material data set $\mathcal{D}$:

$$\mathcal{S} = \arg \min_{(\bar{\boldsymbol{\varepsilon}}, \bar{\boldsymbol{\sigma}}) \in \mathcal{E}} \min_{(\bar{\boldsymbol{\varepsilon}}', \bar{\boldsymbol{\sigma}}') \in \mathcal{D}} \sum_{e=1}^{M} w_e \; d_e \left((\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e), (\bar{\boldsymbol{\varepsilon}}'_e, \bar{\boldsymbol{\sigma}}'_e) \right), \tag{VI.12}$$

where the local distance function d_e is defined in $\mathcal{Z}_{\text{loc}} \times \mathcal{D}_{\text{loc}}$ by

$$d_e \left((\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e), (\bar{\boldsymbol{\varepsilon}}_i, \bar{\boldsymbol{\sigma}}_i) \right) = W(\bar{\boldsymbol{\varepsilon}}_e - \bar{\boldsymbol{\varepsilon}}_i) + W^*(\bar{\boldsymbol{\sigma}}_e - \bar{\boldsymbol{\sigma}}_i), \tag{VI.13}$$

with W and W^* strain and complementary energy density-like deviation functions, written as

$$W(\boldsymbol{\varepsilon}) = \frac{1}{2} (\mathbb{C}\boldsymbol{\varepsilon})\, \boldsymbol{\varepsilon}, \tag{VI.14a}$$

$$W^*(\boldsymbol{\sigma}) = \frac{1}{2} (\mathbb{C}^{-1}\boldsymbol{\sigma})\, \boldsymbol{\sigma}, \tag{VI.14b}$$

with $\mathbb{C}$ a constant positive definite operator.

The coupled continuous and combinatorial optimization Eq. (VI.12) is addressed with an alternated minimization scheme (Kirchdoerfer and Ortiz, 2016):

1. The material states are obtained from independent nearest neighbor searches in local phase space. The material state $(\bar{\boldsymbol{\varepsilon}}^*_{\mathsf{ie}}, \bar{\boldsymbol{\sigma}}^*_{\mathsf{ie}})$ is defined as the closest material data point in $\mathcal{D}_{\text{loc}}$ from the current mechanical state $(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)$ of integration point e:

$$d_e\left((\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e), (\bar{\boldsymbol{\varepsilon}}^*_{\mathsf{ie}}, \bar{\boldsymbol{\sigma}}^*_{\mathsf{ie}})\right) \leq d_e\left((\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e), (\bar{\boldsymbol{\varepsilon}}_i, \bar{\boldsymbol{\sigma}}_i)\right), \\ \forall e \in [1 \,..\, M],\ \forall (\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e) \in \mathcal{Z}_{\text{loc}},\ \forall (\bar{\boldsymbol{\varepsilon}}_i, \bar{\boldsymbol{\sigma}}_i) \in \mathcal{D}_{\text{loc}}, \quad \text{(VI.15)}$$

2. The mechanical states are subsequently derived from the constrained minimization problem

$$\mathcal{S} = \arg \min_{(\bar{\boldsymbol{\varepsilon}}, \bar{\boldsymbol{\sigma}}) \in \mathcal{Z}} \sum_{e=1}^{M} w_e\, d_e\left((\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e), (\bar{\boldsymbol{\varepsilon}}^*_{\mathsf{ie}}, \bar{\boldsymbol{\sigma}}^*_{\mathsf{ie}})\right), \text{ subject to } (\bar{\boldsymbol{\varepsilon}}, \bar{\boldsymbol{\sigma}}) \in \mathcal{E}. \quad \text{(VI.16)}$$

For more details on how the constrained minimization problem Eq. (VI.16) is addressed, the reader is referred back to Chapter II. Here, we simply briefly recall the layout of the data-driven solver, applied to the macroscopic BVP, in Algorithm VI.1.

2.2 Two-scale data-driven solver

2.2.1 Motivation

The major input of the macroscopic data-driven solver Algorithm VI.1 is the material database $\mathcal{D}_{\text{loc}}$. As aforementioned, it is generated by solving standard FE simulations on the RVE of the microstructure. To generate m material data points in $\mathcal{D}$, one must then perform m microscopic evaluations. If linear displacement boundary conditions Eq. (VI.4) are adopted, the space of macroscopic strain tensors $\bar{\boldsymbol{\varepsilon}}$ must be sampled by m points. For example, in a two-dimensional problem, one can sample a regular grid in a domain $\mathbb{E} = \left[\bar{\varepsilon}^{\min}_{11}, \bar{\varepsilon}^{\max}_{11}\right] \times \left[\bar{\varepsilon}^{\min}_{12}, \bar{\varepsilon}^{\max}_{12}\right] \times \left[\bar{\varepsilon}^{\min}_{22}, \bar{\varepsilon}^{\max}_{22}\right] \subset \mathbb{R}^3$, as in Xu et al. (2020). However, for nonlinear materials, the bounds of the domain can hardly be intuited beforehand. In addition, there is no simple way to estimate the necessary resolution of the grid to obtain a satisfactory data-driven macroscopic solution. Finally, as shown in our analysis of material databases in Chapter IV, a large number of the pre-computed material data points using a regular grid like $\mathbb{E}$ turns useless in the macroscopic computation, as the corresponding region in phase space is not necessarily explored by the macroscopic mechanical response. Hence, the computational effort of such *off-line* building of the material database is time consuming and not controlled enough.

2.2.2 General idea and algorithm

In contrast, we propose a novel hierarchical data-driven FE^2 method, which ensures to compute only the material data points that are needed for the equilibrium of the macrostructure. Indeed, until now, we have not used one of the key output of

Algorithm VI.1 Macroscopic data-driven solver.

INPUT: Material database $\mathcal{D}_{\text{loc}}$, mesh of N nodes and M integration points, boundary and loading conditions $\left\{\bar{u}^{\text{D}}_{\alpha}\right\}_{\alpha\in D}$ and $\left\{\bar{\boldsymbol{f}}_a\right\}_{a=1}^{N}$.

OUTPUT: Mechanical states $\{(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)\}_{e=1}^{M}$, material states $\{(\bar{\boldsymbol{\varepsilon}}^{*}_{\text{i}e}, \bar{\boldsymbol{\sigma}}^{*}_{\text{i}e})\}_{e=1}^{M}$, and distances $\{d_e\}_{e=1}^{M}$ between them.

Step 1. Set $k = 0$. Initialize all material states to zero strain-stress state:

$$\left(\bar{\boldsymbol{\varepsilon}}^{*,(k)}_{\text{i}e}, \bar{\boldsymbol{\sigma}}^{*,(k)}_{\text{i}e}\right) = (\boldsymbol{0}, \boldsymbol{0}), \forall e \in [1\,..\,M]. \tag{VI.17}$$

Step 2. Compute mechanical states $(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)$

- Compute nodal displacements $\bar{\boldsymbol{u}}_a$ and Lagrange multipliers $\bar{\boldsymbol{\eta}}_a$ by solving two pseudo-elastic linear systems

$$\sum_{b=1}^{N}\left(\sum_{e=1}^{M} w_e\, \boldsymbol{B}^T_{ea}\, \mathbb{C}\, \boldsymbol{B}_{eb}\right)\bar{\boldsymbol{u}}_b = \sum_{e=1}^{M} w_e \boldsymbol{B}^T_{ea}\, \mathbb{C}\, \bar{\boldsymbol{\varepsilon}}^{*,(k)}_{\text{i}e}, \qquad \forall a \in [1\,..\,N], \tag{VI.18a}$$

$$\sum_{b=1}^{N}\left(\sum_{e=1}^{M} w_e\, \boldsymbol{B}^T_{ea}\, \mathbb{C}\, \boldsymbol{B}_{eb}\right)\bar{\boldsymbol{\eta}}_b = \bar{\boldsymbol{f}}_a - \sum_{e=1}^{M} w_e\, \boldsymbol{B}^T_{ea}\, \bar{\boldsymbol{\sigma}}^{*,(k)}_{\text{i}e} \qquad \forall a \in [1\,..\,N]. \tag{VI.18b}$$

  with boundary conditions $\bar{\boldsymbol{u}}_a \cdot \boldsymbol{e}_i = \bar{u}^{\text{D}}_{\alpha}$, and $\bar{\boldsymbol{\eta}}_a \cdot \boldsymbol{e}_i = 0$, for all $(a, i) \in [1\,..\,N] \times [1\,..\,n]$ such that $\alpha(a, i) \in D$. Note that we also set $\bar{\boldsymbol{f}}_a \cdot \boldsymbol{e}_i = 0, \forall \alpha(a, i) \in D$.

- Compute local states

$$\bar{\boldsymbol{\varepsilon}}_e = \sum_{a=1}^{N} \boldsymbol{B}_{ea}\, \bar{\boldsymbol{u}}_a, \qquad \forall e \in [1\,..\,M], \tag{VI.19a}$$

$$\bar{\boldsymbol{\sigma}}_e = \bar{\boldsymbol{\sigma}}^{*,(k)}_{\text{i}e} + \sum_{a=1}^{N} \mathbb{C}\, \boldsymbol{B}_{ea}\, \bar{\boldsymbol{\eta}}_a, \qquad \forall e \in [1\,..\,M]. \tag{VI.19b}$$

Step 3. Update material states

$$\left(\bar{\boldsymbol{\varepsilon}}^{*,(k+1)}_{\text{i}e}, \bar{\boldsymbol{\sigma}}^{*,(k+1)}_{\text{i}e}\right) = \arg \min_{(\bar{\boldsymbol{\varepsilon}}_i, \bar{\boldsymbol{\sigma}}_i)\in\mathcal{D}_{\text{loc}}} d_e\left((\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e), (\bar{\boldsymbol{\varepsilon}}_i, \bar{\boldsymbol{\sigma}}_i)\right) \tag{VI.20}$$

Step 4. Test convergence

- if $\left(\bar{\boldsymbol{\varepsilon}}^{*,(k)}_{\text{i}e}, \bar{\boldsymbol{\sigma}}^{*,(k)}_{\text{i}e}\right) = \left(\bar{\boldsymbol{\varepsilon}}^{*,(k+1)}_{\text{i}e}, \bar{\boldsymbol{\sigma}}^{*,(k+1)}_{\text{i}e}\right)$, then exit.
- else, set $k = k + 1$ and return to Step 2.

the data-driven solvers: the final local distances $\{d_e\}_{e=1}^{M}$ between the mechanical states and their associated material states, as defined in Eq. (VI.13). It provides a direct information of the density of material data points around the current equilibrated and compatible solution. A large distance d_e indicates that the material state $(\bar{\boldsymbol{\varepsilon}}_{\mathrm{ie}}^{*}, \bar{\boldsymbol{\sigma}}_{\mathrm{ie}}^{*})$ lies too far from the associated mechanical state $(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)$. The region in local phase space surrounding $(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)$ should then be enriched. Such a mechanical state, for which the distance to the closest material data point is larger than a critical distance d_c, is referred to as *isolated*. The set of isolated integration points is denoted I and writes

$$I = \left\{ e \mid d_e\left((\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e), (\bar{\boldsymbol{\varepsilon}}_{\mathrm{ie}}^{*}, \bar{\boldsymbol{\sigma}}_{\mathrm{ie}}^{*}) \right) > d_c \right\} \subset [1 \,..\, M]. \tag{VI.21}$$

In that perspective, we propose to conduct successive macroscopic data-driven simulations, with an enrichment of the material database between each simulation that is based on the final values of d_e. Considering a material database $\mathcal{D}_{\mathrm{loc}}^{(j)}$ at iteration j, from which results a data-driven solution $\left\{ \left(\bar{\boldsymbol{\varepsilon}}_e^{(j)}, \bar{\boldsymbol{\sigma}}_e^{(j)} \right) \right\}_{e=1}^{M}$, we enrich the database for the next iteration $j+1$ *via* the homogenization of isolated mechanical states:

$$\mathcal{D}_{\mathrm{loc}}^{(j+1)} = \mathcal{D}_{\mathrm{loc}}^{(j)} \bigcup_{e \in I} \{ (\bar{\boldsymbol{\varepsilon}}_e^{\mathrm{new}}, \bar{\boldsymbol{\sigma}}_e^{\mathrm{new}}) \}, \tag{VI.22}$$

where $\bar{\boldsymbol{\varepsilon}}_e^{\mathrm{new}}$ and $\bar{\boldsymbol{\sigma}}_e^{\mathrm{new}}$ results from a computational homogenization scheme: (i) localization with one of Eqs. (VI.4) to (VI.6) boundary conditions, (ii) FE resolution of the local BVP Eq. (VI.2), (iii) volume average Eq. (VI.1). Two evident choices are then available:

- the updated strain is computed from the mechanical stress, using uniform traction boundary conditions Eq. (VI.5), as

$$\bar{\boldsymbol{\varepsilon}}^{\mathrm{new}} = \mathcal{H}(\Omega, \boldsymbol{t} = \bar{\boldsymbol{\sigma}}_e \boldsymbol{n}) \text{ and } \bar{\boldsymbol{\sigma}}_e^{\mathrm{new}} = \bar{\boldsymbol{\sigma}}_e, \tag{VI.23}$$

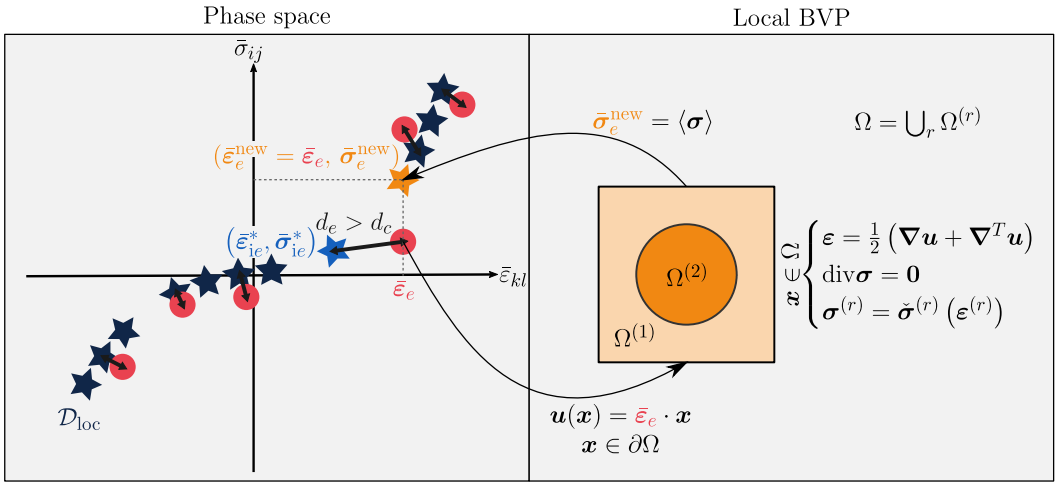
- the updated stress is obtained from the mechanical strain, using linear displacement boundary conditions Eq. (VI.4) (or periodic boundary conditions Eq. (VI.6)), as

$$\bar{\boldsymbol{\sigma}}^{\mathrm{new}} = \mathcal{H}(\Omega, \boldsymbol{u} = \bar{\boldsymbol{\varepsilon}}_e \boldsymbol{x}) \text{ and } \bar{\boldsymbol{\varepsilon}}_e^{\mathrm{new}} = \bar{\boldsymbol{\varepsilon}}_e, \tag{VI.24}$$

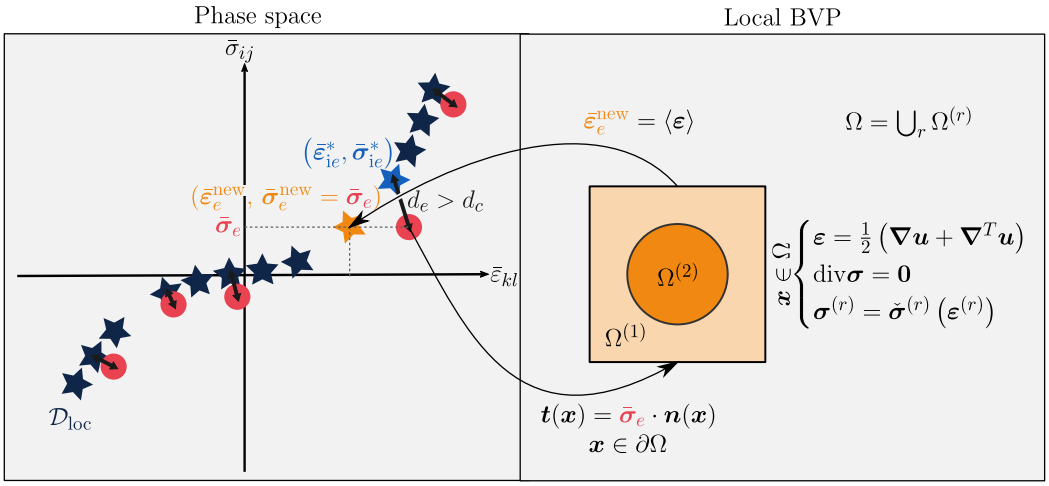
where $\mathcal{H}(\Omega, \bullet)$ denotes the computational homogenization scheme applied to the RVE Ω, using the boundary conditions $\bullet$. The *stress-from-strain* and *strain-from-stress* strategies[1] Eq. (VI.24) are illustrated in Fig. VI.3. The implications of each strategy are discussed next.

In the database update Eq. (VI.22), no unnecessary nor redundant microscopic evaluation is conducted: if a material data point lies close enough from a mechanical state ($d_e \leq d_c$), the corresponding microscopic simulation will not be performed

[1] An intermediate strategy which would compute both macroscopic strain and stress states could also be envisaged in future work.



(a) Stress-from-strain strategy.



(b) Strain-from-stress strategy.

Figure VI.3 – Database update strategies. New material data points are obtained from the resolution of local BVPs on the RVE of the microstructure, with (a) prescribed macroscopic strains $\bar{\boldsymbol{\varepsilon}}_e$ or (b) prescribed macroscopic stresses $\bar{\boldsymbol{\sigma}}_e$, selected from the mechanical states lying too far from their associated material state.

another time. In addition, the effective mechanical response of the microstructure between two close macroscopic strain states is not interpolated nor approximated, since the macroscopic data-driven solver is a model-free, projection-based iterative solver. Our concurrent data-driven multiscale approach is then an iterative process, where the material database is successively enriched *on-line*, in an adaptive manner, based on the current macroscopic solution. The algorithm is listed in Algorithm VI.2.

Algorithm VI.2 Two-scale data-driven solver.

INPUT:

- Macroscale: FE model with N nodes and M integration points, boundary and loading conditions $\left\{\bar{u}^{\mathrm{D}}_{\alpha}\right\}_{\alpha\in D}$ and $\left\{\bar{\boldsymbol{f}}_a\right\}_{a=1}^{N}$.
- Microscale: FE model of a RVE of the microstructure, constitutive models of the constituents.

OUTPUT:

- Macroscale: mechanical states $\{(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)\}_{e=1}^{M}$, material database $\mathcal{D}_{\mathrm{loc}}$.
- Microscale: local and average mechanical response for every macroscopic loading point in $\mathcal{D}_{\mathrm{loc}}$.

Step 1. Set $j = 0$. Initialize the material database $\mathcal{D}^{(0)}_{\mathrm{loc}}$.

Step 2. Compute the macroscopic mechanical response from `DDSOLVER` in Algorithm VI.1:

$$\left(\bar{\boldsymbol{\varepsilon}}^{(j)}, \bar{\boldsymbol{\sigma}}^{(j)}\right) = \texttt{DDSOLVER}\left(\mathcal{D}^{(j)}_{\mathrm{loc}}\right).$$

Step 3. Enrich the material database from the homogenization of isolated mechanical states:

$$\mathcal{D}^{(j+1)}_{\mathrm{loc}} = \mathcal{D}^{(j)}_{\mathrm{loc}} \bigcup_{e\in I} \{(\bar{\boldsymbol{\varepsilon}}^{\mathrm{new}}_e, \bar{\boldsymbol{\sigma}}^{\mathrm{new}}_e)\},$$

with I as in Eq. (VI.21) and $(\bar{\boldsymbol{\varepsilon}}^{\mathrm{new}}_e, \bar{\boldsymbol{\sigma}}^{\mathrm{new}}_e)$ as in Eq. (VI.23) or Eq. (VI.24).

Step 4. Test convergence:

- if $d_e \leq d_c$, $\forall e \in [1\,..\,M]$, then exit.
- else, return to Step 2.

2.2.3 Initial material database

It now remains to build an initial database $\mathcal{D}^{(0)}_{\mathrm{loc}}$. This can be done by a sampling of a coarse grid $\mathbb{E}$ in the macroscopic strain space. In that case, the issue of determining *a priori* suitable bounds and resolution for that grid persists. To completely avoid hazardous assumptions on the range or the density of the grid $\mathbb{E}$, we propose to start from a singleton database, with zero as the only point:

$$\mathcal{D}^{(0)}_{\mathrm{loc}} = \{(\mathbf{0}, \mathbf{0})\}. \tag{VI.25}$$

Note that this point does exist for any elastic heterogeneous material. Step 1 of Algorithm VI.2 is now covered. We examine the consequences on Steps 2 and 3 in

following.

Together with the zero-initialization method of the material states in the macroscopic data-driven solver, a zero initial database implies that the first macroscopic data-driven solution is a linear elastic estimate of stiffness $\mathbb{C}$ of the macroscopic BVP. From Eqs. (VI.17) and (VI.18) in Algorithm VI.1, the first mechanical states are obtained from the boundary and loading conditions, and the parameter $\mathbb{C}$ only, as

$$\bar{\boldsymbol{\varepsilon}}_e^{(0)} = \sum_{a=1}^{N} \boldsymbol{B}_{ea}\, \bar{\boldsymbol{u}}_a,\ \forall e \in [1\,..\,M] \tag{VI.26a}$$

with $\{\bar{\boldsymbol{u}}_a\}_{a=1}^{N}$ such that

$$\sum_{b=1}^{N} \boldsymbol{K}_{ab}\, \bar{\boldsymbol{u}}_b = \boldsymbol{0},\ \forall a \in [1\,..\,N],\ \text{ and } \bar{\boldsymbol{u}}_a \cdot \boldsymbol{e}_i = \bar{u}_\alpha^{\mathrm{D}},\ \forall \alpha(a,i) \in D, \tag{VI.26b}$$

and

$$\bar{\boldsymbol{\sigma}}_e^{(0)} = \sum_{a=1}^{N} \mathbb{C}\, \boldsymbol{B}_{ea}\, \bar{\boldsymbol{\eta}}_a,\ \forall e \in [1\,..\,M] \tag{VI.27a}$$

with $\{\bar{\boldsymbol{\eta}}_a\}_{a=1}^{N}$ such that

$$\sum_{b=1}^{N} \boldsymbol{K}_{ab}\, \bar{\boldsymbol{\eta}}_b = \bar{\boldsymbol{f}}_a,\ \forall a \in [1\,..\,N],\ \text{ and } \bar{\boldsymbol{\eta}}_a \cdot \boldsymbol{e}_i = 0,\ \forall \alpha(a,i) \in D, \tag{VI.27b}$$

where $\boldsymbol{K}_{ab}$ is the common tangent operator between node a and b:

$$\boldsymbol{K}_{ab} = \sum_{e=1}^{M} w_e\, \boldsymbol{B}_{ea}^T\, \mathbb{C}\, \boldsymbol{B}_{eb}. \tag{VI.28}$$

As the material database contains only one material data point, the macroscopic data-driven solvers stops after this first iteration. The output mechanical states are then as in Eqs. (VI.26) and (VI.27). To go a bit further, we consider two standard loading cases: (i) purely displacement-controlle, and (ii) force-driven with homogeneous Dirichlet conditions.

- Let us first consider the case of a purely displacement-controlled macroscopic simulation: $\bar{\boldsymbol{f}}_a = \boldsymbol{0},\ \forall a \in [1\,..\,N]$ and $\bar{\boldsymbol{u}}_a \cdot \boldsymbol{e}_i = \bar{u}_\alpha^{\mathrm{D}},\ \forall \alpha(a,i) \in D$ where there is at least one non-zero prescribed displacement $\bar{u}_\alpha^{\mathrm{D}} \neq 0$. Then, from Eqs. (VI.26) and (VI.27), the first linear elastic estimates are

  $$\bar{\boldsymbol{\sigma}}_e^{(0)} = \boldsymbol{0} \text{ and } \bar{\boldsymbol{\varepsilon}}_e^{(0)} = \bar{\boldsymbol{\varepsilon}}_e^{\mathbb{C}},\ \forall e \in [1\,..\,M] \tag{VI.29}$$

  with $\left\{\bar{\boldsymbol{\varepsilon}}_e^{\mathbb{C}}\right\}_{e=1}^{M}$ the strain solution of the same macroscopic BVP with an homogeneous and linear elastic material of stiffness $\mathbb{C}$.

- Let us now consider the case of a force-driven macroscopic simulation: $\exists a \in [1 \,..\, N]$, $\bar{\boldsymbol{f}}_a \neq \boldsymbol{0}$ and $\bar{\boldsymbol{u}}_a \cdot \boldsymbol{e}_i = 0, \forall \alpha(a,i) \in D$. Then, from Eqs. (VI.26) and (VI.27), the first linear elastic estimates are

$$\bar{\boldsymbol{\sigma}}_e^{(0)} = \bar{\boldsymbol{\sigma}}_e^{\mathbb{C}} \text{ and } \bar{\boldsymbol{\varepsilon}}_e^{(0)} = \boldsymbol{0}, \forall e \in [1 \,..\, M] \tag{VI.30}$$

  with $\left\{\bar{\boldsymbol{\sigma}}_e^{\mathbb{C}}\right\}_{e=1}^{M}$ the stress solution of the same macroscopic BVP with an homogeneous and linear elastic material of stiffness $\mathbb{C}$.

At the end of Step 2 of Algorithm VI.2, the mechanical states are then either

$$\mathcal{S}^{(0)} = \left\{\left(\bar{\boldsymbol{\varepsilon}}_e^{\mathbb{C}}, \boldsymbol{0}\right)\right\}_{e=1}^{M} \tag{VI.31}$$

for a purely displacement-controlled simulation, or

$$\mathcal{S}^{(0)} = \left\{\left(\boldsymbol{0}, \bar{\boldsymbol{\sigma}}_e^{\mathbb{C}}\right)\right\}_{e=1}^{M} \tag{VI.32}$$

for a force-driven simulation.

In Step 3, the new material database $\mathcal{D}_{\text{loc}}^{(1)}$ is determined from $\mathcal{S}^{(0)}$. For the two macroscopic loading cases considered, it is evident that the *stress-from-strain* strategy Eq. (VI.24) must be chosen for purely displacement-controlled simulations and that, conversely, the *strain-from-stress* strategy Eq. (VI.23) is adapted to force-driven simulations: one of the state field in both situations is zero. In addition, the local distances are respectively $d_e = W(\bar{\boldsymbol{\varepsilon}}_e^{\mathbb{C}})$ and $d_e = W^*(\bar{\boldsymbol{\sigma}}_e^{\mathbb{C}})$, with W and W^* as in Eq. (VI.14). If the critical distance d_c is sufficiently small, then every mechanical state is most likely isolated: $d_e \leq d_c$ for all e and $I = [1 \,..\, M]$. At the end of Step 3 of Algorithm VI.2, the material database is then

$$\mathcal{D}_{\text{loc}}^{(1)} = \{(\boldsymbol{0}, \boldsymbol{0})\} \bigcup_{e=1}^{M} \left\{\left(\bar{\boldsymbol{\varepsilon}}_e^{\mathbb{C}}, \mathcal{H}\left(\Omega, \boldsymbol{u} = \bar{\boldsymbol{\varepsilon}}_e^{\mathbb{C}} \cdot \boldsymbol{x}\right)\right)\right\} \tag{VI.33}$$

for a displacement-controlled simulation, and

$$\mathcal{D}_{\text{loc}}^{(1)} = \{(\boldsymbol{0}, \boldsymbol{0})\} \bigcup_{e=1}^{M} \left\{\left(\mathcal{H}\left(\Omega, \boldsymbol{t} = \bar{\boldsymbol{\sigma}}_e^{\mathbb{C}} \cdot \boldsymbol{n},\right) \bar{\boldsymbol{\sigma}}_e^{\mathbb{C}}\right)\right\} \tag{VI.34}$$

for a force-driven simulation. The first iteration in the two-scale data-driven solver is then merely a way to initialize the material database based on the macroscopic loading conditions instead of from a guessed grid in strain space.

Starting from this linear elastic estimate, the objective is to efficiently and rapidly populate the local phase space in the next iterations. It requires to adapt the value of $\mathbb{C}$ and the update strategy to the macroscopic loading conditions, as discussed next.

2.2.4 Adapted parameter $\mathbb{C}$ and modified update strategy

For the sake of simplicity, we restrict to the pure displacement-controlled case from here on out.

Assume that $\mathbb{C}$ is set to an isotropic linear elastic stiffness tensor $\mathbb{H}(E,\nu)$, defined by a Young's modulus E and a Poisson's ratio ν as

$$\mathbb{H}(E,\nu) = \lambda(E,\nu)\ (\boldsymbol{I}\otimes\boldsymbol{I}) + 2\mu(E,\nu)\,\mathbb{I}^{\mathrm{SYM}}, \tag{VI.35}$$

with $\lambda(E,\nu) = E\nu/(1+\nu)/(1-2\nu)$ (resp. $\mu(E,\nu) = E/2/(1+\nu)$) the standard first (resp. second) Lamé constant, $\mathbb{I}^{\mathrm{SYM}} = 1/2(\delta_{ik}\delta_{jl}+\delta_{il}\delta_{ik})$ the symmetric fourth-order identity tensor, and $\boldsymbol{I}$ the second-order identity tensor. Then, the solutions Eqs. (VI.31) and (VI.32) are independent of the modulus E. However, the value of the modulus of $\mathbb{C}$ influences the selection of the closest material data point in data-driven solvers. When the modulus is very large (resp. very small), the minimization of the strain-deviation $W(\bar{\boldsymbol{\varepsilon}}_e - \bar{\boldsymbol{\varepsilon}}_i)$ is favored (resp. depreciated) over the stress part $W^*(\bar{\boldsymbol{\sigma}}_e - \bar{\boldsymbol{\sigma}}_i)$ in Eq. (VI.13). This yields a different pairing between the mechanical and material states, as illustrated in Fig. VI.4 (see also Appendix A for a geometrical interpretation in one dimension). Here, we follow the recommendation

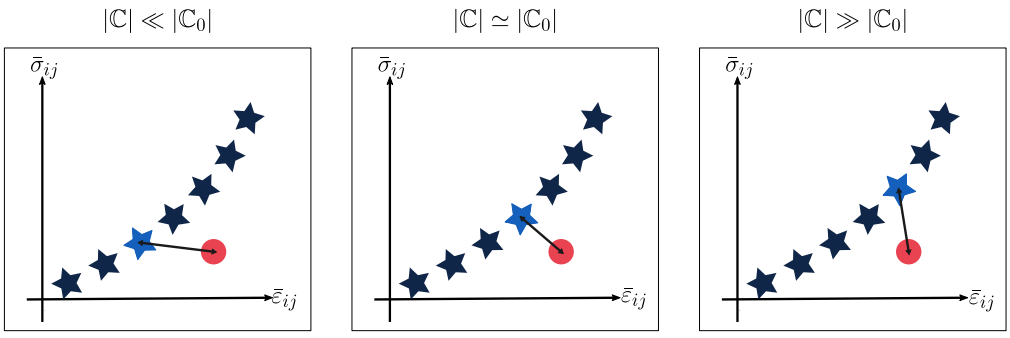


Figure VI.4 – Influence of the modulus of $\mathbb{C}$ on the local data assignment.

made in the DDI approach, were the strains are directly prescribed by the displacements obtained from DIC: $\mathbb{C}$ is more suitably set with a very large modulus, as the mechanical strains are more likely to be accurately predicted and hence more trustworthy (Dalémat, 2019, Chapter 3). We also assessed the relevance of this choice in DDCM with extensive study on truss structures, not reported here. In that case, the next macroscopic data-driven solution $\mathcal{S}^{(1)}$ is most likely to be close to the material database $\mathcal{D}^{(1)}_{\mathrm{loc}}$ of Eq. (VI.33). Then, every mechanical state is paired with the material data points resulting from the corresponding homogenization conducted in the previous state. The mechanical strains are unchanged from one iteration to the next, hence no new material data points can be added to the database, and our iterative process (prematurely) stops at the end of iteration $j = 1$. The phenomena is illustrated in Fig. VI.5 with $M = 6$. Each stage represented in Fig. VI.5 is de-

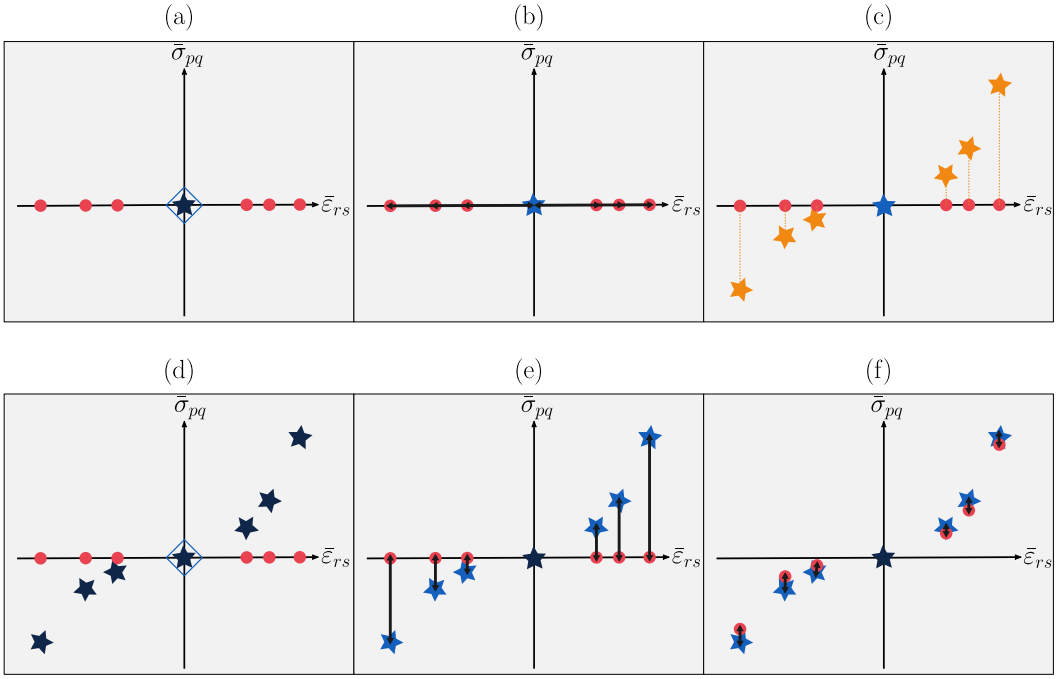


Figure VI.5 – Premature exit of the two-scale data-driven solver for a purely displacement-controlled macroscopic simulation and a large value of the $\mathbb{C}$ parameter. See in the text for a description of each phase.

scribed below (note that k refer to the iterations in Algorithm VI.1 and j refers to the iterations in Algorithm VI.3):

(a) $j = 0, k = 0$: the mechanical states (red circle) results from the zero singleton database $\mathcal{D}^{(0)}_{\text{loc}}$ (dark blue star), with zero initial material state (light blue diamond).

(b) $j = 0, k = 1$: the mechanical states are all associated to the same zero material state (light blue star). The macroscopic data-driven solver stops.

(c) end of $j = 0$: all mechanical states lie too far from their associated material states: $I = \{1, 2, 3, 4, 5, 6\}$. Corresponding points are added to the database with the stress-from-strain update strategy.

(d) $j = 1, k = 0$: the mechanical states (red circle) results from the new database $\mathcal{D}^{(1)}_{\text{loc}}$ (dark blue stars), with zero initial material state (light blue diamond).

(e) $j = 1$, end of $k = 0$: the new material states (light blue stars) are found from a high value of $\mathbb{C}$.

(f) $j = 1$, end of $k = 1$: the mechanical states (red circles) are computed from the corresponding material states (light blue stars). The material and mechanical strains are equal, the database can not be enriched further. The two-scale

data-driven solver stops without having computed any other material data points than those resulting from the first linear elastic estimate (orange stars in (c)).

To avoid this issue, we slightly modified the update strategy: instead of adding *all* isolated mechanical states to the database, we only add the most isolated ones, provided that their number does not exceed a certain limit $M_{\text{limit}} < M$. A typical choice can be $M_{\text{limit}} = M/2$. The threshold d_c is then adapted at each iteration so that $|I| \leq M_{\text{lim}}$, where $|I|$ denotes the number of elements in or cardinality of I. We show in Section 3 that this simple modification is enough to get the two-scale solver working. This also allows for limiting the number of microscopic evaluations. All-in-all, the total number of microscopic evaluations required by our method is bounded by:

$$N_{\text{micro. eval.}} \leq N_{\text{multi. iter.}} \times M_{\text{lim}}. \tag{VI.36}$$

As compared with standard FE^2 (see Eq. (VI.8)) the number of microscopic evaluations is greatly reduced, provided that $N_{\text{multi. iter.}}$ remains reasonably small, since $M_{\text{lim}} < M$.

Note that alternative methods to the workarounds exposed in this section could be employed in future work. In particular, instead of arbitrarily selecting the most isolated mechanical states, one could think of adding a fixed number of centroids resulting from a clustering of isolated mechanical states in phase space. In addition, it might be preferable to have differents values of the $\mathbb{C}$ parameter for the macroscopic simulation and for the selection of isolated mechanical states: the former should insure good convergence properties of the macroscopic data-driven solver while the latter could be optimized to add the relevant mechanical states.

The two-scale data-driven solver thus adapted to purely displacement-controlled macroscopic simulations is listed in Algorithm VI.3.

Algorithm VI.3 Two-scale data-driven solver for displacement-controlled simulations.

INPUT:

- Macroscale: FE model with N nodes and M integration points, boundary conditions $\left\{\bar{u}_\alpha^{\mathrm{D}}\right\}_{\alpha\in D}$.
- Microscale: FE model of a RVE of the microstructure, constitutive models of the constituents.

OUTPUT:

- Macroscale: mechanical states $\{(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)\}_{e=1}^{M}$, material data set $\mathcal{D}$.
- Microscale: local and average mechanical response for every macroscopic loading point in $\mathcal{D}_{\mathrm{loc}}$.

Require: $\mathbb{C}$ parameter with large modulus, d_c critical distance threshold.

Step 1. Set $j = 0$. Initialize the material database $\mathcal{D}_{\mathrm{loc}}^{(0)}$.

Step 2. Compute the macroscopic mechanical response from `DDSOLVER` in Algorithm VI.1:

$$\left(\bar{\boldsymbol{\varepsilon}}^{(j)}, \bar{\boldsymbol{\sigma}}^{(j)}\right) = \texttt{DDSOLVER}\left(\mathcal{D}_{\mathrm{loc}}^{(j)}\right).$$

Step 3. Compare the local distances with the threshold d_c and select up to M_{lim} isolated mechanical states:

- $I = \{e \in [1\,..\,M] \mid d_e > d_c\}$
- if $|I| > M_{\mathrm{lim}}$, then find $d_{c,\mathrm{lim}} > d_c$ such that

$$I = \{e \in [1\,..\,M] \mid d_e > d_{c,\mathrm{lim}}\} \text{ and } |I| \leq M_{\mathrm{lim}}. \tag{VI.37}$$

Step 4. Test convergence:

- if $d_e \leq d_c$, $\forall e \in [1\,..\,M]$ (or equivalently $I = \emptyset$), then exit.
- else,
  - enrich the material database with the *stress-from-strain* update strategy:

$$\mathcal{D}_{\mathrm{loc}}^{(j+1)} = \mathcal{D}_{\mathrm{loc}}^{(j)} \bigcup_{e\in I} \left\{\left(\bar{\boldsymbol{\varepsilon}}_e^{(j)}, \bar{\boldsymbol{\sigma}}_e^{\mathrm{new}} = \mathcal{H}\left(\Omega,\, \boldsymbol{u} = \bar{\boldsymbol{\varepsilon}}_e^{(j)} \cdot \boldsymbol{x}\right)\right)\right\},$$

  - set $j = j + 1$ and return to Step 2.

3 Numerical results

In this section, we evaluate the two-scale data-driven solver listed in Algorithm VI.3 by means of a two-dimensional macroscopic purely displacement-controlled problem. We consider a $l \times h = 20\,\text{mm} \times 50\,\text{mm}$ rectangular object with a circular hole in the thickness. Plane strain is assumed. The macroscopic geometry, boundary conditions and meshes of the sample are given in Fig. VI.6. We use two different meshes, composed of bi-linear quadrangular elements comprising 4 integration points each:

- the so-called coarse mesh consists of $N = 1092$ nodes, connected by 1015 elements, and thus comprises $M = 4060$ integration points;
- the so-called fine mesh comprises $N = 2495$ and 2375 bi-linear quadrangular elements, which amounts to $M = 9500$ integration points.

The Dirichlet boundary conditions are represented in Fig. VI.6 and write

$$\bar{\boldsymbol{u}}_a = \boldsymbol{0} \qquad \forall a \in [1\,..\,N] \text{ such that } \boldsymbol{x}_a = (x_a, 0) \tag{VI.38a}$$

$$\bar{\boldsymbol{u}}_a = \bar{u}_1^{\mathrm{D}} \boldsymbol{e}_1 + \bar{u}_2^{\mathrm{D}} \boldsymbol{e}_2, \qquad \forall a \in [1\,..\,N] \text{ such that } \boldsymbol{x}_a = (x_a, h) \tag{VI.38b}$$

with $\bar{u}_1^{\mathrm{D}} = 0.3\,\text{mm}$ and $\bar{u}_2^{\mathrm{D}} = 0.05\,\text{mm}$.

We study two different test cases:

- in a first example, the material is considered homogeneous. The stress-from-strain update Eq. (VI.24) is then simply reduced to the evaluation of a macroscopic constitutive model.
- in a second example, the material is a two-phases isotropic composite, thus requiring a computational homogenization of the microstructure.

3.1 Microscopically homogeneous test case

This example allows for (i) a quick assessment of the two-scale solver as the evaluation of the constitutive model is cost-less, (ii) an easy comparison with the standard FE reference solution, which does not require a FE^2 simulation.

In this example, we only consider the fine mesh.

3.1.1 Computational problem

The macroscopically homogeneous material is represented by an isotropic nonlinear elastic model of the form

$$\bar{\boldsymbol{\sigma}} = \kappa \operatorname{tr} \bar{\boldsymbol{\varepsilon}} \left(1 + \beta \left(\operatorname{tr} \bar{\boldsymbol{\varepsilon}}\right)^2\right) \boldsymbol{I} + 2\mu \left(1 + \beta \operatorname{dev} \bar{\boldsymbol{\varepsilon}} : \operatorname{dev} \bar{\boldsymbol{\varepsilon}}\right) \operatorname{dev} \bar{\boldsymbol{\varepsilon}}, \tag{VI.39}$$

with $\kappa = 20\,\text{MPa}$ the bulk modulus, $\mu = 9.23\,\text{MPa}$ the shear modulus and $\beta = 1 \times 10^4$ a nonlinear coefficient. The corresponding Young's modulus and Poisson's ratio are $E = 24\,\text{MPa}$ and $\nu = 0.3$.

The parameters of the data-driven solvers are as follows:

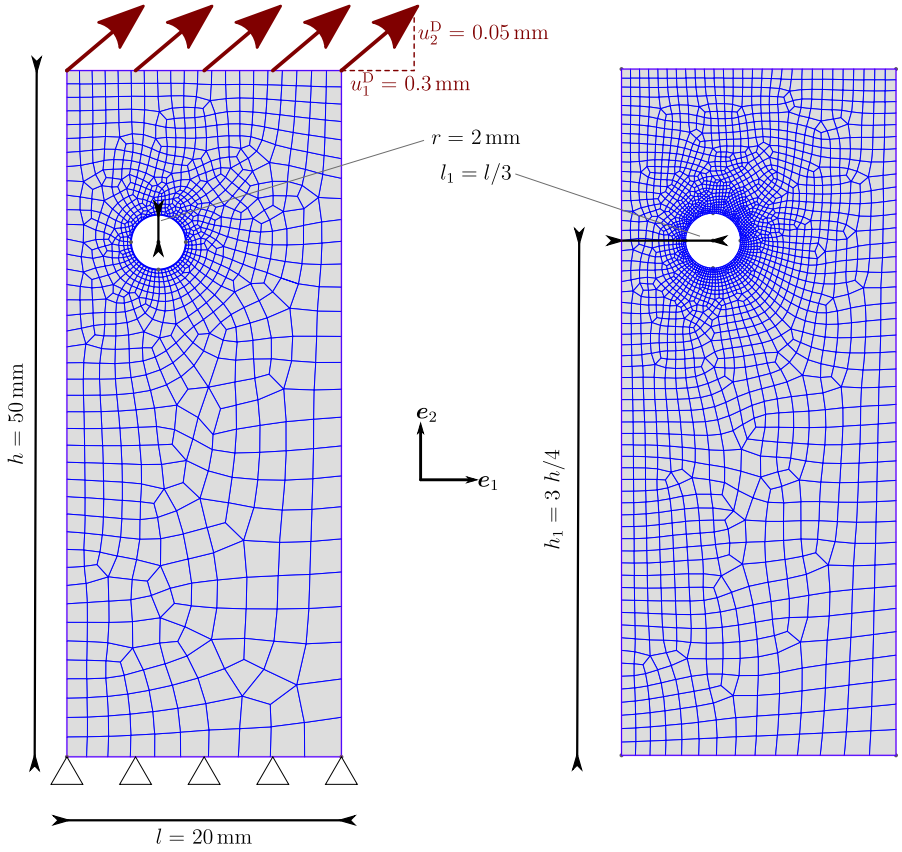


Figure VI.6 – Geometry, boundary conditions and mesh of the macroscopic sample.

- the $\mathbb{C}$ parameter is set to an isotropic linear elastic stiffness tensor $\mathbb{H}$, defined as in Eq. (VI.35). We compare the results obtained from two different $\mathbb{C}$:
  - $\mathbb{C}_1 = \mathbb{H}(1000E, \nu)$,
  - $\mathbb{C}_2 = \mathbb{H}(1000E, 0.8\,\nu)$.

  As recommended, the modulus of $\mathbb{C}$ in both cases is large and we evaluate the influence of the Poisson's ratio.

- the critical distance threshold d_c is defined with respect to $\mathbb{C}$ and the maximum desired gap $\Delta\bar{\boldsymbol{\varepsilon}} = 1 \times 10^{-6}$ between mechanical and material strains as

$$d_c = W(\Delta\bar{\boldsymbol{\varepsilon}}) = \frac{1}{2}(\mathbb{C}\Delta\bar{\boldsymbol{\varepsilon}})\Delta\bar{\boldsymbol{\varepsilon}} \tag{VI.40}$$

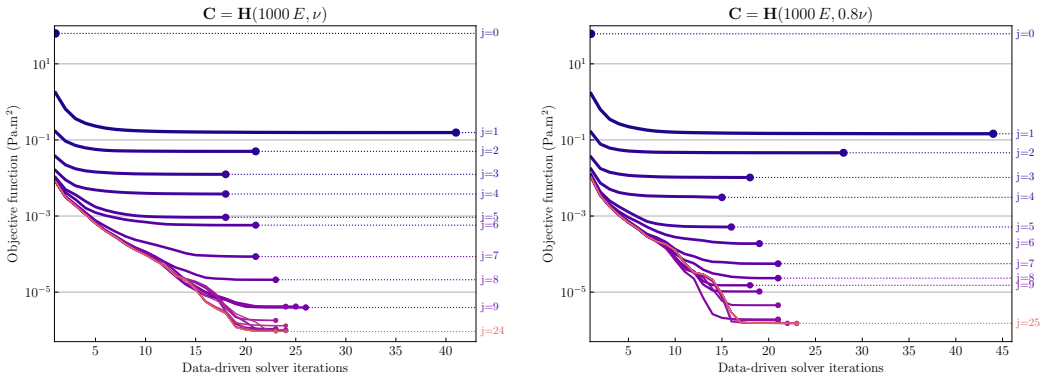
- the maximum number of points that can be added to the database at each iteration is set to $M_{\mathrm{lim}} = M/2$, thus restricting the number of constitutive model evaluations allowed.

3.1.2 Convergence analysis

Let us first evaluate the convergence of the successive macroscopic data-driven solvers. We recall that, at the end of the iteration k in Algorithm VI.1, the objective function writes

$$f^{(k)} = \sum_{e=1}^{M} w_e \; d_e \left(\left(\bar{\boldsymbol{\varepsilon}}_e^{(k)}, \bar{\boldsymbol{\sigma}}_e^{(k)} \right), \left(\bar{\boldsymbol{\varepsilon}}_{\mathrm{i}e}^{*,(k)}, \bar{\boldsymbol{\sigma}}_{\mathrm{i}e}^{*,(k)} \right) \right) \tag{VI.41}$$

As the material database is continuously enriched, we expect the value of the objective function (distance to the material data set) to decrease accordingly. This is demonstrated in Fig. VI.7 for both two-scale simulations using $\mathbb{C}_1$ and $\mathbb{C}_2$. Each solid line represent the convergence of f within one macroscopic data-driven simulation. The iteration j indicated at the end of the line correspond to the iteration in the two-scale data-driven solver Algorithm VI.3. The dot at the end of the line indicates the final value $f^{(k_{\max})}$ of the objective function. In both cases, the two-



(a) Solution achieved from $\mathbb{C}_1 = \mathbb{H}(1000E, \nu)$.

(b) Solution achieved from $\mathbb{C}_2 = \mathbb{H}(1000E, 0.8\nu)$.

Figure VI.7 – Convergence of the successive macroscopic data-driven solvers.

scale solvers required about 25 updates of the database. As expected, the value of the objective function is lower and lower as the material database is completed.

This is also confirmed from Fig. VI.8 where we plot the converged value $f^{(k_{\max}(j))}$ of the objective function with respect to the number of points in the material database $\mathcal{D}_{\mathrm{loc}}^{(j)}$, for each two-scale iteration j. As can be seen in Fig. VI.8, the distance between mechanical and material states drastically decreases with the number of material data points, in both cases. We remark that the $\mathbb{C}_1$ simulation gets to a lower value of the objective function with slightly less material data points than the $\mathbb{C}_2$ simulation.

Finally, we count the number of constitutive model evaluations throughout the simulation and compare it with the systematic evaluations made in FE^2 scheme. In Table VI.1 we report the total number of times the constitutive model was evaluated during the two-scale data-driven simulation. In this case, since the material is

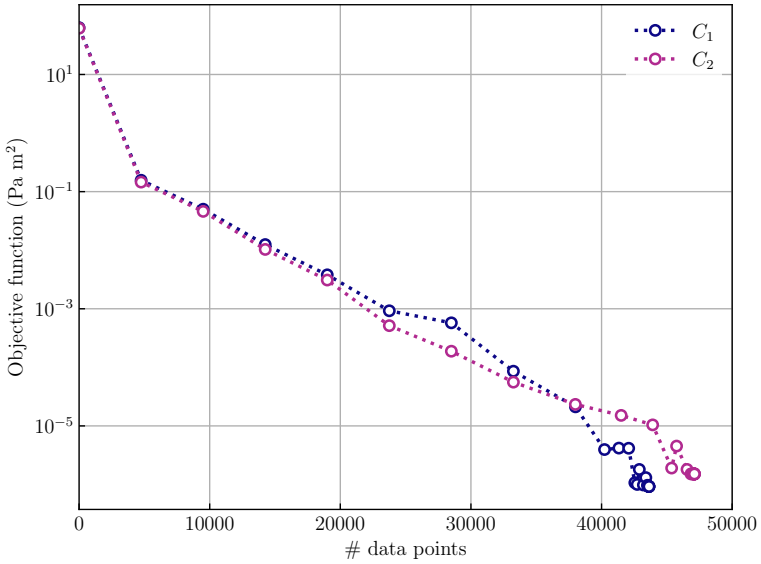


Figure VI.8 – Convergence of the two-scale data-driven solver with respect to the size of the material database $\mathcal{D}_{\text{loc}}$.

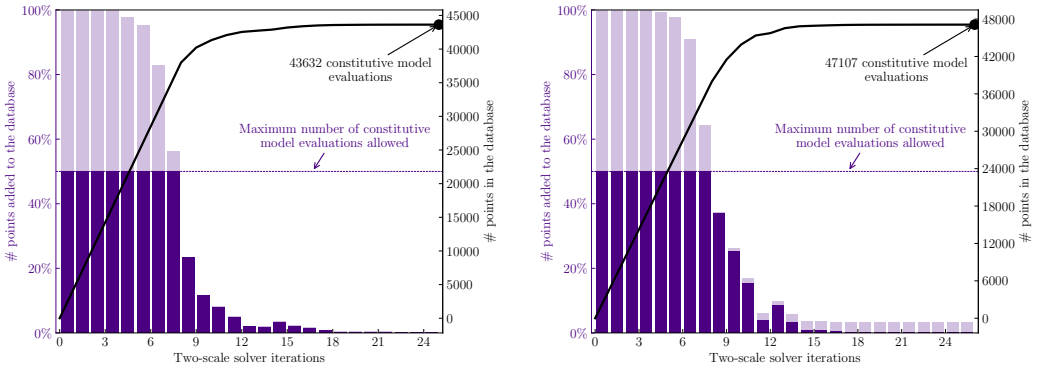
homogeneous, we compare with the number of stress evaluations in a standard FE computation: $N_{\text{stress eval.}} = N_{\text{iter.}} \times M$ where $N_{\text{it.}}$ is the number of iterations necessary to reach equilibrium within a given tolerance tol (here, a Newton-Raphson scheme was used). The number of constitutive evaluations is then at least two times

Table VI.1 – Number of constitutive model (stress) evaluations. The number of integration points is $M = 9500$.

Type of simulation	Number of stress evaluations
Standard FE simulation (tol $= 1 \times 10^{-5}$)	76 000
Standard FE simulation (tol $= 1 \times 10^{-12}$)	123 500
Data-driven simulation ($\mathbb{C} = \mathbb{C}_1$)	43 632
Data-driven simulation ($\mathbb{C} = \mathbb{C}_2$)	47 107

lower in the data-driven scheme. This result is very encouraging for two-scale simulations as, here, we did not take into account the additional microscopic evaluations required to compute the tangent operator in FE^2 schemes (see Eq. (VI.8)).

Finally, in Fig. VI.9 we give the number of points added to the database at each iteration of the two-scale solver. As aforementioned, the number of additions allowed is limited by $M_{\text{lim}} = M/2$. As can be seen from Fig. VI.9, the $\mathbb{C}_1$ simulation is more efficient that the $\mathbb{C}_2$ one. Note that at the end of the $\mathbb{C}_1$ (resp. $\mathbb{C}_2$) simulation there remain 29 (resp. 302) isolated mechanical states for which the corresponding material data point already exists in the database (not visible in the figure).



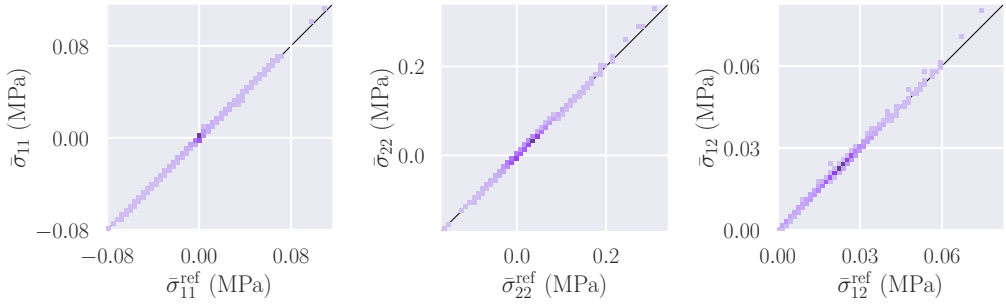
(a) Solution achieved from $\mathbb{C}_1 = \mathbb{H}(1000E, \nu)$ (b) Solution achieved from $\mathbb{C}_2 = \mathbb{H}(1000E, 0.8\nu)$

Figure VI.9 – Number of points added to the database at each iterations. The violet bins indicate the number of points actually added to the material database after each iteration. The black solid line represents the cumulative number of material data points in the database. The semi-transparent bins indicate the number of isolated mechanical states for which $d_e > d_c$, that were yet not added to the database for one of following reasons: (i) the limitation M_{lim} was triggered, they were thus rejected because more isolated points were prioritized, or (ii) the material data points already exist in the database (previously added).

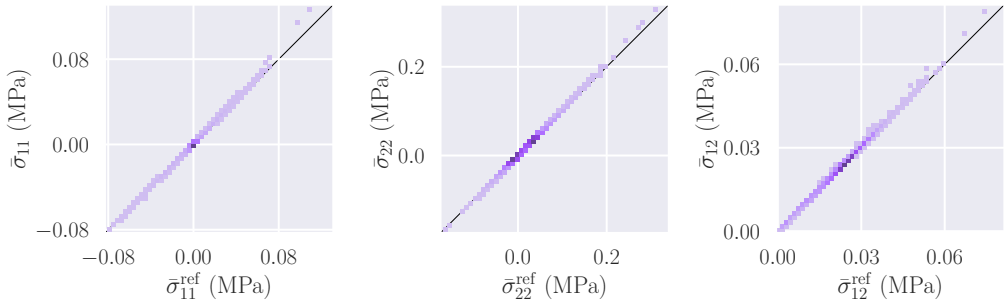
3.1.3 Strain and stress results

In Fig. VI.10, we compare the components of the macroscopic stress tensor with the FE reference solution, for both $\mathbb{C}_1$ and $\mathbb{C}_2$ simulations. The solid black line represents perfect identity. The data is binned in two-dimensional histograms: the darker the bin, the more points in the corresponding area of the plot. As can be seen in the figure, the majority of the stress data points are close to zero and very well predicted by the data-driven two-scale solver in both cases. The value of the Poisson's ratio used in the distance parameter $\mathbb{C}$ has then little influence on the prediction of the stress. In contrast, the values of the strain tensors are much more impacted, as can be seen from Fig. VI.11. In particular, $\bar{\boldsymbol{\varepsilon}}_{11}$ and shear $\bar{\boldsymbol{\varepsilon}}_{12}$ components are poorly predicted when $\mathbb{C} = \mathbb{C}_2$ (Fig. VI.10(b)), as compared with $\mathbb{C} = \mathbb{C}_1$ (Fig. VI.11(a)). In both cases, the largest errors occur for large (positive or negative) values of $\bar{\boldsymbol{\varepsilon}}_{11}$ and $\bar{\boldsymbol{\varepsilon}}_{12}$. In the $\mathbb{C}_1$ case however, the number of concerned points is small (light bins) and the error remains reasonable.

In addition, we report in Table VI.2 the strain and stress RMS percent errors for



(a) Solution achieved from $\mathbb{C}_1 = \mathbb{H}(1000E, \nu)$.



(b) Solution achieved from $\mathbb{C}_2 = \mathbb{H}(1000E, 0.8\nu)$.

Figure VI.10 – Comparison of the data-driven and reference solution for the stress.

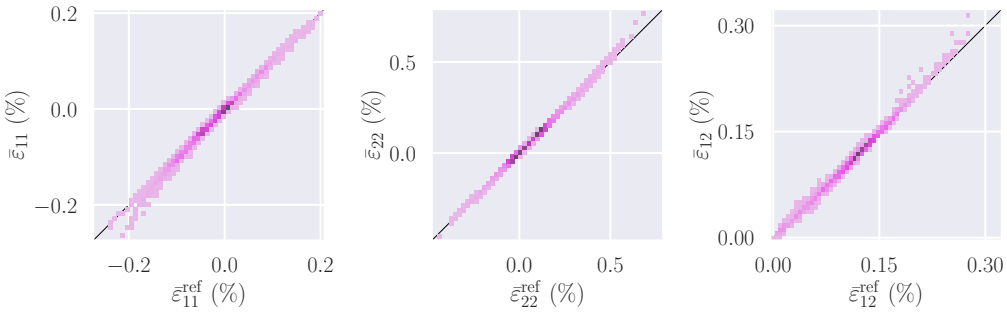
both $\mathbb{C}_1$ and $\mathbb{C}_2$ cases, re-defined as

$$\bar{\varepsilon}_{\text{RMS\%}} = \frac{\sum_{e=1}^{M} w_e \, W(\bar{\boldsymbol{\varepsilon}}_e - \bar{\boldsymbol{\varepsilon}}_e^{\text{ref}})}{\sum_{e=1}^{M} w_e \, W(\bar{\boldsymbol{\varepsilon}}_e^{\text{ref}})}$$
$$\bar{\sigma}_{\text{RMS\%}} = \frac{\sum_{e=1}^{M} w_e \, W^*(\bar{\boldsymbol{\sigma}}_e - \bar{\boldsymbol{\sigma}}_e^{\text{ref}})}{\sum_{e=1}^{M} w_e \, W^*(\bar{\boldsymbol{\sigma}}_e^{\text{ref}})},$$

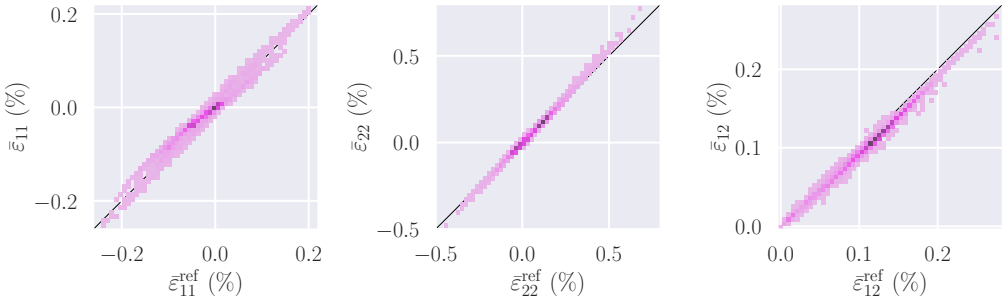
with W and W^* depending on $\mathbb{C}$ as in Eq. (VI.14). Again, when the value of the Poisson's ratio in the data-driven parameter $\mathbb{C}$ equals the one of the material, the accuracy with respect to the reference solution is higher.

Table VI.2 – Strain and stress RMS percent errors for the homogeneous test case.

Parameter $\mathbb{C}$	$\bar{\varepsilon}_{\text{RMS\%}}$	$\bar{\sigma}_{\text{RMS\%}}$
$\mathbb{C}_1$	4.45 %	1.33 %
$\mathbb{C}_2$	9.37 %	1.98 %



(a) Solution achieved from $\mathbb{C}_1 = \mathbb{H}(1000E, \nu)$.



(b) Solution achieved from $\mathbb{C}_2 = \mathbb{H}(1000E, 0.8\nu)$.

Figure VI.11 – Comparison of the data-driven and reference solution for the strain.

3.1.4 Summary

This first example has demonstrated that the zero initial database together with the update strategy provide very satisfactory results with little to no assumption on the material or the macroscopic response. In addition, the number of constitutive model evaluations is much lower that in a standard FE^2 scheme. It has also shown the influence of the parameters of the method:

- the maximum number of constitutive model evaluations was set to $M_{\mathrm{lim}} = M/2$, which proved relevant;
- the pseudo-stiffness $\mathbb{C}$ was set to an isotropic linear elastic stiffness tensor, for which the Poisson's ratio must correspond to the one of the (nonlinear) constitutive model.

Based on these encouraging results, we applied the method to a two-phases composite test case, as detailed next.

3.2 Two-phases composite test case

The macroscopic problem is the same as previously: the geometry and boundary conditions are the ones represented in Fig. VI.6. Only the material changes: we consider a two-phases composite.

In this example, we present the data-driven results only: we have not performed the corresponding FE^2 simulation.

3.2.1 Computational problem

The material is now a two-phases composite made of a stiff linear elastic circular inclusion embedded in a softer nonlinear elastic matrix. The former has a Young's modulus $E^{(i)} = 24\,\text{GPa}$ and a Poisson's ratio $\nu^{(i)} = 0.3$. The latter is represented by the same nonlinear constitutive model as in Eq. (VI.39):

$$\begin{aligned}\boldsymbol{\sigma}^{(m)} = \kappa^{(m)} \operatorname{tr} \boldsymbol{\varepsilon}^{(m)} \left(1 + \beta^{(m)} (\operatorname{tr} \boldsymbol{\varepsilon}^{(m)})^2\right) \boldsymbol{I} \\ + 2\mu^{(m)} \left(1 + \beta^{(m)} \operatorname{dev} \boldsymbol{\varepsilon}^{(m)} : \operatorname{dev} \boldsymbol{\varepsilon}^{(m)}\right) \operatorname{dev} \boldsymbol{\varepsilon}^{(m)}\end{aligned} \tag{VI.43}$$

with $\kappa^{(m)}$ and $\mu^{(m)}$ such that $E^{(m)} = 24\,\text{MPa}$ and $\nu^{(m)} = 0.3$, and with $\beta^{(m)} = 1 \times 10^4$. The geometry[2] and mesh of the RVE of the microstructure are presented in Fig. VI.12. The mesh consists of 235 nodes and 466 linear triangular elements

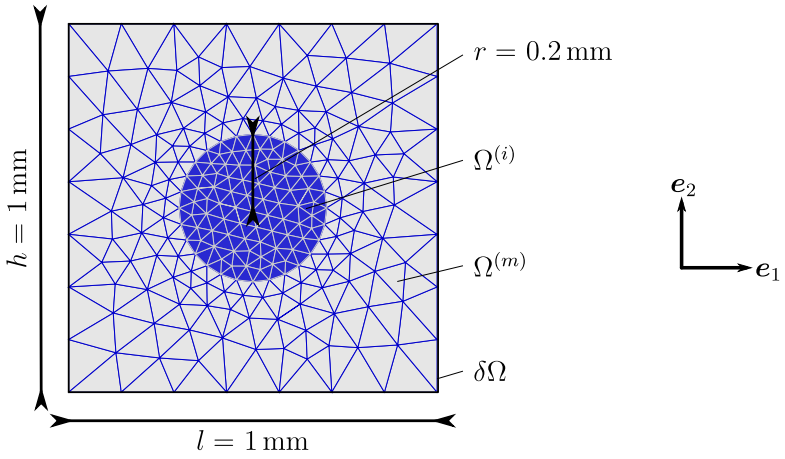


Figure VI.12 – Geometry and mesh of the microstructure.

[2]The microstructure used here is more a Representative Unit Cell (RUC) than a RVE *per se*. Such a RUC is typically representative of a periodic unidirectional composite with infinitely long fibers in the out-of-plane direction. For a distinction of RUC and RVE notions, see Matouš et al. (2017). In addition, we know that using linear displacement boundary conditions overestimates the linear elastic effective properties of such geometries (Bornert et al., 2001, Chapitre 2). In this preliminary study, we neglected these difficulties and selected this type of microstructure for its simplicity.

with 1 integration point each. Note that the mesh is deliberately coarse to reduce the computational cost of this preliminary study. The effective response of the microstructure is obtained from the resolution of the local BVP Eq. (VI.2) on the RVE $\Omega = \Omega^{(m)} \cup \Omega^{(i)}$ using the linear displacements boundary conditions Eq. (VI.4) on $\partial\Omega$.

The parameters of the data-driven solvers are as follows:

- as the inclusion is much more rigid than the matrix, the $\mathbb{C}$ parameter is set to the isotropic linear elastic stiffness tensor $\mathbb{H}(E^{(i)}, \nu^{(i)})$ of the inclusion. As recommended, the modulus of $\mathbb{C}$ is then large while the Poisson's ratio is consistent with the expected one of the composite material.
- the critical distance threshold d_c is defined as previously, with respect to $\mathbb{C}$ and the maximum desired gap $\Delta\bar{\boldsymbol{\varepsilon}} = 1 \times 10^{-6}$ between mechanical and material strains as $d_c = W(\Delta\bar{\boldsymbol{\varepsilon}})$.
- the maximum number of points that can be added to the database at each iteration is set to $M_{\text{lim}} = M/2$, as previously.

We conduct two successive two-scale simulations with our method:

1. the macroscopic simulation is first conducted on the macroscopic coarse mesh of Fig. VI.6, starting from the zero database $\mathcal{D}^{(0)} = \{(\mathbf{0}, \mathbf{0})\}$.
2. the simulation is then conducted on the macroscopic fine mesh of Fig. VI.6, starting from the final database of the previous computation $\mathcal{D}^{(0)}_{\text{finemesh}} = \mathcal{D}^{(j_{\max})}_{\text{coarsemesh}}$.

That way, we aim to assess one interesting feature of the method: contrary to standard FE^2, the microscale computations are not discarded and can be re-used for another simulation. Here, we only change the mesh resolution and not the boundary conditions but the latter could be studied in future work.

3.2.2 Convergence analysis

Let us first evaluate the convergence of the two-scale data-driven solvers for both macroscopic coarse and fine meshes. In Fig. VI.13, we give the converged value $f^{(k_{\max}(j))}$ of the objective function Eq. (VI.41) as a function of the number of points in the material database $\mathcal{D}^{(j)}_{\text{loc}}$. As before, the overall distance between mechanical and material states decreases very rapidly. In addition, we see from Fig. VI.13 that the simulation with the fine mesh required few additional material data points to converge to the same values of objective function. Indeed, during the first simulation with the coarse mesh, a total of 17 271 microstructure computations have been conducted within 23 iterations. Starting from $\mathcal{D}^{(0)}_{\text{loc,fine}} = \mathcal{D}^{(23)}_{\text{loc,coarse}}$ in the second simulation, 34 094 additional microscopic evaluations have been made within 21 iterations. These figures are represented in Fig. VI.14. As expected, the number of additions to the material database decreases with the iterations in both cases.

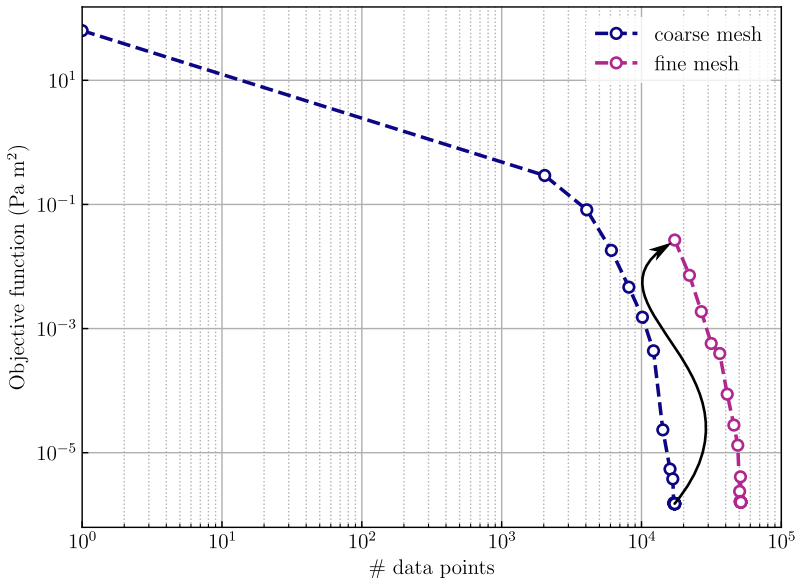
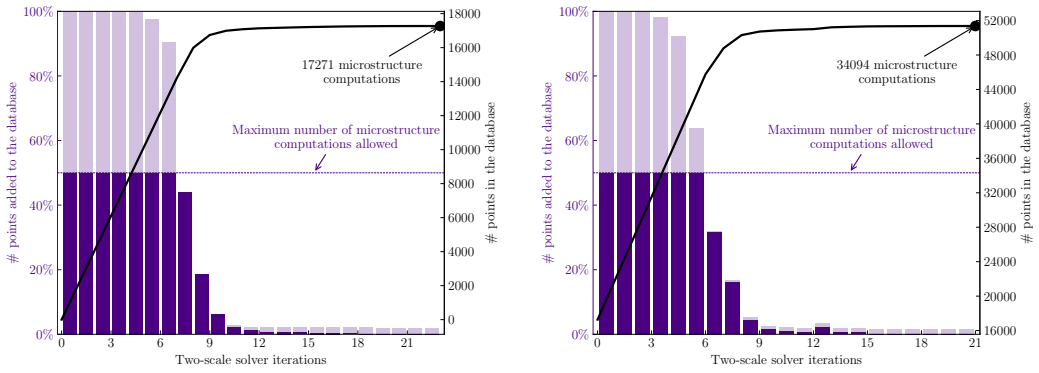


Figure VI.13 – Convergence of the objective function with respect to the size of the material database for the two-phases composite problem.



(a) Simulation on the macroscopic coarse mesh

(b) Simulation on the macroscopic fine mesh

Figure VI.14 – Number of point added to the material database during simulations on the two-phases composite.

Interestingly, the number of microstructure computations required by the fine mesh is higher than that of the coarse mesh, even if the simulations starts with *a priori* a suitable material database. This indicates that the finer the mesh the denser the material database needs to be: indeed, when refining the mesh, the region of local phase space explored by the macroscopic mechanical response is enlarged due to strain and stress concentrations in certain areas of the mesh.

Note that the microstructure computations conducted for the material database used for the fine mesh were partly off-line and on-line. To honestly compare with

a FE^2 scheme, we must take into account both parts. In Fig. VI.15, we compare the number of microscopic evaluations between data-driven and standard multi-level simulations, using Eq. (VI.8), for both meshes. We see that, in the present case, the data-driven solver outperforms the standard FE^2 solver right from the second iteration.

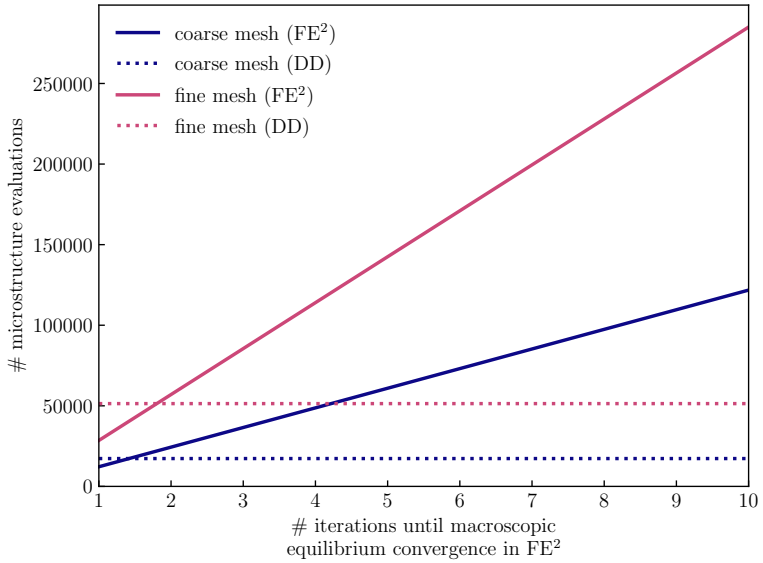


Figure VI.15 – Number of microstructure computations depending on the solver.

Finally, we present in Fig. VI.16 the total number of microstructure computations conducted in each integration point of the mesh throughout the iterations. As can be seen from Figs. VI.16 and VI.17, the area with the highest number of microscopic evaluations coincide with the area of stress concentration.

3.2.3 Strain and stress results

In absence of a FE reference solution to compare with, we give in Fig. VI.17 the strain and stress components achieved from the two scale data-driven simulation only. The results are qualitatively reasonable given the loading conditions: positive shear state in the structure and opposite tension states around the hole. In addition, the fields are rather smooth, indicating good convergence with respect to mesh size.

As pointed out by Xu et al. (2020), the interesting feature in data-driven multiscale schemes is that the microscopic states are stored by default. Examining the material strain state $\bar{\boldsymbol{\varepsilon}}^*_{\mathsf{ie}}$ of a given macroscopic integration point, we then have access to the corresponding microscopic response. In Fig. VI.18, we represent the microscopic equivalent strain ε_{eq} and stress σ_{eq} corresponding to a macroscopic point

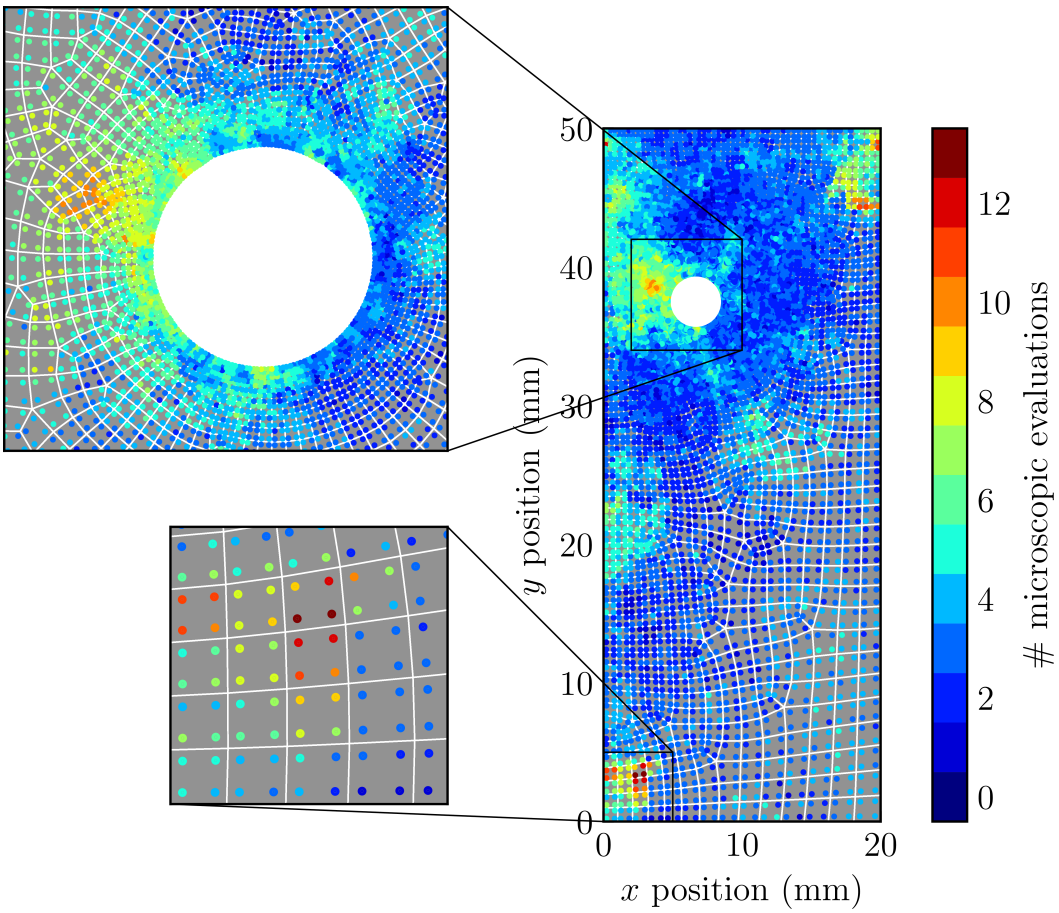


Figure VI.16 – Number of microstructure computations at each macroscopic integration point.

around the hole. The in-plane equivalent strain and stress norms write:

$$\varepsilon_{\text{eq}} = \frac{2}{3}\sqrt{3\varepsilon_{12}^2 + (\varepsilon_{11} - \varepsilon_{22})^2} \tag{VI.44a}$$

$$\sigma_{\text{eq}} = \frac{\sqrt{6}}{2}\sqrt{3\sigma_{12}^2 + (\sigma_{11} - \sigma_{22})^2} \tag{VI.44b}$$

At the price of one other microscopic computation, we can also evaluate the microscopic strain and stress fields obtained from the corresponding macroscopic *mechanical* strain state $\bar{\boldsymbol{\varepsilon}}_e$. In Fig. VI.19, we give the maximum error norms between the two types of microscopic fields, defined as

$$\Delta\varepsilon_{\max} = \|\boldsymbol{\varepsilon}_{\text{mat}} - \boldsymbol{\varepsilon}_{\text{mech}}\|_{\max}, \tag{VI.45a}$$

$$\Delta\sigma_{\max} = \|\boldsymbol{\sigma}_{\text{mat}} - \boldsymbol{\sigma}_{\text{mech}}\|_{\max}, \tag{VI.45b}$$

where the max norm of tensor $\boldsymbol{A}$ writes $\|\boldsymbol{A}\|_{\max} = \max_{i,j} |A_{ij}|$, $\boldsymbol{\varepsilon}_{\text{mat}}$ (resp. $\boldsymbol{\varepsilon}_{\text{mech}}$) denotes the microscopic strain field obtained from the macroscopic material (resp.

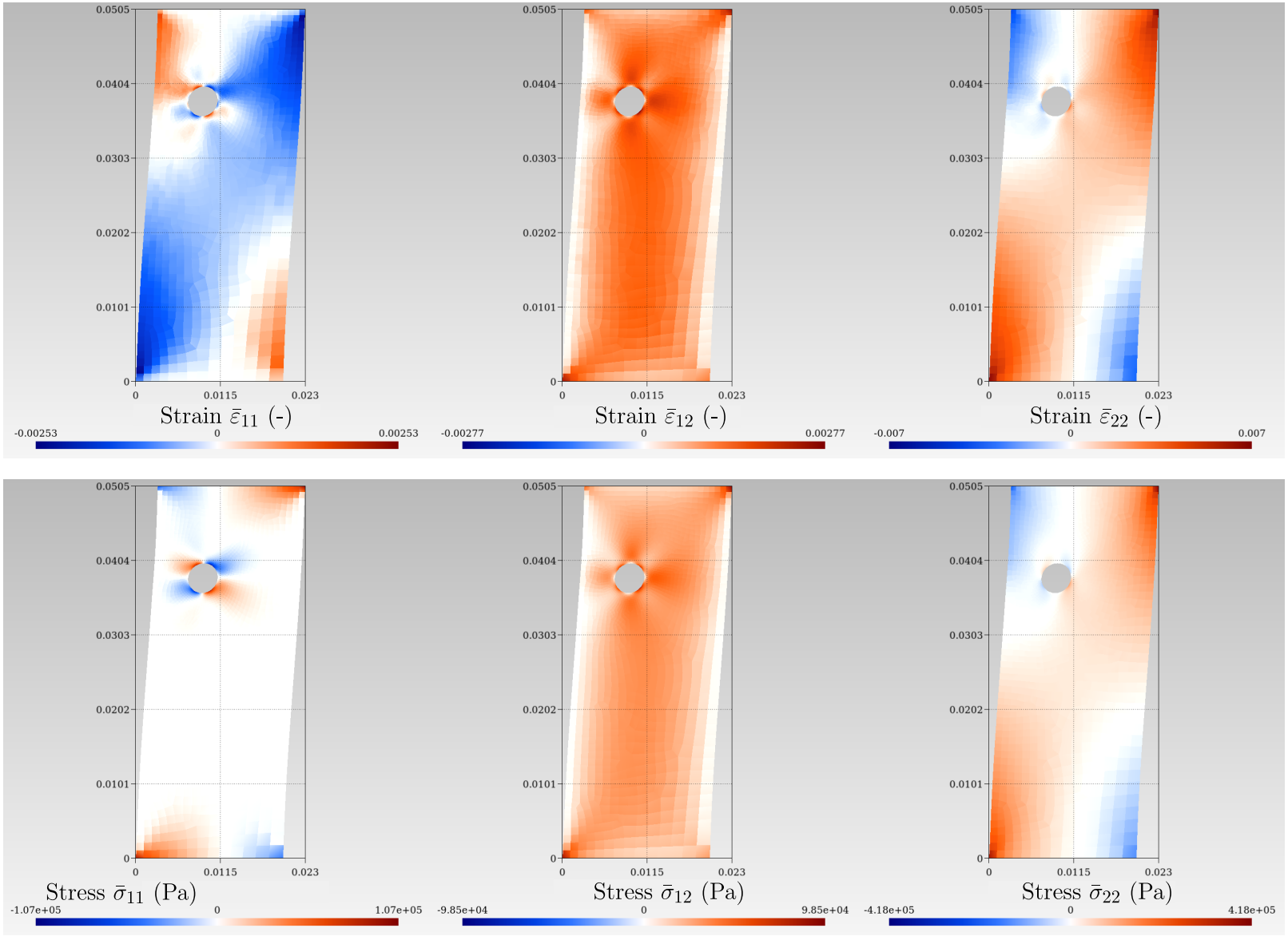


Figure VI.17 – Macroscopic strain and stress fields on the exaggerated deformed configuration (displacement factor $\times 10$).

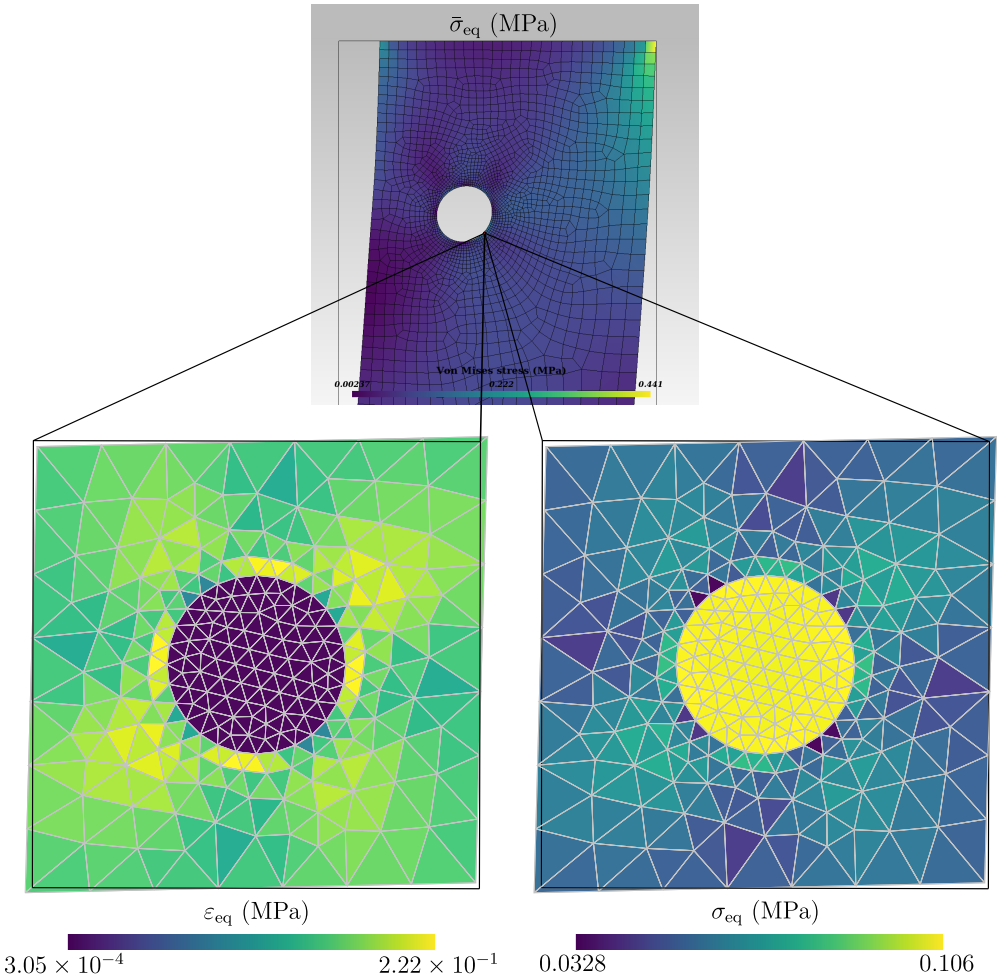


Figure VI.18 – Microscopic strain and stress fields at a macroscopic point on the edge of the hole. The deformed configuration of the microstructure is exaggerated by a factor 20.

mechanical) strain $\bar{\boldsymbol{\varepsilon}}^*_{\text{ie}}$ (resp. $\bar{\boldsymbol{\varepsilon}}_e$), and likewise for the microscopic field $\boldsymbol{\sigma}_{\text{mat}}$ (resp. $\boldsymbol{\sigma}_{\text{mech}}$). For the integration point considered, the macroscopic material and mechanical strain tensors are very close:

$$\bar{\boldsymbol{\varepsilon}}_{\text{mat}} = \begin{pmatrix} 0.037\,681\,07 & 0.122\,335\,27 \\ 0.122\,335\,27 & -0.010\,264\,29 \end{pmatrix} \% \tag{VI.46a}$$

$$\bar{\boldsymbol{\varepsilon}}_{\text{mech}} = \begin{pmatrix} 0.037\,668\,51 & 0.122\,334\,68 \\ 0.122\,334\,68 & -0.010\,273\,72 \end{pmatrix} \%. \tag{VI.46b}$$

Consequently, the microscopic field are also very similar, as can be seen in Fig. VI.19. This indicates that the microscopic response is reliable in a post-processing analysis.

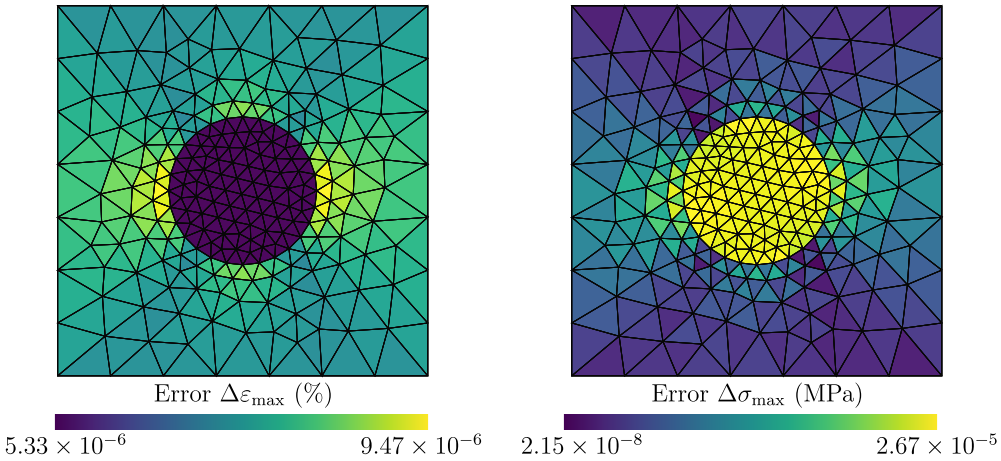


Figure VI.19 – Errors between the macroscopic strain and stress fields achieved from the macroscopic material and mechanical strain tensor.

4 Conclusion

In this final chapter, we proposed a first attempt in accelerating multiscale computations with the aid of DDCM solvers, in small strain. For heterogeneous materials, the material database used in DDCM is constituted from the computational homogenization of the effective response of the microstructure. Typically, as in standard FE^2 schemes, the macroscopic stress $\bar{\boldsymbol{\sigma}}$ is obtained from the volume average of the microscopic stress field $\boldsymbol{\sigma}$, solution of a local BVP on the RVE of the microstructure, with boundary conditions defined by the macroscopic strain $\bar{\boldsymbol{\varepsilon}}$ (Feyel, 1999). Sampling the macroscopic strain space with a regular grid, Xu et al. (2020) thus produced material databases suitable for single scale DDCM computations at the macroscale, by performing a microscopic computation per point in the grid. Their contribution is the first, to the best of our knowledge, to bring DDCM and computational homogenization together. In their approach, all micro-evaluations are performed off-line, which seems appealing at first glance. However, the range and density of the grid must be guessed beforehand, based on the expected response of the macrostructure under consideration. A large number of microscopic evaluations are then likely to be performed in no use for the current problem.

In contrast, our method follows the philosophy of standard FE^2: the microscopic evaluations are performed only when necessary, with the loading defined by the current macroscopic mechanical response. To reduce the amount of local computations and avoid redundance, we make advantage of the data-driven paradigm: the average microscopic response can be stored in phase space and the distance-minimizing solver provides information on the quality of the database. We then embedded the original data-driven solver in an iterative process which consists in adaptively updating the material database at each iteration. Specifically, the material database

is enriched wherever its sparsity has been measured: the minimal distance between a mechanical state and its associated material state indicates whether or not the material database is sufficiently dense in the corresponding area of the phase space. On that basis, we developed an efficient update strategy which allows for starting with a quasi-empty database: the process starts with only the zero strain-stress state. In particular, we discussed the application of our strategy depending on the macroscopic boundary and loading conditions in the FE setting. Whether the macro-simusimulation displacement- or force-controlled, the new material data point is determined from the mechanical strain or the mechanical stress respectively.

The thus derived two-scale data-driven FE^2 solver has been applied to a two-dimensional problem, in plane strain. In a first test case, the structure was made of a microscopically homogeneous material. Thanks to an easy comparison with the FE reference solution and cost-less constitutive evaluations, we showed the robustness of our method. In a second case, we considered a two-phases nonlinear composite. We demonstrated that the number of micro-evaluations is greatly reduced with respect to the theoretical number in standard FE^2. Part of the reduction comes from an interesting feature of data-driven solvers: there is no requirement for the computation of a material tangent operator, as the material nonlinearity is not specifically being dealt with in DDCM. In addition the macroscopic and microscopic response was proved to be qualitatively satisfactory.

To conclude, from this preliminary study on coupling computational homogenization and DDCM we found very encouraging results to explore further the construction of multiscale data-driven solvers. Of course, our update strategy was determined from a trial-and-error approach and must certainly be consolidated with more rigorous considerations. In addition, the method has yet to be compared with FE^2 results. But we are confident that this kind of approaches offer a wide field of applications for the DDCM paradigm. As a perspective, the same rationale can be followed for the data-driven computational homogenization of nonlinear materials in large strain, using the finite strain data-driven solvers developed in this thesis. Finally, one could even imagine a fully data-driven multiscale solver, where both nested BVPs are solved using DDCM instead of standard FE method for the microscale, as was done here.

Conclusion and perspectives

Conclusion and perspectives

The present thesis has explored the field of *data-driven computational mechanics*, in which the constitutive model of the material response is replaced by a database of raw constitutive points, namely strain-stress pairs in the case of elastic behavior. With this new computing paradigm, appropriate data coverage of the strain-stress space is critical to obtain a predictive mechanical response of the structure. We have addressed this question in two particular cases, presented in two parts of the manuscript: (i) the extension of the approach to finite strain elasticity, and (ii) the application of the method to multiscale simulation of structures made of heterogeneous materials.

Before addressing these two aspects, we have recalled in Chapter I the governing equations and numerical challenges met in nonlinear computational mechanics. In particular, we have highlighted that several work conjugate pairs of strain and stress tensors can be used to describe the static equilibrium of a deformable body, yielding different, yet equivalent, formulations of the classical solid mechanics BVP. In this opening chapter, we also showed that computational mechanics and data science are intertwined at the level of material modeling, which motivates the development of data-driven solvers. The data-driven computing paradigm introduced by Kirchdoerfer and Ortiz (2016) and consolidated in Conti et al. (2018) was then extensively recalled in Chapter II. We insist on two specific features of the method, driving the two aforementioned studies. First, the data-driven solvers reformulate the mechanical problem as a constrained distance-minimization problem coupled with a combinatorial optimization: the mechanical state field (representing the admissible mechanical response of the body) *must verify* both compatibility and equilibrium equations, while minimizing the distance to the material state field (representing the material response) which takes value in a *discrete* set of material data points. This challenging numerical problem is addressed with dedicated schemes, which need be adapted to finite strain. Second, the final distance between the mechanical and material states indicates regions of the strain-stress space requiring higher density of material data points. These regions can be adaptively enriched from numerical simulations at finer scales in the context of heterogeneous materials.

The largest part of the manuscript (Part B) was dedicated to the study of finite strain data-driven computational mechanics. Based on the work of Nguyen and Keip (2018) and Conti et al. (2020), we compared their Lagrangian and nominal formulations respectively. The former relies on the Green-Lagrange strain–second Piola-Kirchhoff stress symmetric tensors as work conjugate pairs to formulate the data-driven problem; the latter involves the deformation gradient–first Piola-Kirchhoff stress non symmetric tensor pairs. In Chapter III, we proposed a finite element solver for both, taking into account their respective implications. We then conducted in Chapter IV a thorough study on material database generation and control of data coverage with mechanically meaningful tools, focusing on the Lagrangian formula-

tion only. Finally, a complex three-dimensional mechanical problem was considered in Chapter V to evaluate the performance of both finite strain data-driven solvers.

In the last Part C and closing Chapter VI of this thesis, we have proposed a first attempt to develop a multiscale data-driven solver. Here, the material database consists of macroscopic strain-stress pairs, obtained from finite element computational homogenization of the microstructure response. The original data-driven solver of Kirchdoerfer and Ortiz (2016) is then used as a black-box in an iterative process. At a fixed iteration, it provides (i) the macroscopic mechanical state of the body, for a given database of homogenized strain-stress pairs, and (ii) the error measure between the current mechanical state and the current database. The material database is subsequently enriched, on-the-fly, from appropriately chosen microscopic evaluations, in the sparsest regions of the macroscopic strain-stress space only.

The major contributions of the present thesis are highlighted in the following.

- **We have developed two finite element data-driven solvers for finite strain elasticity**, with full disclosure on the implementation, in concern of open science: we provided extensive algorithms for both Lagrangian (Nguyen and Keip, 2018) and nominal (Conti et al., 2020) formulations. If the Lagrangian solver is essentially a consolidation of the one presented in Nguyen and Keip (2018), **the nominal solver is an original contribution**, to the best of our knowledge. Both rely on the alternated minimization but differ in their resolution method of the constrained minimization on the mechanical states:

  - The Lagrangian solver makes use of the same method of Lagrange multipliers as the one presented in small strain to enforce equilibrium (Kirchdoerfer and Ortiz, 2016). This yields two nonlinear coupled systems of equations that are solved using a Newton-Raphson procedure (Nguyen and Keip, 2018). The size of the linearized system is then $(2n_{\text{dof}})^2$, with n_{dof} the total number of degrees of freedom in the finite element mesh.
  - The nominal solver requires an additional algorithmic layer: the mechanical constraints are enforced with an augmented Lagrangian approach, which solves a series of unconstrained minimization problems, defined from the Lagrangian of the objective function, augmented with a penalty term. This approach again yields two nonlinear coupled systems of equations addressed with a Newton-Raphson scheme. The size of the linearized system is then $(N \times n + n_{\text{states}})^2$, with N the number of nodes in the mesh, n the dimension and $n_{\text{states}} = M \times n^2$ the number of stress unknowns at M integration points[3].

[3] As a concrete example, let us consider the two-dimensional mesh ($n = 2$) of the membrane with a hole and three-dimensional mesh ($n = 3$) of the T-shaped column presented in this thesis. The former (resp. the latter) contains $N = 1092$ (resp. $N = 925$) nodes and $M = 4060$ (resp. $M = 4608$) integration points. The boundary conditions yields a total number of degrees of freedom

The major difference between these two formulations can be summarized as follows: while there is **no mathematical proof of the existence of solutions to the Lagrangian data-driven problem, contrary to its nominal counterpart** (Conti et al., 2020), the implementation of the corresponding finite element solver is easier, **the robustness is higher** and **the computational cost is much lower**.

- We showed that, in practice, **the conservation of angular momentum in finite strain must be explicitly enforced** in the minimization of the mechanical state, whether working *ad-hoc* with symmetric stress tensors in the Lagrangian formulation or handling the rotational equilibrium equation as a minimization constraint in the nominal formulation.

- We have developed a dense sampling method which allows for generating **tailored material databases** from a regular distribution of points in the high-dimensional strain space, with acute control on the amplitudes and modes of deformation. From comparing this (almost) infinitely rich database with a specifically designed one, we showed that **material databases resulting from *importance sampling* of the strain-stress space are more performing than those resulting from *dense sampling***, as they require far fewer data points while providing comparable mechanical responses.

- **We have performed the first data-driven computation of a complex three-dimensional problem in finite strain**, using the Lagrangian formulation. This example revealed the challenge of data coverage in this case, where the strain-stress space is of dimension $6 + 6 = 12$.

- Finally, **we provided a prototype multiscale data-driven solver in small strain nonlinear elasticity**, for accelerating and reducing the cost of multilevel computational homogenization schemes, such as FE^2.

Minor contributions of the present thesis include:

- a study on the initialization of the material states in the alternated minimization and its impacts on the convergence of the solver. We showed that the most reasonable choice is to initialize the material states to the zero strain-stress state, so that the first iteration boils down to a linear elastic estimate of the mechanical response, from the boundary and loading conditions only. Otherwise, non-zero initial material states act as random pre-strain and pre-stress conditions on the structure.

- we used efficient tree-based algorithms to search for optimal material states in the material database. In particular, we provided the transformation of strain-stress tensor pairs into standard Euclidean vectors, necessary to use

of $n_{\text{dof}} = 2120$ and $n_{\text{dof}} = 2550$ for each problem respectively. The system of the Lagrangian and nominal systems are then respectively 4240 *vs.* 18 424 for the two-dimensional problem and 5100 *vs.* 44 247 for the three-dimensional problem.

open-source interfaces to these nearest-neighbor search algorithms (Pedregosa et al., 2011).

We organize the prospects of the present work around three central challenges: the enhancement of data-driven solvers, the acquisition of material data, and multiscale data-driven simulation.

1. All studies conducted in the present thesis were limited by the alternated minimization, which we know is sub-optimal. The biggest challenge in developing more robust data-driven solvers is then to find other heuristics. The problem is particularly acute in finite strain as the minimization constraints are nonlinear. Possible directions of research can be to extend the entropy-maximizing scheme of Kirchdoerfer and Ortiz (2017) to finite strain or to investigate MIQP formulations, as proposed by Kanno (2019). In the shorter term, the shortcomings of the nominal solver in the three-dimensional example studied in this thesis should be elucidated and fixed.

2. Future work should focus on the acquisition of material data points and their use in the data-driven computing paradigm. In particular, future work should focus on using the material database obtained from the model-free stress identification technique developed by Leygue et al. (2018) and successfully applied to elastomers by Dalémat et al. (2019) in finite strain data-driven simulations. In the short term, care should be taken to define a training set used in data-driven identification, and a validation set used in data-driven computation, based on the experimental testing and numerical simulation of two different structures made of the same material. In the long term, the standardization of this type of approaches can provide an integrated framework of model-free identification of the material response used in data-driven prediction of the mechanical response of complex structures made of complex materials.

3. We showed in the last part of this thesis that material databases can be appropriately generated from numerical simulations at the finer scale, with a reasonable computational cost. The proposed prototype data-driven multiscale solver should be generalized to adaptively take into account any kind of external loading conditions, as the present solver was developed and tested for displacement-controlled simulations only. In addition, the update strategy of the material database could be more carefully elaborated, based on a more physically meaningful selection of regions requiring additional data resolution. Nevertheless, we believe that bringing the data-driven computing paradigm and multiscale simulation together offers very interesting industrial applications.

Appendices

Appendix A

Analytical solution for a one-dimensional bar

1 Small strain solution

1.1 Formulation of the problem

Let us consider the elementary problem represented in Fig. A.1 : an elastic one-dimensional bar, of length l and sectional area a is loaded with a system of stiffness k subject to a displacement u_0. The linearized kinematics assumption hold; the longitudinal cauchy stress σ and the longitudinal linearized strain ε are assumed homogeneous in the bar. They must verify the following equations:

$$\begin{cases} \varepsilon = \frac{u}{l} & \text{(compatibility)}, \\ \sigma = \frac{k}{a}\left(u_0 - u\right) & \text{(equilibrium in the bulk and on the loaded edge)}. \end{cases} \tag{A.1}$$

The *mechanical state* of the bar $z = (\varepsilon, \sigma)$ thus belong to a so-called *constraint set* of mechanically admissible strain-stress states $\mathcal{E} = \{z = (\varepsilon, \sigma)\,|\sigma = K(\varepsilon_0 - \varepsilon)\}$ with $K = \frac{kl}{a}$ and $\varepsilon_0 = \frac{u_0}{l}$. It is a subset of the so called *phase space* $\mathcal{Z} = \{(\varepsilon, \sigma)\}$ which collects all possible strain-stress pairs. Here, $\mathcal{Z} \subseteq \mathbb{R}^2$.

The classical elastic constitutive model $\check{\sigma}(\varepsilon)$ is a (nonlinear) bijective relationship which yields a constitutive manifold $\check{\mathcal{D}}$ in phase space:

$$\check{\mathcal{D}} = \{(\varepsilon, \sigma = \check{\sigma}(\varepsilon))\} \subset \mathcal{Z} \tag{A.2}$$

In the classical approach, the solution of the BVP is determined by the strain-stress state verifying both mechanical admissibility Eq. (A.1) and the constitutive model, *i.e.* it is found at the exact intersection $\mathcal{E} \cap \check{\mathcal{D}}$. Let us now assume that the constitutive model is unknown. Instead, the material response is represented by m material data points collected into a so-called material data set

$$\mathcal{D} = \{z_i^* = (\varepsilon_i^*, \sigma_i^*) \;|\; i = 1, \cdots, m\}\,; \tag{A.3}$$

the intersection $\mathcal{E} \cap \mathcal{D}$ is then most likely to be empty. The data-driven approach, illustrated in Fig. A.1 then consists in relaxing the intersection with the constitutive model by reformulating the problem as a minimization problem of a distance d between the two subspaces $\mathcal{E}$ and $\mathcal{D}$:

$$\min_{z \in \mathcal{E}} \min_{z_i^* \in \mathcal{D}} d^2(z, z_i^*), \tag{A.4}$$

where the distance d is defined by

$$\forall z \in \mathcal{E},\ \forall z_i^* \in \mathcal{D},\ d^2(z, z_i^*) = \frac{1}{2} C \left(\varepsilon - \varepsilon_i^*\right)^2 + \frac{1}{2C} \left(\sigma - \sigma_i^*\right)^2, \tag{A.5}$$

with C a scalar parameter used to scale strain and stress values. The data-driven

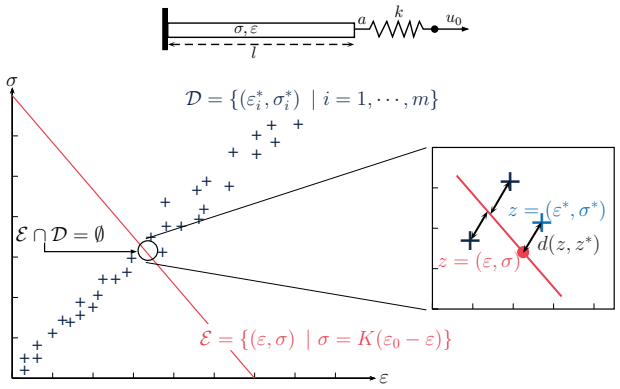


Figure A.1 – The data-driven solution is the mechanical state $z \in \mathcal{E}$ which is closest to the material data set $\mathcal{D}$, with associated material state z^*. Inspired from Kirchdoerfer and Ortiz (2017).

problem also writes as a *constrained* double minimization problem:

$$\min_{z \in \mathcal{Z}} \min_{z_i^* \in \mathcal{D}} \frac{1}{2} C \left(\varepsilon - \varepsilon_i^*\right)^2 + \frac{1}{2C} \left(\sigma - \sigma_i^*\right)^2, \tag{A.6a}$$

$$\text{subject to } z \in \mathcal{E} \ : \quad \sigma = K(\varepsilon_0 - \varepsilon). \tag{A.6b}$$

1.2 Resolution

The double minimization Eq. (A.6) couples continuous-valued functions z and discrete-valued functions z_i^*. To address this complex problem, Kirchdoerfer and Ortiz (2016) proposed an alternated minimization scheme : the mechanical state z is first determined from a fixed material state z^*, and then the material state z^* is updated as the closest material data points to z, selected from the material data set. For the very simple problem under consideration herein, we show that the resolution boils down to one iteration of the alternated minimization. We thus demonstrate that there exists a unique, semi-analytical solution.

Let $z^* = (\varepsilon^*, \sigma^*) \in \mathcal{D}$ be a fixed material state. Eq. (A.6) is solved using a method of Lagrangian multipliers. The problem is then reformulated as the following

stationary problem, for all $z = (\varepsilon, \sigma) \in \mathcal{Z}$:

$$\delta \left(\frac{1}{2} C \left(\varepsilon - \varepsilon^* \right)^2 + \frac{1}{2C} \left(\sigma - \sigma^* \right)^2 - \eta \left(\sigma - K \left(\varepsilon_0 - \varepsilon \right) \right) \right) = 0 \tag{A.7a}$$

$$\delta\varepsilon \Rightarrow \quad C(\varepsilon - \varepsilon^*) - \eta K = 0 \tag{A.7b}$$

$$\delta\sigma \Rightarrow \quad \frac{1}{C}(\sigma - \sigma^*) - \eta = 0 \tag{A.7c}$$

$$\delta\eta \Rightarrow \quad \sigma - K(\varepsilon_0 - \varepsilon) = 0 \tag{A.7d}$$

Manipulating Eq. (A.7) yields

$$d^2 = \frac{1}{2} C \left(\varepsilon - \varepsilon^* \right)^2 + \frac{1}{2C} \left(\sigma - \sigma^* \right)^2 = \frac{C}{2(K^2 + C^2)} \left(K \left(\varepsilon_0 - \varepsilon^* \right) - \sigma^* \right), \tag{A.8}$$

or equivalently

$$\left(\frac{\varepsilon - \varepsilon^*}{e} \right)^2 + \left(\frac{\sigma - \sigma^*}{s} \right)^2 = 1, \tag{A.9a}$$

with

$$e^2 = C^{-1} l_c^2 \text{ and } s^2 = C l_c^2 \tag{A.9b}$$

where

$$l_c^2 = \frac{C}{K^2 + C^2} \left(K \left(\varepsilon_0 - \varepsilon^* \right) - \sigma^* \right)^2. \tag{A.9c}$$

Eq. (A.9) indicate that the mechanical state solution $z = (\varepsilon, \sigma)$ belongs to an ellipse $\mathcal{C}(z, z^*)$ in the (ε, σ) plane, of center $(\varepsilon^*, \sigma^*)$and semi-axes e and s defined in Eq. (A.9b). The aspect ratio s/e of the ellipse is then entirely determined by the parameter C.

Since the mechanical state solution $z \in \mathcal{E}$ also verifies compatibility and equilibrium equations, the ellipse $\mathcal{C}(z, z^*)$ is tangent to the constraint set $\mathcal{E}$ at point z. The mechanical state solution $z \in \mathcal{E}$ then results from:

$$\forall z^* \in \mathcal{D}, \ \arg\min_{z \in \mathcal{E}} d^2(z, z^*) = \mathcal{C}(z, z^*) \cap \mathcal{E}. \tag{A.10}$$

Moreover, note that from Eq. (A.8) and Eq. (A.9b):

$$\forall z \in \mathcal{E}, \ \min_{z^* \in \mathcal{D}} d^2(z, z^*) = \min_{z^* \in \mathcal{D}} 2\pi d^2(z, z^*) = \min_{z^* \in \mathcal{D}} \pi e s = \min_{z^* \in \mathcal{D}} \text{area}(\mathcal{C}(z, z^*)) \tag{A.11}$$

From Eqs. (A.10) and (A.11), the data-driven solution of the present BVP is then the mechanical state $z = (\varepsilon, \sigma)$, defined as the intersection point between the constraint set $\mathcal{E}$ and the ellipse $\mathcal{C}$ of smallest surface area. It is a semi-analytical solution: for each material data point z_i^* in $\mathcal{D}$, one can compute the surface area $\mathcal{A}_i = \pi e_i s_i$ of the corresponding ellipse $\mathcal{C}_i$ from (Eq. (A.9b)), determined the smallest one and consequently obtain the corresponding mechanical state from Eqs. (A.6b) and (A.8).

The data-driven approach then yields a model-free solution to the mechanical BVP, directly from material data strain-stress points. We remark that the (non)linearity of the material response sampled by the material data points has no influence on the formulation nor the resolution of the problem.

1.3 Numerical example

We illustrate the data-driven approach with a numerical example of the one-dimensional bar. The material data set is synthetically generated from sampling a nonlinear strain-stress curve $\sigma = E\varepsilon + 100E\varepsilon^3$ according to the parameters given in Table A.1. The other parameters for the problem are given in Table A.2. The so-

Table A.1 – Parameters for sampling the constitutive model.

$\varepsilon_{\min}$ (-)	$\varepsilon_{\max}$ (-)	E (GPa)	number of material data points m (-)
-0,01	0,01	210	101

Table A.2 – Parameters of mechanical problem.

l (cm)	a (cm^2)	u_0 (cm)	k (N.m^1)	C (GPa)
10	3,1416	2,5	4,2E8	210

lution is obtained by computing the minimal distance between the mechanical state and the material data set according to Eq. (A.8). Results are shown in Fig. A.2, where we also plot *a posteriori* a few ellipses $\mathcal{C}_i$, centered on some material data points of $\mathcal{D}$ and on the material state solution, tangent to the constraint set $\mathcal{E}$.

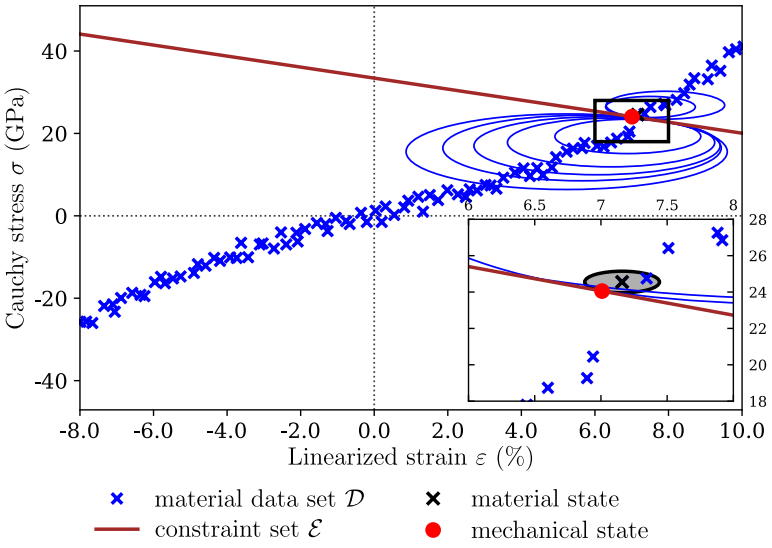


Figure A.2 – Data-driven solution for linearized kinematics: the mechanical state solution is the tangent point between ellipse $\mathcal{C}_i$ of smallest area and the constraint $\mathcal{E}$ (in red).

2 Finite strain solution

Extending the approach to finite strain elasticity requires to redefine the phase space and associated distance function. In this section, we compare two possible formulations with the aid of the one-dimensional bar example presented above. The initial or undeformed length (resp. cross sectional area) of the bar is denoted L (resp. A).

2.1 Lagrangian formulation and resolution

The Lagrangian phase space $\mathcal{Z}^L = \{(E, S)\}$ consisting of Lagrangian strain – second Piola-Kirchhoff stress pairs has been proposed by Nguyen and Keip (2018). As shown next, this choice induces nonlinearities which complicate the resolution.

Let $\mathcal{D}^L = \{(E_i^*, S_i^*) \| i = 1, \cdots, m\}$ denote the Lagrangian material data set and $\mathcal{E}^L$ the Lagrangian constraint set. The minimization problem writes:

$$\min_{z \in \mathcal{Z}^L} \min_{z_i^* \in \mathcal{D}^L} \frac{1}{2} C \left(E - E_i^*\right)^2 + \frac{1}{2C} \left(S - S_i^*\right)^2, \tag{A.12a}$$

$$\text{subject to } z \in \mathcal{E}^L \Leftrightarrow \begin{cases} E = \frac{\lambda^2 - 1}{2} \text{ with } \lambda = \frac{u}{L} + 1 & \text{(compatibility)}, \\ S = \frac{K}{\lambda}(\lambda_0 - \lambda) & \text{(equilibrium)}. \end{cases} \tag{A.12b}$$

with $K = kL/A$ and $\lambda_0 = u_0/L + 1$. The minimization constraints Eq. (A.12b) are nonlinear.

As previously, we first solve a stationary problem with fixed material state $z^* = (E^*, S^*) \in \mathcal{D}$:

$$\delta \left(\frac{1}{2} C \left(\frac{\lambda^2 - 1}{2} - E^* \right)^2 + \frac{1}{2C} \left(S - S^*\right)^2 - \eta \left(S - \frac{K}{\lambda} \left(\lambda_0 - \lambda\right) \right) \right) = 0 \tag{A.13a}$$

$$\delta\lambda \Rightarrow \quad \eta = \frac{\lambda^5 C - \lambda^3 C \left(2E^* + 1\right)}{2\lambda_0 K} \; ; \tag{A.13b}$$

$$\delta S \Rightarrow \quad \frac{1}{C}(S - S^*) - \eta = 0 \; ; \tag{A.13c}$$

$$\delta\eta \Rightarrow \quad S - \frac{K\left(\lambda_0 - \lambda\right)}{\lambda} = 0. \tag{A.13d}$$

Combining Eqs. (A.13b) to (A.13d) yields a 6-order polynomial equation in λ:

$$C^2\lambda^6 - C^2\left(2E^* + 1\right)\lambda^4 + 2\lambda_0 K \left(S^* + K\right)\lambda - 2\lambda_0^2 K^2 = 0 \tag{A.14}$$

It then not possible in that case to find a semi-analytical to the data-driven problem.

The mechanical state solution is determined numerically, by computing the distance Eq. (A.12a) between all material data points in $\mathcal{D}$ and their associated mechanical states given by Eq. (A.14) and Eq. (A.13d).

2.2 Nominal formulation and resolution

Another possible choice for the phase space is the so-called nominal state pairs: $\mathcal{Z}^N = \{(\boldsymbol{F}, \boldsymbol{P})\}$ of deformation gradient tensor – first Piola-Kirchhoff stress tensor, proposed by Conti et al. (2020). In 1D, it is reduced to the longitudinal stretch ratio – longitudinal stress plane $\mathcal{Z}^N = \{(\lambda, P)\}$. Let $\mathcal{D}^N = \{(\lambda_i^*, P_i^*) \| i = 1, \cdots, m\}$ denote the nominal material data set and $\mathcal{E}^N$ denote the nominal constraint set. The data-driven minimization then writes:

$$\min_{z \in \mathcal{Z}^N} \min_{z_i^* \in \mathcal{D}^N} \frac{1}{2} C \left(\lambda - \lambda_i^*\right)^2 + \frac{1}{2C} \left(P - P_i^*\right)^2, \tag{A.15a}$$

$$\text{subject to } z \in \mathcal{E}^N \Leftrightarrow \begin{cases} \lambda = \frac{u}{L} + 1 & \text{(compatibility)}, \\ P = K(\lambda_0 - \lambda) & \text{(equilibrium)}. \end{cases} \tag{A.15b}$$

We remark that here, the minimization constraints Eq. (A.15b) are linear.

Resolution is then equivalent to the small strain case Appendix 1.2. For a fixed material state $z^* = (\lambda^*, P^*)$ the stationary problem

$$\delta \left(\frac{1}{2} C \left(\lambda - \lambda^*\right)^2 + \frac{1}{2C} \left(P - P^*\right)^2 - \eta \left(P - K \left(\lambda_0 - \lambda\right)\right) \right) = 0 \tag{A.16}$$

yields the equation of an ellipse $\mathcal{C}^N(z, z^*)$, centered on $z^* = (\lambda^*, P^*)$, of semi-axes f and p defined as

$$\left(\frac{\lambda - \lambda^*}{f} \right)^2 + \left(\frac{P - P^*}{p} \right)^2 = 1, \tag{A.17a}$$

with

$$f^2 = C^{-1} L_c^2 \text{ and } p^2 = C L_c^2, \tag{A.17b}$$

with

$$L_c^2 = \frac{C}{K^2 + C^2} \left(K \left(\lambda_0 - \lambda^*\right) - P^*\right)^2. \tag{A.17c}$$

The aspect ratio of the ellipse is again fully determined by the parameter C. Let us remark also that

$$2\pi \left(\frac{1}{2} C \left(\lambda - \lambda^*\right)^2 + \frac{1}{2C} \left(P - P^*\right)^2 \right) = \pi f p = \text{aire}(\mathcal{C}^N(z, z^*)), \tag{A.18}$$

then, the semi-analytical data-driven solution to the nominal formulation is obtained from the mechanical state $z = (\lambda, P)$ defined as the intersection point between $\mathcal{E}^N$ and the ellipse $\mathcal{C}^N$ of smallest surface area.

2.3 Numerical example

Both formulations are compared using identical material data sets, generating from sampling the same constitutive model as follows:

- sample m evenly space values of stretch ratios λ between extremal values $\lambda_{\min}$ and $\lambda_{\max}$,
- compute the corresponding nominal stresses P and Lagrangian stresses S from a neo-Hookean model ($P = \mu(\lambda - \lambda^{-2})$ and $S = \lambda P$),
- apply Gaussian noise to the data: $\lambda \leftarrow \lambda + \tilde{\lambda}$,
- compute the corresponding Lagrangian strains $E = \left(\lambda^2 - 1\right)/2$.

The sampling parameters are given in Table A.3. The mechanical parameters are listed in Table A.4.

Table A.3 – Sampling parameters for the constitutive model in large strain.

$\lambda_{\min}$ (-)	$\lambda_{\max}$ (-)	μ (MPa)	number of material data points m (-)
1	3	0,276	100

Table A.4 – Parameters of the mechanical problem in large strain.

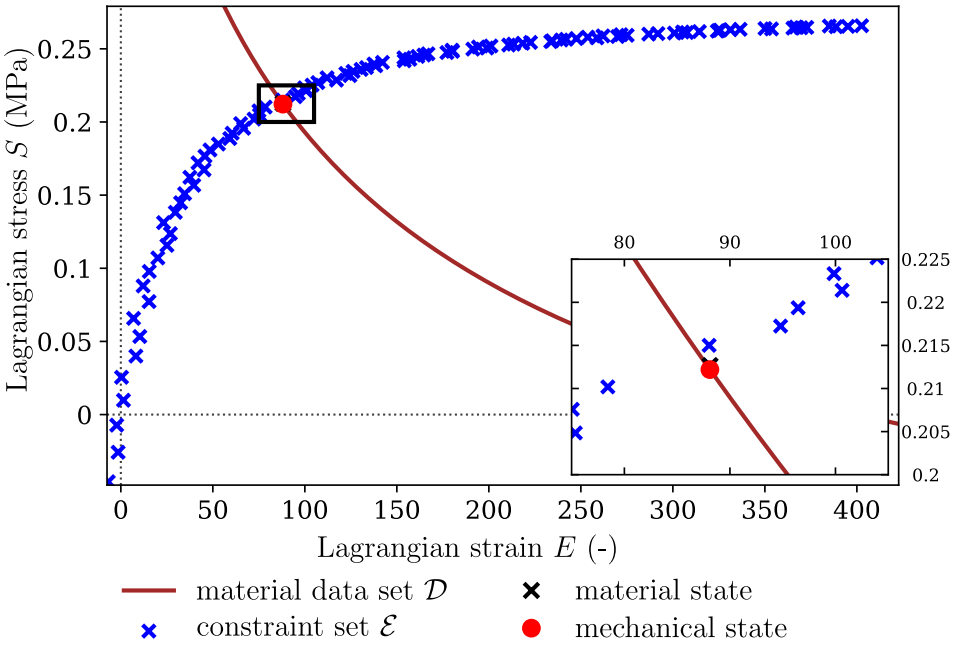
l (cm)	a (cm^2)	k (N.m^1)	u_0 (cm)	C (MPa)
10	3,1416	828	20	0,828

Comparing numerical values of the mechanical state solution in Table A.5 indicates that the two approaches are equivalent. In addition, Fig. A.3 shows the

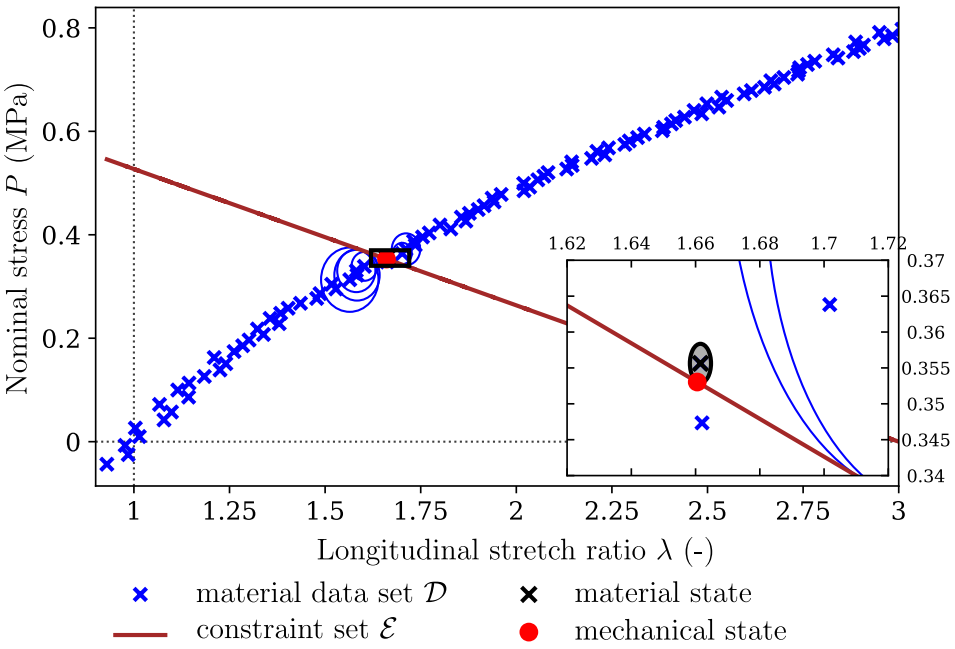
Table A.5 – Mechanical state solution for both Lagrangian and nominal formulation in large strain.

	λ (-)	P (MPa)	E (%)	S (MPa)
Nominal formulation	1,661	0,353	87,9	0,213
Lagrangian formulation	1,662	0,353	88,1	0,212

consistency between the nominal and small strain formulations as the solution again lies at the intersection between the constraint set and the ellipse of smallest area.



(a) Lagrangian phase space (E, S): the mechanical state solution is the point of the constraint set $\mathcal{E}$ which is the closest to the discrete material data set



(b) Nominal phase space (λ, P), the mechanical state solution lies at the intersection between the ellipse $\mathcal{C}_i^N$ of smallest area and the constraint set $\mathcal{E}$.

Figure A.3 – Data-driven solution in large strain.

3 Conclusion

The data-driven approach introduced by Kirchdoerfer and Ortiz (2016) has been extended to large strain with a Lagrangian formulation (Nguyen and Keip, 2018) and a nominal formulation (Conti et al., 2020). The former induces nonlinearities which already complicates the resolution of a very simple one-dimensional problem. The latter yields a constrained minimization with linear constraints, easier to solver. The geometrical interpretations formulated in small strain are then retrieved in large strain for the nominal formulation: the mechanical state solution lies at the intersection of the ellipse of smallest surface area and the linear constraint set of mechanical admissible states. These encouraging results invite to adopt this formulation for more complex problems, in two or three dimensions.

Appendix B

Euclidean mapping

The tree-based search algorithm interface of Pedregosa et al. (2011) requires the data set and the query points to sit in a standard Euclidean space. The local states $z_e \in \mathcal{Z}^e_{\text{loc}}$, which consist of pairs of second-order tensors, must then be recast into a single vector, such that the data-driven distance between two points in local phase space is equivalent to the Euclidean distance between their respective transformed vectors. In the following, we give the corresponding transformation for each of the Lagrangian and nominal formulations. In both cases, the transformation relies on the Mandel notation.

1 Mandel notation

Let consider a second-order tensor $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, in dimension n. We denote $\mathbf{A} \in \mathbb{R}^{n^2}$ the corresponding vector in Mandel notation. For $n = 3$, the components of $\mathbf{A}$ are related to $\boldsymbol{A}$ as follows Brannon (2018)

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{33} & A_{\underline{23}} & A_{\underline{31}} & A_{\underline{12}} & A_{\overline{32}} & A_{\overline{13}} & A_{\overline{21}} \end{bmatrix}^T, \tag{B.1a}$$

with

$$A_{\underline{ij}} = \frac{\sqrt{2}}{2}\left(A_{ij} + A_{ji}\right), \tag{B.1b}$$

$$A_{\overline{ij}} = \frac{\sqrt{2}}{2}\left(A_{ij} - A_{ji}\right). \tag{B.1c}$$

Note that when $\boldsymbol{A}$ is symmetrical, then $A_{\overline{ij}} = 0$, $\forall(i,j)$ and the Mandel vector $\mathbf{A}$ can be reduced to its upper part in $\mathbb{R}^{n(n-1)}$. In particular, when $n = 2$, the Mandel notation of $\boldsymbol{A}_{(2)} \in \mathbb{R}^{2\times 2}$ writes

$$\mathbf{A}_{(2)} = \begin{cases} \begin{bmatrix} A_{11} & A_{22} & A_{\underline{12}} \end{bmatrix}^T & \text{if} \quad \boldsymbol{A}^T = \boldsymbol{A}, \\ \begin{bmatrix} A_{11} & A_{22} & A_{\underline{12}} & A_{\overline{21}} \end{bmatrix}^T & \text{otherwise.} \end{cases} \tag{B.2}$$

Taking into account the major and minor symmetries, the Mandel form of a positive-definite fourth-order tensor $\mathbb{C} \in \mathbb{R}^{2\times2\times2\times2}$ in two dimensions is a positive-definite matrix $\mathbf{C} \in \mathbb{R}^{3\times3}_{\text{sym}}$, defined as

$$\mathbf{C} = \begin{pmatrix} C_{1111} & C_{1122} & \sqrt{2}C_{1112} \\ & C_{2222} & \sqrt{2}C_{2212} \\ \text{sym.} & & 2C_{1212} \end{pmatrix} \tag{B.3}$$

2 Lagrangian formulation

We recall that the data-driven local distance d in the Lagrangian formulation is defined as

$$d^2\left((\boldsymbol{E},\boldsymbol{S}),\,(\boldsymbol{E}',\boldsymbol{S}')\right) = \frac{1}{2}\left(\Delta\boldsymbol{E} : \mathbb{C} : \Delta\boldsymbol{E} + \Delta\boldsymbol{S} : \mathbb{C}^{-1} : \Delta\boldsymbol{S}\right), \tag{B.4}$$

with $\Delta\boldsymbol{E} = \boldsymbol{E} - \boldsymbol{E}'$ and $\Delta\boldsymbol{S} = \boldsymbol{S} - \boldsymbol{S}'$ and $\mathbb{C}$ a positive-definite fourth-order tensor. Using the Mandel form of each tensor, the distance can be expressed as

$$d^2\left((\mathbf{E},\mathbf{S}),\,(\mathbf{E}',\mathbf{S}')\right) = \frac{1}{2}\left(\Delta\mathbf{E}\cdot\mathbf{C}\Delta\mathbf{E} + \Delta\mathbf{S}\cdot\mathbf{C}^{-1}\Delta\mathbf{S}\right) \tag{B.5}$$

with $\Delta\mathbf{E} = \mathbf{E} - \mathbf{E}'$ and likewise for $\Delta\mathbf{S}$, and $\cdot$ the standard inner product in $\mathbb{R}^{n(n-1)}$. Let denote $\mathbf{C}^{1/2}$ the positive-definite matrix such that $\mathbf{C} = \mathbf{C}^{1/2}\mathbf{C}^{1/2}$ and $\mathbf{C}^{-1/2}$ such that $\mathbf{C}^{-1/2} = \mathbf{C}^{-1/2}\mathbf{C}^{-1/2}$. Then, the data-driven distance can be expressed as the standard inner product of a single vector $\mathbf{z} - \mathbf{z}' \in \mathbb{R}^{2n(n-1)}$:

$$d^2\left((\mathbf{E},\mathbf{S}),\,(\mathbf{E}',\mathbf{S}')\right) = \left(\mathbf{z} - \mathbf{z}'\right)\cdot\left(\mathbf{z} - \mathbf{z}'\right) \tag{B.6}$$

where the transformed vector $\mathbf{z}$ (resp. $\mathbf{z}'$) of the local state $z = (\boldsymbol{E},\boldsymbol{S})$ (resp. $z' = (\boldsymbol{E}',\boldsymbol{S}')$) is obtained from the *Lagrangian Euclidean mapping* $\mathcal{M}_{\text{Lag}}$, defined as

$$\mathcal{M}_{\text{Lag}}: \quad \mathbb{R}^{n\times n}_{\text{sym}} \times \mathbb{R}^{n\times n}_{\text{sym}} \to \mathbb{R}^{2n(n-1)} \tag{B.7}$$

$$z = (\boldsymbol{E},\boldsymbol{S}) \mapsto \mathbf{z} = \begin{pmatrix} \frac{1}{\sqrt{2}}\,\mathbf{C}^{1/2}\,\mathbf{E} \\ \frac{1}{\sqrt{2}}\,\mathbf{C}^{-1/2}\,\mathbf{S} \end{pmatrix}. \tag{B.8}$$

3 Nominal formulation

Provided that the C-parameter in the nominal distance is a scalar, the *nominal Euclidean mapping* $\mathcal{M}_{\text{nom}}$ is more easily derived. The deformation gradient and first Piola-Kirchhoff stress tensors are simply recast into their factorized Mandel form by

$$\mathcal{M}_{\text{nom}}: \quad \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n} \to \mathbb{R}^{2n^2} \tag{B.9}$$

$$z = (\boldsymbol{F},\boldsymbol{P}) \mapsto \mathbf{z} = \begin{pmatrix} \frac{1}{\sqrt{2}}\,C^{1/2}\,\mathbf{F} \\ \frac{1}{\sqrt{2}}\,C^{-1/2}\,\mathbf{P} \end{pmatrix}. \tag{B.10}$$

Thus,

$$V(\boldsymbol{F}) + V^*(\boldsymbol{P}) = \mathbf{z}\cdot\mathbf{z}, \tag{B.11}$$

with V and V^* the quadratic functions defined in Eq. (III.67) with $(p,q) = (2,2)$.

Appendix C

neo-Hookean model for incompressible plane stress problems

The standard incompressible neo-Hookean hyperelastic potential $\Psi(\boldsymbol{C})$ is given as

$$\Psi(\boldsymbol{C}) = \frac{1}{2}\mu\left(\operatorname{tr}\boldsymbol{C} - 3\right), \tag{C.1}$$

where $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F} \in \mathbb{R}^{3\times3}_{\text{sym}}$ is the right Cauchy-Green stretch tensor and μ is the shear modulus. The second Piola-Kirchhoff stress tensor $\boldsymbol{S} \in \mathbb{R}^{3\times3}_{\text{sym}}$ is expressed as a function of the unknown hydrostatic pressure p (Bonet et al., 2016):

$$\boldsymbol{S} = \mu III_C^{-1/3}\left(\boldsymbol{I} - \frac{1}{3}I_C\boldsymbol{C}^{-1}\right) + pJ\boldsymbol{C}^{-1}, \tag{C.2}$$

where $I_C = \operatorname{tr}\boldsymbol{C}$ and $III_C = \det\boldsymbol{C}$ are the first and third invariants of $\boldsymbol{C}$ respectively, and $J = \det\boldsymbol{F}$. When the material is incompressible, $J = 1$, and plane stress conditions are assumed, the hydrostatic pressure can be explicitly determined. Eq. (C.2) is then reduced to a two-dimensional constitutive model.

Specifically, if $\boldsymbol{e}_3$ is the plane stress direction, $\boldsymbol{S}\boldsymbol{e}_3 = \boldsymbol{0}$ and the right Cauchy-Green tensor $\boldsymbol{C}$ writes

$$\boldsymbol{C} = \begin{pmatrix} C_{11} & C_{12} & 0 \\ C_{21} & C_{22} & 0 \\ 0 & 0 & C_{33} \end{pmatrix}, \tag{C.3}$$

where, from the incompressibility constraint $J = \sqrt{III_C} = 1$,

$$C_{33} = \left(\det\boldsymbol{C}_{(2)}\right)^{-1} \tag{C.4}$$

with $\boldsymbol{C}_{(2)} \in \mathbb{R}^{2\times2}_{\text{sym}}$ the two-dimensional left upper part of $\boldsymbol{C}$ in Eq. (C.3). Together with the plane stress condition $S_{33} = 0$, Eq. (C.4) enables the pressure p Eq. (C.2) to be explicitly evaluated as

$$p = \frac{1}{3}\mu\left(\operatorname{tr}\boldsymbol{C}_{(2)} - 2\left(\det\boldsymbol{C}_{(2)}\right)^{-1}\right). \tag{C.5}$$

The in-plane components of the second Piola-Kirchhoff stress tensor are then determined directly by the two-dimensional expression of the neo-Hookean model in plane stress and incompressible conditions as

$$\boldsymbol{S}_{(2)} = \mu\left(\boldsymbol{I}_{(2)} - \left(\det\boldsymbol{C}_{(2)}\right)^{-1}\boldsymbol{C}_{(2)}^{-1}\right), \tag{C.6}$$

where the subscript (2) indicates the 2×2 components of a tensor.

Appendix D

Analysis of constitutive models for nominal data-driven finite strain elasticity

---

1 Introduction

1.1 Statements

In Lemma 3.6. and Example 3.7., Conti et al. (2020) provide us with two strain energy density functions which generate (p, q)-coercive data sets, suitable for data-driven finite elasticity computations.

In 2D, the strain energy functions writes:

$$\hat{W}_2(\boldsymbol{F}) = \frac{1}{2}|\boldsymbol{F}|^2 + \frac{1}{4}a|\boldsymbol{F}|^4 + g(\det \boldsymbol{F}) \tag{D.1}$$

with $\boldsymbol{F} \in \mathbb{R}^{2\times 2}$ the 2D deformation gradient tensor and

$$g(\det \boldsymbol{F}) = \frac{1}{2}\beta\left(\det \boldsymbol{F} - 1 - \frac{1+2a}{\beta}\right)^2 \tag{D.2}$$

with $a > 0$ and $\beta \in (0, 2a)$. $|A|$ denotes the Frobenius norm of tensor A:

$$|A| = \sqrt{\mathrm{tr}\,(A^T : A)} \tag{D.3}$$

In 3D, the strain energy density function writes:

$$\hat{W}_3(\boldsymbol{F}) = \frac{1}{2}|\boldsymbol{F}|^2 + \frac{1}{4}a|\boldsymbol{F}|^4 + \frac{1}{6}e|\boldsymbol{F}|^6 + g(\det \boldsymbol{F}) \tag{D.4}$$

with $a \geq 0$, $e > 0$ and $\boldsymbol{F} \in \mathbb{R}^{3\times 3}$ the 3D deformation gradient tensor. For the volumetric function g we use a similar one to the 2D case, which insures that the stress is 0 when no deformation occurs:

$$g(\det \boldsymbol{F}) = \frac{1}{2}\beta\left(\det \boldsymbol{F} - 1 - \frac{1+3a+9e}{\beta}\right)^2. \tag{D.5}$$

1.2 Objective

The objective of this appendix is to compare the proposed Conti-Müller-Ortiz (CMO) model to standard hyperelastic models for compressible materials, in order to interpret the constants as material parameters.

We first derive, analytically, the definition of the second Piola-Kirchhoff stress tensors for the proposed and the standard models, namely compressible neo-Hookean and Yeoh models (Section 2). We then compare the behavior of the different formulations resulting from 3 standards tests (Section 3):

1. simple shear test,
2. uniaxial tensile test,
3. hydrostatic compression test (3D only).

Finally, we show that the CMO model is incompatible with physical considerations in compression (Section 4).

2 Analytical derivation

2.1 Derive stress tensor of CMO model

The strain energy density functions can be re-written in terms of the invariants of the 2D or 3D Cauchy-Green strain tensors:

$$W_2(i_1, i_3) = \frac{1}{2}i_1 + \frac{1}{4}ai_1^2 + \frac{1}{2}\beta\left(\sqrt{i_3} - 1 - \frac{1+2a}{\beta}\right)^2 \tag{D.6}$$

$$W_3(I_1, I_3) = \frac{1}{2}I_1 + \frac{1}{4}aI_1^2 + \frac{1}{6}eI_1^3 + \frac{1}{2}\beta\left(\sqrt{I_3} - 1 - \frac{1+3a+9e}{\beta}\right)^2. \tag{D.7}$$

where (i_1, i_3) (respectively (I_1, I_3)) are the first and third invariants of the 2D (respectively 3D) strain tensor $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F}$. The 2D invariants may indeed differ from the 3D case depending on the dimensionality reduction assumption. In the following, $j = \det \boldsymbol{F}$ (respectively $J = \det \boldsymbol{F}$) denotes the determinant of the 2D (respectively 3D) deformation gradient tensor.

First, we insure that the volumetric function $g(\det \boldsymbol{F})$ verifies the condition for (p, q)-coercivity of the material data set:

> Let $g \in \boldsymbol{C}^1(\mathbb{R})$ be convex and such that, for some $b, d \geq 0$,
>
> $$|g'(t)| \leq b + d|t|, \ \forall t \in \mathbb{R} \tag{D.8}$$

Here we have, for $J > 0$,

$$g(J) = \frac{\beta\left(J - 1 - \frac{3a+9e+1}{\beta}\right)^2}{2} \tag{D.9}$$

hence,

$$g'(J) = J\beta - 3a - \beta - 9e - 1$$

and then

$$|g'(J)| \leq b + \delta|J| \text{ with } (b,\delta) = (3a+\beta+9e+1,\ \beta) > (0,0)$$

Second, the strain energy density function is re-written so that $W = 0$ when no deformation occurs:

$$W = \frac{I_1}{2} + \frac{a\left(I_1^2-9\right)}{4} + \frac{\beta\left(\sqrt{I_3}-1-\frac{3a+9e+1}{\beta}\right)^2}{2} + \frac{e\left(I_1^3-27\right)}{6} - \frac{3}{2} \tag{D.10}$$

Finally, the 2nd Piola-Kirchhoff stress tensor is derived using the standard formula:

$$s = 2\frac{\partial W_2}{\partial i_1}\boldsymbol{i} + 2\frac{\partial W_2}{\partial i_3}\ i_3(\boldsymbol{F}^T\boldsymbol{F})$$

in 2D and,

$$\boldsymbol{S} = 2\frac{\partial W_3}{\partial I_1}\boldsymbol{I} + 2\frac{\partial W_3}{\partial I_3}\ I_3(\boldsymbol{F}^T\boldsymbol{F})$$

in 3D, where $\boldsymbol{i}$ (respectively $\boldsymbol{I}$) is the 2D (respectively 3D) second-order identity tensor. The 2nd Piola-Kirchhoff stress tensor writes

$$\boldsymbol{S}(\boldsymbol{C}) = J\beta\left(J-1-\frac{3a+9e+1}{\beta}\right)\boldsymbol{C}^{-1} + \left(I_1^2 e + I_1 a + 1\right)\boldsymbol{I} \tag{D.11}$$

2.2 Linear elasticity limits

The linear elastic constants can be derived in the limit case where $\boldsymbol{F} \to \boldsymbol{I}$, using the following property:

$$\left.\frac{\partial^2 W}{\partial\lambda_i\lambda_j}\right|_{\lambda_1=1\lambda_2=1\lambda_3=1} = \lambda + 2\mu\delta_{ij},$$

where λ_i is the $\boldsymbol{i}^{\text{th}}$ principal stretch and (λ,μ) the Lamé constants. Here,

$$\begin{aligned} W(\lambda_1,\lambda_2,\lambda_3) = {} & \frac{a\left(\left(\lambda_1^2+\lambda_2^2+\lambda_3^2\right)^2-9\right)}{4} \\ & + \frac{\beta\left(\lambda_1\lambda_2\lambda_3-1-\frac{3a+9e+1}{\beta}\right)^2}{2} + \frac{e\left(\left(\lambda_1^2+\lambda_2^2+\lambda_3^2\right)^3-27\right)}{6} \\ & + \frac{\lambda_1^2}{2} + \frac{\lambda_2^2}{2} + \frac{\lambda_3^2}{2} - \frac{3}{2}. \end{aligned} \tag{D.12}$$

Then,

$$\lambda = -a + \beta + 3e - 1 \tag{D.13a}$$

$$\mu = 3a + 9e + 1 \tag{D.13b}$$

$$E = \frac{(3a+9e+1)(3a+3\beta+27e-1)}{2a+\beta+12e} \tag{D.13c}$$

$$\nu = \frac{-a+\beta+3e-1}{2(2a+\beta+12e)} \tag{D.13d}$$

2.3 Define neoHokean and Yeoh compressible models

The strain energy density function of standard compressible neo-Hookean and Yeoh (Yeoh, 1993) models respectively write

$$W_d^{\mathrm{NH}}(I_1, I_3) = C_{10}(I_1 - d) + h(J) \tag{D.14}$$

$$W_d^{\mathrm{Yeoh}}(I_1, I_3) = C_{10}(I_1 - d) + C_{20}(I_1 - d)^2 + h(J) \tag{D.15}$$

with $h(J)$ a volumetric function. We draw attention to the fact that this formulation is *not* the standard additive split of the strain energy function between an isochoric and a volumetric part.

2.3.1 Volumetric function

Here, we choose a commonly used volumetric function

$$h(J) = -2C_{10}\log(J) + D_1(J-1)^2 \tag{D.16}$$

As will be shown in the next sections, this choice insures that the hydrostatic pressure exhibit a physical behavior at high compression levels:

$$p(J \to 0) \to -\infty \tag{D.17}$$

However, this function does not fully verify the condition for coercivity Eq. (D.8):

$$h'(J) = -\frac{2C_{10}}{J} + 2D_1 J - 2D_1$$

$$|h'(J)| \leq b + \delta|J| \text{ with } (b,\delta) = \left(\frac{2C_{10}}{J} + 2D_1,\ 2D_1\right) > (0,0)$$

Nevertheless, we use these models as comparison for they are standard.

2.3.2 neo-Hookean compressible

The 2nd Piola-Kirchhoff stress tensor is derived as previously:

$$\boldsymbol{S}^{\mathrm{NH}} = 2C_{10}\boldsymbol{I} + \left(-2C_{10} + 2D_1 J(J-1)\right)\boldsymbol{C}^{-1} \tag{D.18}$$

In the limit case, the linear elastic constants are also derived as before:

$$\lambda^{\mathrm{NH}} = 2D_1 \tag{D.19a}$$

$$\mu^{\mathrm{NH}} = 2C_{10} \tag{D.19b}$$

$$E^{\mathrm{NH}} = \frac{2C_{10}(2C_{10} + 3D_1)}{C_{10} + D_1} \tag{D.19c}$$

$$\nu^{\mathrm{NH}} = \frac{D_1}{2(C_{10} + D_1)} \tag{D.19d}$$

2.3.3 Yeoh compressible

Similarly, the 2nd Piola-Kirchhoff stress tensor and associated linear elastic constants for the Yeoh model are:

$$\boldsymbol{S}^{\text{Yeoh}} = \left(-2C_{10} + 2D_1 J\left(J-1\right)\right)\boldsymbol{C}^{-1} + \left(2C_{10} + 4C_{20}\left(I_1 - 3\right)\right)\boldsymbol{I} \tag{D.20}$$

and

$$\lambda^{\text{Yeoh}} = 8C_{20} + 2D_1 \tag{D.21a}$$

$$\mu^{\text{Yeoh}} = 2C_{10} \tag{D.21b}$$

$$E^{\text{Yeoh}} = \frac{2C_{10}\left(2C_{10} + 12C_{20} + 3D_1\right)}{C_{10} + 4C_{20} + D_1} \tag{D.21c}$$

$$\nu^{\text{Yeoh}} = \frac{2C_{20} + \frac{D_1}{2}}{C_{10} + 4C_{20} + D_1} \tag{D.21d}$$

3 Comparison of the models

3.1 Testing values for the parameters

We choose arbitrary values of the parameters for each model:

- the parameters for the CMO model are set to $\alpha = 0.1$, $\beta = 0.2$, $e = 0.3$;
- the parameters of the neo-Hookean model read $C_{10} = 1$, $D_1 = 0.1$;
- the parameters of the Yeoh model read $C_{10} = 1$, $C_{20} = 0.2$, $D_1 = 0.1$.

3.2 Simple shear test

In a simple shear test, the deformation gradient writes:

$$\boldsymbol{F} = \begin{bmatrix} 1 & \gamma & (0) \\ 0 & 1 & (0) \\ (0) & (0) & (1) \end{bmatrix}_{(\vec{e}_1,\vec{e}_2,(\vec{e}_3))} \tag{D.22}$$

where $\gamma = \tan\alpha$, with α the shear angle.

3.2.1 Conti-Müller-Ortiz model (CMO):

The shear stress writes

$$\tau = \gamma\left(a\left(\gamma^2+3\right) + e\left(\gamma^2+3\right)^2 + 1\right) \tag{D.23}$$

which yields the following Taylor expansion:

$$\tau = \gamma\left(3a + 9e + 1\right) + O\left(\gamma^2\right) \tag{D.24}$$

3.2.2 neo-Hookean model

The shear stress writes

$$\tau^{\mathrm{NH}} = 2C_{10}\gamma \tag{D.25}$$

which yields the following Taylor expansion:

$$\tau^{\mathrm{NH}} = 2C_{10}\gamma + O\left(\gamma^2\right) \tag{D.26}$$

3.2.3 Yeoh model

The shear stress writes

$$\tau^{\mathrm{Yeoh}} = 2\gamma\left(C_{10} + 2C_{20}\gamma^2\right) \tag{D.27}$$

which yields the following Taylor expansion:

$$\tau^{\mathrm{Yeoh}} = 2C_{10}\gamma + O\left(\gamma^2\right) \tag{D.28}$$

3.2.4 Strain-stress curves

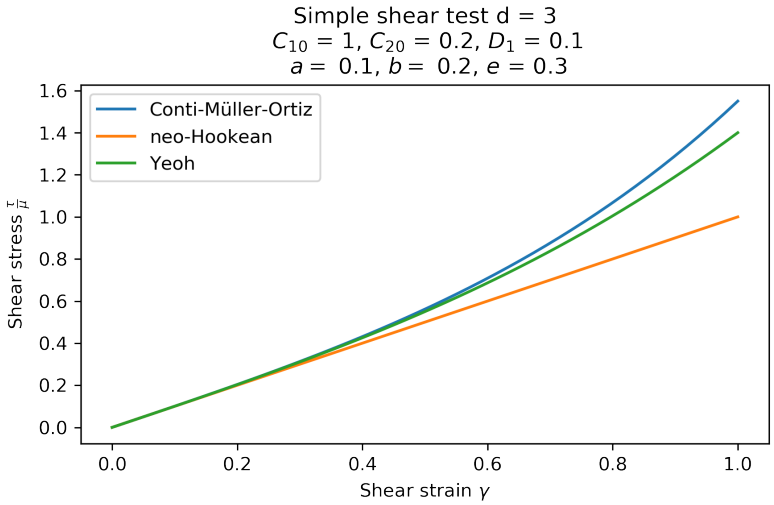


Figure D.1 – Comparison of the consistutive models on a shear test.

3.3 Uniaxial tensile test

In a uniaxial tensile test, the deformation gradient writes:

$$\boldsymbol{F} = \begin{bmatrix} \lambda_1 & 0 & (0) \\ 0 & \lambda_2 & (0) \\ (0) & (0) & (\lambda_2) \end{bmatrix}_{(\vec{e}_1,\vec{e}_2,(\vec{e}_3))} \tag{D.29}$$

The first Piola-Kirchhoff stress tensor) is first derived with the usual push-forward operation $P = \boldsymbol{FS}$. To obtain the nominal tensile stress P_{11} as a function of λ_1, we proceed as follows: we first solve equation $P_{22} = 0$ to obtain λ_2 as a function of λ_1. Reinjecting $\lambda_2(\lambda_1)$ into $P_{11}(\lambda_1, \lambda_2)$ we then obtain $P_{11}(\lambda_1)$.

In practice, the analytical solution for the equation $\lambda_2(\lambda_1)$ is too complicated in general. We then solve it for each point of the stretch-stress curve, numerically.

3.3.1 Conti-Muller-Ortiz model

The stress reads

$$P_{11}(\lambda_1, \lambda_2) = \lambda_1 \left(a\left(\lambda_1^2 + 2\lambda_2^2\right) + \frac{\beta\lambda_2^2\left(\lambda_1\lambda_2^2 - 1 - \frac{3a+9e+1}{\beta}\right)}{\lambda_1} + e\left(\lambda_1^2 + 2\lambda_2^2\right)^2 + 1 \right) \tag{D.30}$$

and λ_2 is obtained from $P_{22} = 0$:

$$\lambda_2 \left(a\left(\lambda_1^2 + 2\lambda_2^2\right) + \beta\lambda_1\left(\lambda_1\lambda_2^2 - 1 - \frac{3a+9e+1}{\beta}\right) + e\left(\lambda_1^2 + 2\lambda_2^2\right)^2 + 1 \right) = 0 \tag{D.31}$$

3.3.2 neo-Hookean model

The stress reads

$$P_{11}^{\mathrm{NH}} = \lambda_2 \left(2C_{10} + \lambda_1^2\lambda_2^2 \left(-\frac{2C_{10}}{\lambda_1^2\lambda_2^4} + \frac{2D_1\left(\lambda_1\lambda_2^2 - 1\right)}{\lambda_1\lambda_2^2} \right) \right) \tag{D.32}$$

and λ_2 is obtained from $P_{22} = 0$:

$$\lambda_2 \left(2C_{10} + \lambda_1^2\lambda_2^2 \left(-\frac{2C_{10}}{\lambda_1^2\lambda_2^4} + \frac{2D_1\left(\lambda_1\lambda_2^2 - 1\right)}{\lambda_1\lambda_2^2} \right) \right) = 0 \tag{D.33}$$

Here, the solutions can be obtained analytically:

$$\text{solutions} = \left[\frac{1}{2\lambda_2^2} - \frac{\sqrt{-4C_{10}\lambda_2^2 + 4C_{10} + D_1}}{2\sqrt{D_1}\lambda_2^2},\ \frac{1}{2\lambda_2^2} + \frac{\sqrt{-4C_{10}\lambda_2^2 + 4C_{10} + D_1}}{2\sqrt{D_1}\lambda_2^2} \right] \tag{D.34}$$

3.3.3 Yeoh model

The stress reads

$$P_{11}^{\mathrm{Yeoh}} = \lambda_1 \left(2C_{10} + 2C_{20}\left(2\lambda_1^2 + 4\lambda_2^2 - 6\right) + \lambda_2^4 \left(-\frac{2C_{10}}{\lambda_1^2\lambda_2^4} + \frac{2D_1\left(\lambda_1\lambda_2^2 - 1\right)}{\lambda_1\lambda_2^2} \right) \right) \tag{D.35}$$

and λ_2 is obtained from $P_{22} = 0$:

$$\lambda_2 \left(2C_{10} + 2C_{20} \left(2\lambda_1^2 + 4\lambda_2^2 - 6 \right) + \lambda_1^2 \lambda_2^2 \left(-\frac{2C_{10}}{\lambda_1^2 \lambda_2^4} + \frac{2D_1 \left(\lambda_1 \lambda_2^2 - 1 \right)}{\lambda_1 \lambda_2^2} \right) \right) = 0 \tag{D.36}$$

Here, the solutions can also be obtained analytically:

$$\text{solutions} = \left[\frac{D_1 \lambda_2 - \sqrt{\Delta}}{4C_{20}\lambda_2 + 2D_1 \lambda_2^3}, \frac{D_1 \lambda_2 + \sqrt{\Delta}}{4C_{20}\lambda_2 + 2D_1 \lambda_2^3} \right] \tag{D.37}$$

where

$$\begin{aligned} \Delta^2 = 8C_{10}C_{20} - 16C_{20}D_1\lambda_2^6 + \lambda_2^4 \left(-4C_{10}D_1 - 32C_{20}^2 + 24C_{20}D_1 \right) \\ + \lambda_2^2 \left(-8C_{10}C_{20} + 4C_{10}D_1 + 48C_{20}^2 + D_1^2 \right) \end{aligned} \tag{D.38}$$

3.3.4 Response curves

Top-left Fig. D.2 represents the stress-stretch curves (P_{11}, λ_1) of the different models. It shows hardening in extension phase ($\lambda_1 > 1$) and a convex behavior in compression phase ($\lambda_1 < 1$). The (λ_1, λ_2) curves in the bottom-left Fig. D.2 shows that this behavior in compression in non-physical: the contraction stretch λ_2 is inferior to 1 when $\lambda_1 < 1$, meaning that the more compressed in one direction, the more contracted in the other. The (ε, P_{11}/E) curves in the top-right Fig. D.2 show that the derivation of the Young's modulus in the two first sections was correct since all three models exhibit linear elastic behavior in small strains. Finally, the bottom-right Fig. D.2 shows the (J, p) curves of the different models: the unaxial tension test does not yield enough compression (J is still "large") to exhibit much differences between the CMO model and the others.

Hence, in the next section, we evaluate the models in a pure dilatation/compression test in 3D.

3.4 Pure dilatation / compression test in 3D

In a pure dilatation or compression test, the deformation gradient writes:

$$\boldsymbol{F} = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}_{(\vec{e}_1, \vec{e}_2, \vec{e}_3)} . \tag{D.39}$$

with $\lambda > 0$. The hydrostatic pressure is defined from the Cauchy stress tensor $\sigma = \frac{1}{J} \boldsymbol{F}\ \boldsymbol{S}\ \boldsymbol{F}^T$ as :

$$p = \frac{1}{3} \text{tr}\ \sigma \tag{D.40}$$

For each model, we compute the hydrostatic pressure as a function of the Jacobian and evaluate its limits. We also calculate the bulk modulus in the limit case $J \to 1$.

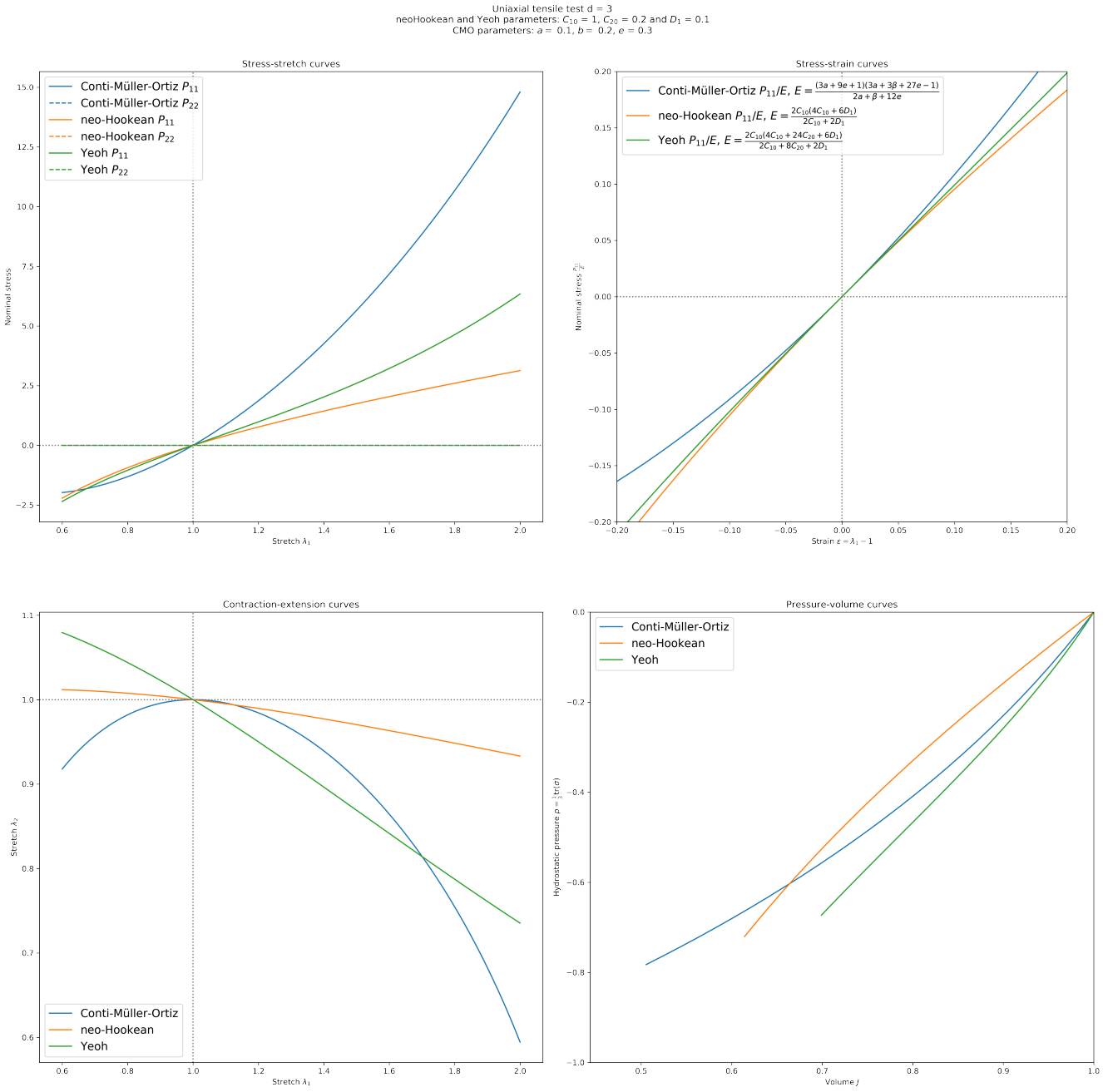


Figure D.2 – Comparison of the constitutive models on a uniaxial tensile test.

3.4.1 Conti-Müller-Ortiz model

$$p = \frac{3a\lambda^2 + \beta\lambda\left(\lambda^3 - 1 - \frac{3a+9e+1}{\beta}\right) + 9e\lambda^4 + 1}{\lambda} \tag{D.41}$$

The limits of $p(J)$ written as

$$p(J) = 3\sqrt[3]{J}a + J\beta + 9Je - 3a - \beta - 9e - 1 + \frac{1}{\sqrt[3]{J}} \tag{D.42}$$

read

$$p(J=1) = 0,\ p(J\to 0) \to \infty,\ \text{and}\ p(J\to\infty) \to \infty \tag{D.43}$$

As mentionned before, in a physical behavior, the hydrostatic pressure must tend to $-\infty$ when the continuum degenerates to a single point ($J \to 0$) (Doll and Schweizerhof, 2000). Here, it is not the case. This anomaly is discussed further in the next section.

The bulk modulus can be obtained in the limit case where $J \to 1$:

$$\kappa^{\mathrm{CMO}} = \lim_{J\to 1} p(J) = a + \beta + 9e - \frac{1}{3} \tag{D.44}$$

3.4.2 neo-Hookean model

$$p^{\mathrm{NH}}(J) = 2D_1 J - 2D_1 + \frac{-2C_{10}\sqrt[3]{J} + 2C_{10}J}{J^{\frac{4}{3}}} \tag{D.45}$$

and

$$p^{\mathrm{NH}}(J=1) = 0,\ p^{\mathrm{NH}}(J\to 0) \to -\infty,\ \text{and}\ p^{\mathrm{NH}}(J\to\infty) \to \infty \tag{D.46}$$

The bulk modulus writes

$$\kappa^{\mathrm{NH}} = \frac{4C_{10}}{3} + 2D_1 \tag{D.47}$$

3.4.3 Yeoh model

$$p^{\mathrm{Yeoh}}(J) = 12C_{20}\sqrt[3]{J} + 2D_1 J - 2D_1 + \frac{-2C_{10}\sqrt[3]{J} + 2C_{10}J - 12C_{20}J}{J^{\frac{4}{3}}} \tag{D.48}$$

$$p^{\mathrm{Yeoh}}(J=1) = 0,\ p^{\mathrm{Yeoh}}(J\to 0) \to -\infty,\ \text{and}\ p^{\mathrm{Yeoh}}(J\to\infty) \to \infty \tag{D.49}$$

The bulk modulus reads

$$\kappa^{\mathrm{Yeoh}} = \frac{4C_{10}}{3} + 8C_{20} + 2D_1 \tag{D.50}$$

3.4.4 Volume – pressure curves

The normalized compression curves given in Fig. D.3 show that:

- the bulk moduli were correctly evaluated using $\lim_{J\to 1} p(J)$;
- the behavior of the CMO model differs from the two other standard models for relatively low compression levels ($J \simeq 0.8$);
- the hydrostatic pressure (volumetric stress) tends to positive infinity for high compression levels.

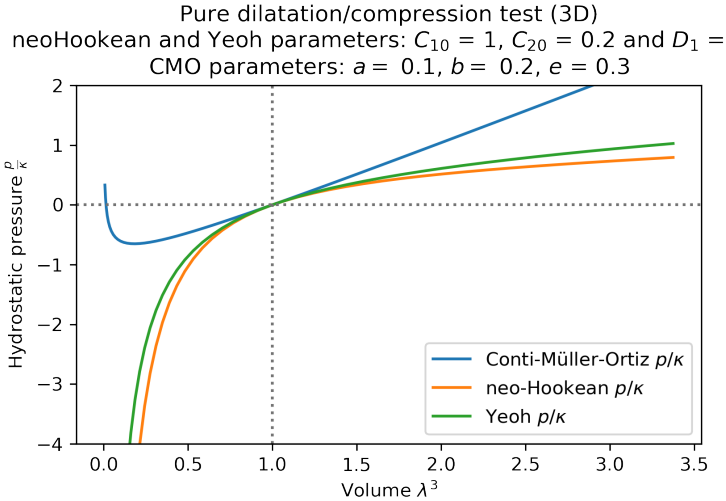


Figure D.3 – Comparison of the constitutive models on a compression test.

4 Relationship between the volumetric function and pressure

In this appendix, we compared the CMO models proposed in Conti et al. (2020) with standard hyperlastic models for compressible materials. In both 2D (not shown) and 3D cases, the CMO model exhibit a non-physical response in compression.
Proposition. The condition (3.24) in (Conti et al., 2020)

$$|g'(t)| \leq b + c|t|, \forall t \in \mathbb{R} \tag{D.51}$$

is incompatible with the physical condition on the hydrostatic pressure

$$\lim_{J\to 0} p = -\infty. \tag{D.52}$$

In this section, we attempt to prove the above proposition in the most general case. Let first consider a generic strain energy density function of the form:

$$W(I_1, J) = f(I_1) + g(J) \tag{D.53}$$

Then, the hydrostatic pressure can be derived as a function of f and g:

$$p(I_1, J) = \frac{2I_1 \frac{d}{dI_1} f(I_1)}{3J} + \frac{d}{dJ} g(J) \tag{D.54}$$

Now, if the condition Eq. (D.51) is satisfied, we have:

$$-(b + cJ) \leq g'(J) \leq b + cJ, \forall J \in \mathbb{R}^+$$

with $b > 0$ and $c > 0$. Then

$$p^-(I_1, J) \leq p(I_1, J) \leq p^+(I_1, J), \forall J \in \mathbb{R}^+ \tag{D.55}$$

where

$$p^-(I_1, J) = \frac{2I_1 f'(I_1)}{3J} - (b + cJ) \tag{D.56}$$

$$p^+(I_1, J) = \frac{2I_1 f'(I_1)}{3J} + (b + cJ) \tag{D.57}$$

It should be noted that I_1 and J are not independent. Hence, determining the limits of p^+ or p^- when J tends to 0 is not straightforward as they strongly depend on the evolution of the f function. Let then evaluate these bounds when f is a polynomial function, such as the one proposed in Conti et al. (2020), with:

$$f_{\text{poly}}(I_1) = \sum_{k=1}^{n} I_1^k \alpha_k, \text{ with } \alpha_k > 0, \forall k \tag{D.58}$$

Then

$$p^+_{\text{poly}}(I_1, J) = \frac{2I_1 \sum_{k=1}^{n} I_1^{k-1} k\alpha_k}{3J} + Jc + b \tag{D.59a}$$

$$p^-_{\text{poly}}(I_1, J) = \frac{2I_1 \sum_{k=1}^{n} I_1^{k-1} k\alpha_k}{3J} - Jc - b \tag{D.59b}$$

To simplify further, let then consider the pure dilatation/compression case where $I_1 = 3\lambda^2$ and $J = \lambda^3$. Then

$$I_1 = 3J^{\frac{2}{3}} \tag{D.60}$$

and

$$p^+_{\text{poly}}(J) = Jc + b + \frac{2\sum_{k=1}^{n} k \left(3J^{\frac{2}{3}}\right)^{k-1} \alpha_k}{\sqrt[3]{J}} \tag{D.61a}$$

$$p^-_{\text{poly}}(J) = -Jc - b + \frac{2\sum_{k=1}^{n} k \left(3J^{\frac{2}{3}}\right)^{k-1} \alpha_k}{\sqrt[3]{J}} \tag{D.61b}$$

Hence, the limit of p^+ (and equivalently p^-) when the volume tends to 0 is driven by the constant term in the summation. Indeed,

$$\frac{\left(J^{\frac{2}{3}}\right)^{k-1}}{\sqrt[3]{J}} \tag{D.62}$$

always tends to 0 except when $\frac{2k}{3} - 1 < 0$, *i.e.* when $k \leq 1$. Then, from the squeeze theorem, if $\alpha_1 > 0$, then

$$\lim_{J \to 0} p = +\infty, \tag{D.63}$$

which completes the proof of the proposition, valid for the CMO model for which $f(I_1) = \sum_{k=1}^{n} \alpha_k I_1^k$.

Following a similar rationale, the limit can also be derived when f takes the form of the polynomial functions that are used for the neo-Hookean and Yeoh models presented here:

$$f_{\text{poly}} = \sum_{k=1}^{n} (I_1 - 3)^k. \tag{D.64}$$

The limit is again driven by the constant term in the summation which now writes:

$$\bar{\alpha} = \sum_{k=1}^{n} 2k\alpha_k (-3)^{k-1}. \tag{D.65}$$

Then, the limit depends on the materials parameters and *can* be negative infinity:

$$\lim_{J \to 0} p = \text{sign}(\bar{\alpha}) \times \infty, \tag{D.66}$$

which contradicts the above proposition.

5 Conclusion

To conclude, the coercivity condition and the form of the strain energy density function proposed in Conti et al. (2020) both imply positive infinite pressure when the continuum tends to a single point, which is non-physical. Possibles ways to avoid this issue could be:

- either to relax the growth condition on the volumetric function g if the coercivity can still hold by other means;
- or to reformulate the strain energy function with the suitable polynomial form:

$$W_d(I_1, J) = \sum_{k=1}^{n} \alpha_k (I_1 - d)^k + g(J) \tag{D.67}$$

such that

$$\sum_{k=1}^{n} 2k\alpha_k (-3)^{k-1} < 0 \tag{D.68}$$

Annexe E

Résumé étendu en français

Résumé *Cette dernière annexe répond à la demande de l'École Doctorale d'un « résumé substantiel en français » lorsque le manuscrit est rédigé en anglais. Pour qu'il soit utile au lecteur ou à la lectrice, nous avons choisi de suivre* stricto sensus *le plan de la thèse. Pour alléger le manuscrit, la grande majorité des figures ne sont pas reproduites ici.*

Contents

1 Chapitre 1 : État de l'art

Dans ce premier chapitre, nous faisons la relation entre les méthodes numériques pour la mécanique non linéaire et la science des données. Dans un premier temps, nous rappelons les principales équations de la mécanique non linéaire des solides et les méthodes numériques associées. Dans un deuxième temps, nous montrons que la modélisation du comportement mécanique des matériaux est un pan de la recherche dans lequel il est pertinent de faire appel à la science des données.

1.1 Rappels de mécanique numérique non linéaire

Cette première section se veut un condensé du livre de référence de Bonet and Wood (2008). Elle permet de présenter le cadre théorique nécessaire à l'étude des grandes transformations élastiques, qui sont l'objet d'une grande partie de ce travail de thèse. On rappelle à la fois les équations de mécanique des milieux continus et la méthode des éléments finis.

1.1.1 Mécanique non linéaire des milieux continus

Dans le cadre de la mécanique non linéaire des solides, deux sources de non-linéarité existent : (i) la non-linéarité dite géométrique qui provient notamment des grandes transformations, (ii) la non-linéarité dite matériau qui provient de la non-linéarité de la relation contrainte-déformation. De la première source découle notamment la définition de deux configurations distinctes : la configuration de référence ou non déformée, et la configuration courante ou déformée. La première est appelée matérielle ou Lagrangienne tandis que la seconde est appelée spatiale ou Eulérienne. Chaque quantité décrivant la déformation d'un corps peut alors se rapporter soit aux coordonnées de la configuration de référence, notées $\boldsymbol{X}$, soit aux coordonnées de la configuration déformée, notées $\boldsymbol{x}$. L'ensemble de cette partie est structurée autour de cette distinction : chaque type d'équation (décrivant le mouvement ou les efforts) est classée selon la configuration à laquelle elle se rapporte. Nous ajoutons aux formulations Lagrangienne et Eulérienne une formulation intermédiaire, dite nominale, qui comprend les quantités de transport entre les deux configurations. L'ensemble des équations nécessaires est fourni dans une série de tableaux récapitulatifs.

On rappelle d'abord les équations de la cinématique et notamment les relations déplacements-déformation, qui sont désignées par la suite comme les équations de compatibilité. En effet, en grandes transformations, le tenseur des petites déformations $\boldsymbol{\varepsilon} = 1/2\left(\boldsymbol{\nabla u} + \boldsymbol{\nabla}^T \boldsymbol{u}\right)$ n'est plus suffisant pour décrire la déformation du corps, où $\boldsymbol{u} = \boldsymbol{x} - \boldsymbol{X}$ est le champ de déplacement. La transformation est alors mesurée par le tenseur gradient de la transformation $\boldsymbol{F} = \boldsymbol{\nabla}_0 \boldsymbol{u} + \boldsymbol{I}$ dont découle les définitions du tenseur des déformations de Green-Lagrange $\boldsymbol{E} = 1/2\left(\boldsymbol{F}^T\boldsymbol{F} - \boldsymbol{I}\right)$ (matériel) et du tenseur des déformations d'Euler-Almansi $\boldsymbol{e} = 1/2\left(\boldsymbol{I} - (\boldsymbol{F}\boldsymbol{F}^T)^{-1}\right)$ (spatial).

Les différentes mesures de contrainte sont ensuite introduites. Le tenseur (spatial) des contraintes de Cauchy $\boldsymbol{\sigma}$ est défini comme une force par unité de surface

déformée tandis que les tenseurs de Piola-Kirchhoff 1 ($\boldsymbol{P}$) et 2 ($\boldsymbol{S}$) sont définis au moyen du principe des puissances virtuelles, comme les mesures de contrainte conjuguées respectivement à $\dot{\boldsymbol{F}}$ et $\dot{\boldsymbol{E}}$, les dérivées temporelles de $\boldsymbol{F}$ et $\boldsymbol{E}$. Les équations d'équilibre dans leurs formes forte (locale) et faible (intégrale) sont rappelées.

Après un rapide bilan des équations et des inconnues, il apparaît qu'une relation est manquante : le modèle de comportement qui relie les déformations et les contraintes. Nous rappelons alors le cadre théorique de développement des modèles de comportement avant de préciser le cas particulier de l'hyperélasticité. La contrainte dérive d'un potentiel représentant l'énergie élastique stockée durant la déformation du matériau. Dans le cas isotrope, on peut notamment écrire les contraintes $\boldsymbol{\sigma}$ et $\boldsymbol{S}$ comme des fonctions des invariants des tenseurs $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F}$ et $\boldsymbol{b} = \boldsymbol{F}\boldsymbol{F}^T$. On définit également le module tangent au comportement élastique du matériau qui provient de la dérivée directionnelle des relations de comportement.

1.1.2 Discrétisation éléments finis

La méthode des éléments finis, appliquée à la mécanique non linéaire, est rappelée dans cette section.

La géométrie du corps est approchée par un maillage éléments finis composé de nœuds reliés par des éléments isoparamétriques, qui constituent le support compact des fonctions d'interpolation. On insiste dans un premier temps sur l'approximation de la géométrie (des positions des points matériels), du déplacement et de la vitesse virtuelle. En injectant ces approximations dans le principe des puissances virtuelles on obtient un système d'équations non linéaires à résoudre : le résidu mesurant l'écart entre les efforts intérieurs (énergie de déformation) et les efforts extérieurs (forces appliquées) doit être nul. En raison des non-linéarités géométrique et matériau, ces équations sont non linéaires.

Classiquement, ce système algébrique est résolu au moyen d'une méthode de Newton-Raphson dont nous détaillons les différents ingrédients. Il s'agit d'une méthode de résolution itérative qui consiste à résoudre une série de systèmes linéaires, dont la matrice de raideur est chaque fois recalculée. Dans le cadre de l'hyperélasticité, cette matrice de raideur contient deux termes : un premier terme constitutif qui contient la tangente au modèle de comportement, et un second terme géométrique qui provient de la relation non linéaire des déformations aux déplacements. Les expressions détaillées de ces différents termes sont rappelées pour chacune des formulations matérielle, nominale et Lagrangienne.

1.1.3 Bilan

La mécanique numérique non linéaire a été présentée dans le cadre de la méthode des éléments finis appliquée aux grandes transformations élastiques. Dans ce cas, l'équilibre est appliqué dans sa forme faible, le principe des puissances virtuelles, dont la discrétisation aboutit à un système d'équations non linéaires. Nous avons rappelée la méthode de résolution de Newton-Raphson habituellement utilisée.

Cette présentation nous a permis de bien distinguer les différentes formulations Lagrangienne, nominale et Eulérienne du problème, ce qui se révélera important dans la suite du travail. Les formulations matérielle et spatiale présentent l'avantage de ne faire intervenir que des tenseurs symétriques tandis que la formulation nominale offre une relation de compatibilité linéaire.

1.1.4 La modélisation du comportement matériau : un défi central

Les équations de la mécanique non linéaire des milieux continus sont établies depuis longtemps ; de même la méthode des éléments finis est aujourd'hui considérée mature. La difficulté centrale de la mécanique numérique non linéaire des matériaux est donc la modélisation du comportement matériau. Il s'agit de développer, implémenter et identifier des modèles de comportement toujours plus complexes.

- Les modèles sont généralement développés de façon phénoménologique consistant à « ajuster une relation mathématiques à des données expérimentales » sans « relier les mécanismes de déformation à la physique de la structure microscopique sous-jacente du matériau » (Holzapfel, 2000). Des cadres théoriques comme l'hyperélasticité ou l'approche thermodynamique pour les matériaux standards généralisés permettent d'élaborer de tels modèles.

- Le cadre théorique étant fixé, le choix du modèle de comportement à utiliser pour un matériau particulier reste ouvert. Par ailleurs la complexité du modèle choisi implique de développer des méthodes numériques spécifiques et rendent ainsi l'implémentation parfois complexe.

- L'identification des paramètres du modèle reste néanmoins le point le plus difficile, le problème étant généralement mal-posé (solution non unique). Par ailleurs, les données expérimentales utilisées sont bien souvent réduites à une courbe de traction uniaxiale, à partir de laquelle le comportement triaxial est extrapolé par la forme tensorielle de la relation de comportement, sans réelle garantie de validité.

- Les points ci-dessus sont d'autant plus prégnants pour la modélisation du comportement des matériaux hétérogènes. Leur réponse mécanique dépend de manière complexe des différentes propriétés des constituants mais aussi de leur arrangement dans la microstructure. Des modèles dits à champs moyens permettent de prédire la réponse de certaines microstructures simplifiées. La plus récente avancée est l'utilisation des méthodes numériques pour prédire, grâce à de nombreux calculs microstructuraux, la réponse effective du matériau.

En définitive, une bonne partie de la recherche actuelle en mécanique des matériaux est dédiée à l'élaboration de modèles de comportement. La complexité croissante de ces modèles requiert un nombre important et une richesse de données expérimentales pour identifier au mieux les paramètres. C'est pourquoi cet aspect de la mécanique peut être considéré comme un point d'entrée pour la science des données dans le domaine de la mécanique numérique.

1.2 La science des données dans la mécanique numérique

1.2.1 Les données en mécanique numérique

En amont, les données sont utilisées pour identifier les paramètres du modèle de comportement utilisé dans le calcul éléments finis. Elles peuvent être massives et riches grâce aux progrès expérimentaux et notamment aux mesures de champs. Ensuite en aval, les simulations numériques effectuées sur des super-calculateurs génèrent un très grand nombre de données ; c'est notamment le cas pour les calculs sur des matériaux hétérogènes. Ces données microstructurales peuvent ensuite être utilisées pour ajuster un modèle macroscopique. Une analyse éléments finis en mécanique se trouve donc aujourd'hui à la fois en amont et en aval d'une chaîne de production de données matériau.

1.2.2 Ce qu'est la science des données et où elle peut intervenir en mécanique numérique

Nous reprenons dans cette partie la définition de la science des données (*data science*) proposée par Dhar (2013). Il s'agit notamment d'extraire de la connaissance de données non structurées et de répondre à la question « quel modèle satisfont ces données ? » plutôt que « quelle données satisfont ce modèle ? ». Pour appliquer cette approche à la mécanique numérique, nous revenons rapidement sur la structure d'un problème aux conditions aux limites en mécanique. Les équations de ce problème peuvent se diviser en trois groupes : (i) la cinématique et les conditions aux limites de Dirichlet, qui proviennent de la géométrie, (ii) l'équilibre des efforts et les conditions aux limites de Neumann, qui proviennent de lois de conservation, et (iii) les modèles de comportement, qui traduisent la réponse mécanique du matériau. Tandis que les deux premiers types d'équations sont universellement valides (elles reposent sur des principes physiques fondamentaux), il existe davantage de sources d'incertitudes dans le troisième groupe. Mais surtout, c'est bien dans ce dernier groupe que le besoin de données est le plus important. La science des données au sens de Dhar (2013) nous paraît donc particulièrement adaptée à aux modèles de comportement : il s'agit bien d'élaborer, à partir de connaissances extraites de données non structurées, des relations mathématiques prédictives du comportement du matériau.

1.2.3 Exemples d'utilisation de la science des données en modélisation et simulation

Les exemples sont très nombreux dans le champ de la micromécanique, qui s'intéresse à la modélisation et à la simulation numérique des matériaux hétérogènes. On cite notamment l'utilisation de diverses techniques comme la réduction de modèle, les réseaux de neurones ou les méthodes de réduction de la dimensionalité qui sont utilisées pour développer des métamodèles du comportement homogénéisé de la microstructure. Il existe également des méthodes d'apprentissage de variété ou d'approximation par des polynômes pour les matériaux homogènes.

Néanmoins, dans tous ces exemples, il s'agit toujours d'élaborer un modèle de comportement et non d'utiliser les données brutes, qui sont d'ailleurs souvent mises de côté après le processus d'apprentissage ou d'identification du modèle. En fait, il s'agit d'une perte de données massives et riches : le modèle de comportement agit comme un entonnoir entre les données amont et les données aval. C'est pourquoi Kirchdoerfer and Ortiz (2016) ont proposé un nouveau paradigme de la mécanique numérique, appelé *data-driven computational mechanics* (DDCM) qui propose de reformuler le problème mécanique pour utiliser directement les données, sans passer par un modèle de comportement. Leur travail constitue la base de nos travaux de recherche. Nous le détaillons donc dans le chapitre suivant.

2 Chapitre 2 : *data-driven computational mechanics*

Dans ce chapitre, nous présentons la méthode DDCM telle qu'introduite par Kirchdoerfer and Ortiz (2016) et complétée par Conti et al. (2018). Nous présentons ensuite les recherches qui se sont inspirées de cette approche avant de proposer nos pistes de travail pour la thèse.

2.1 Présentation de la méthode

2.1.1 Idée générale

L'approche DDCM de Kirchdoerfer and Ortiz (2016) a été proposée comme un « nouveau paradigme » qui permet de se passer de la « modélisation empirique du comportement du matériau ». À la place d'une relation mathématique, la réponse mécanique du matériau est représentée par une base de données de couples tensoriels déformation-contrainte. L'approche repose sur la séparation précédemment évoquée des équations de la mécanique en deux groupes : la caractérisation de la réponse matériau d'une part, et la satisfaction de principes physiques fondamentaux d'autre part. Tandis que le modèle de comportement ne fait qu'*approcher* la réponse du matériau dans certaines conditions, les équations fondamentales de la mécanique (équilibre et compatibilité) doivent toujours être satisfaites. Le problème mécanique est reformulé en ce sens comme un problème de minimisation de la distance entre deux types de champs d'état (*state fields*) : l'*état matériau* représente la réponse matériau du corps déformable et prend ses valeurs dans la base de données, l'*état mécanique* vérifie les équations d'équilibre et de compatibilité. Le premier appartient à l'ensemble données matériau (*material data set*) et le deuxième à l'ensemble contrainte (*constraint set*), qui sont deux sous-espaces de l'espace des phases (*phase space*) défini par l'ensemble des champs d'état possibles (c.-à-d. des couples déformation-contrainte). Le solveur *data-driven* consiste alors à assigner à chaque point matériel du corps un état déformation-contrainte qui (i) vérifie les contraintes de compatibilité et d'équilibre du problème dans l'espace physique, et (ii) est le plus proche possible, dans l'espace des phases, d'une base de données matériau préétablie.

L'approche a d'abord été introduite par Kirchdoerfer and Ortiz (2016) comme une façon d'utiliser les données brutes directement dans les simulations de structure. Depuis, Conti et al. (2018) ont apporté une formulation en mécanique des milieux continus du problème *data-driven*. Une formulation variationnelle a également été proposée dans Nguyen et al. (2020).

Dans cette section, nous présentons d'abord la formulation en mécanique des milieux continus de Conti et al. (2018) avant de rappeler la méthode et l'ensemble des résultats numériques de Kirchdoerfer and Ortiz (2016).

2.1.2 Formulations continue et discrète

Dans la formulation continue, on travaille avec des champs d'état relatifs à des espaces fonctionnels distincts. Les états mécaniques définis ci-dessus appartiennent

à l'ensemble contrainte, noté $\mathcal{E}$, qui contient l'ensemble des champs déformation-contrainte vérifiant la compatibilité et l'équilibre. Considérant un corps $\Omega \in \mathbb{R}^3$ en équilibre quasi-statique, soumis à des déplacements imposés $\boldsymbol{u}_D$ et des forces volumiques $\boldsymbol{f}$ et surfaciques $\boldsymbol{t}$, les équations de la cinématique et d'équilibre s'écrivent :

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(\boldsymbol{\nabla u} + \boldsymbol{\nabla}^T \boldsymbol{u}\right) \qquad \text{dans } \Omega, \tag{E.1a}$$

$$\boldsymbol{u} = \boldsymbol{u}_D \qquad \text{sur } \Gamma_D, \tag{E.1b}$$

et

$$\operatorname{div} \boldsymbol{\sigma} + \boldsymbol{f} = \boldsymbol{0} \qquad \text{dans } \Omega, \tag{E.2a}$$

$$\boldsymbol{\sigma n} = \boldsymbol{t} \qquad \text{sur } \Gamma_N, \tag{E.2b}$$

avec $\boldsymbol{n}$ la normale à la surface Γ_N tel que $\Gamma_D \cup \Gamma_N = \partial\Omega$ et $\Gamma_N \cap \Gamma_D = \emptyset$. Ainsi, l'ensemble contrainte s'écrit :

$$\mathcal{E} = \{(\boldsymbol{\varepsilon}, \boldsymbol{\sigma}) \mid \text{Eqs. (E.1) et (E.2)}\}. \tag{E.3}$$

Les états matériau prennent leur valeur dans des bases de données

$$\mathcal{D}_{\text{loc}} = \{(\boldsymbol{\varepsilon}_i, \boldsymbol{\sigma}_i) \mid i \in [1 \,..\, m]\}; \tag{E.4}$$

ils appartiennent alors à l'ensemble données matériau, noté $\mathcal{D}$ et défini par

$$\mathcal{D} = \{(\boldsymbol{\varepsilon}, \boldsymbol{\sigma}) \mid (\boldsymbol{\varepsilon}(\boldsymbol{x}), \boldsymbol{\sigma}(\boldsymbol{x})) \in \mathcal{D}_{\text{loc}} \text{ pour tout } \boldsymbol{x} \text{ dans } \Omega\}. \tag{E.5}$$

La solution du problème classique se situe à l'intersection de ces deux sous-espaces : elle doit à la fois vérifier les équations de la mécanique et le modèle de comportement. Sans loi de comportement, la représentation de la réponse matériau est seulement discrète ; cette intersection est donc probablement vide. Il convient donc de relaxer le problème : la solution est déterminée comme le champ déformation-contrainte qui vérifie les équations de la mécanique et s'approche au plus près de l'ensemble données matériau. Le problème s'écrit alors comme une double minimisation :

$$\mathcal{S} = \arg \min_{(\boldsymbol{\varepsilon},\boldsymbol{\sigma}) \in \mathcal{E}} \min_{(\boldsymbol{\varepsilon}',\boldsymbol{\sigma}') \in \mathcal{D}} \| (\boldsymbol{\varepsilon}, \boldsymbol{\sigma}) - (\boldsymbol{\varepsilon}', \boldsymbol{\sigma}') \|_{\mathbb{C}}, \tag{E.6}$$

où la norme $\| \bullet \|_{\mathbb{C}}$ est définie par

$$\| (\boldsymbol{\varepsilon}, \boldsymbol{\sigma}) \|_{\mathbb{C}}^2 = \int_\Omega \left(\frac{1}{2} (\mathbb{C} : \boldsymbol{\varepsilon}) : \boldsymbol{\varepsilon} + \frac{1}{2} (\mathbb{C}^{-1} : \boldsymbol{\sigma}) : \boldsymbol{\sigma} \right) \mathrm{d}v, \tag{E.7}$$

avec $\mathbb{C}$ un tenseur du quatrième ordre défini positif.

Dans Conti et al. (2018), les auteurs prouvent que le problème ci-dessus est bien posé et contient la solution classique linéaire élastique lorsque la base de données est en fait le graphe de la réponse linéaire élastique. Ils démontrent également que lorsque la base de données discrète converge (au sens de la densité de points et

de leur écart au graphe) vers le graphe de la réponse linéaire, la solution *data-driven* converge également vers la solution classique. Il en va de même lorsque la réponse du matériau est non linéaire (mais toujours en petites déformations). Ainsi, la formulation *data-driven* inclut la formulation classique du problème aux limites en élasticité, sous l'hypothèse des petites perturbations.

La formulation discrète a d'abord été proposée dans Kirchdoerfer and Ortiz (2016) pour les treillis. L'état déformation-contrainte d'un point matériel est alors réduit au couple de scalaires $(\varepsilon_e, \sigma_e)$, représentant la déformation et la contrainte longitudinales (constantes) dans une barre e du treillis. Considérant un treillis de M barres et N degrés de liberté, soumis à des efforts $\{f_i\}_{i=1}^N$, le problème *data-driven* s'écrit comme la minimisation suivante :

Trouver $\mathcal{S} = \{(\varepsilon_e, \sigma_e)\}_{e=1}^M$ tel que

$$\mathcal{S} = \arg\min_{(\varepsilon,\sigma)} \sum_{e=1}^{M} w_e \min_{(\varepsilon'_e,\sigma'_e)\in\mathcal{D}_{\text{loc}}^{(e)}} \left(\frac{1}{2} C_e \left(\varepsilon_e - \varepsilon'_e\right)^2 + \frac{1}{2} C_e^{-1} \left(\sigma_e - \sigma'_e\right) \right),$$

sous contraintes

$$\varepsilon_e = \sum_{i=1}^{N} B_{ei} u_i, \ \forall e\,; \quad \sum_{e=1}^{M} w_e B_{ei} \sigma_e - f_i = 0, \ \forall i,$$

avec $\{u_i\}_{i=1}^N$ les degrés de liberté, w_e le poids d'une barre, B_{ei} représentant la géométrie et la connectivité du treillis, C_e une constante jouant le même rôle que $\mathbb{C}$ et $\mathcal{D}_{\text{loc}}^{(e)} = \{(\varepsilon_i, \sigma_i) \mid i \in [1\,..\,m_e]\}$ la base de données représentant la réponse du matériau de la barre e par m_e couples déformation-contrainte.

L'approche est ensuite naturellement étendue en trois dimensions par une formulation éléments finis, appliquée aux solides déformables élastiques. Considérant un maillage de N nœuds et M points d'intégration, l'état d'un point matériel $e \in [1\,..\,M]$ est caractérisé par le couple tensoriel $(\boldsymbol{\varepsilon}_e, \boldsymbol{\sigma}_e)$. La structure est soumise à des forces nodales $\{\boldsymbol{f}_a\}_{a=1}^N$ et les déplacements nodaux sont notés $\{\boldsymbol{u}_a\}_{a=1}^N$. La base de données est maintenant constituée de couple tensoriels : $\mathcal{D}_{\text{loc}}^{(e)} = \{(\boldsymbol{\varepsilon}_i, \boldsymbol{\sigma}_i) \mid i \in [1\,..\,m_e]\}$. Le problème de minimisation s'écrit alors :

Trouver $\mathcal{S} = \{(\boldsymbol{\varepsilon}_e, \boldsymbol{\sigma}_e)\}_{e=1}^M$ tel que

$$\mathcal{S} = \arg\min_{(\boldsymbol{\varepsilon},\boldsymbol{\sigma})} \sum_{e=1}^{M} w_e \min_{(\boldsymbol{\varepsilon}'_e,\boldsymbol{\sigma}'_e)\in\mathcal{D}_{\text{loc}}^{(e)}} \left(W\left(\boldsymbol{\varepsilon}_e - \boldsymbol{\varepsilon}'_e\right) + W^*\left(\boldsymbol{\sigma}_e - \boldsymbol{\sigma}'_e\right) \right)$$

sous contraintes

$$\boldsymbol{\varepsilon}_e = \sum_{a=1}^{N} \boldsymbol{B}_{ea} \boldsymbol{u}_a, \ \forall e\,; \quad \sum_{e=1}^{M} w_e \boldsymbol{B}_{ea}^T \boldsymbol{\sigma}_e - \boldsymbol{f}_a = 0, \ \forall a$$

avec $W(\boldsymbol{\varepsilon}) = 1/2(\mathbb{C} : \boldsymbol{\varepsilon}) : \boldsymbol{\varepsilon}$ et $W^{-1}(\boldsymbol{\sigma}) = 1/2(\mathbb{C}^{-1} : \boldsymbol{\sigma}) : \boldsymbol{\sigma}$.

2.1.3 Algorithme original

La double minimisation ci-dessus est complexe car elle couple une minimisation sous contraintes avec une optimisation combinatoire : les états mécaniques sont à valeurs continues tandis que les états matériau prennent des valeurs discrètes. Une heuristique qui consiste à résoudre alternativement chacune des optimisations est proposée dans Kirchdoerfer and Ortiz (2016).

- Les états matériau $(\boldsymbol{\varepsilon}_e^*, \boldsymbol{\sigma}_e^*)$ sont déterminés par une recherche de plus proche voisin dans l'espace des phases :

$$(\boldsymbol{\varepsilon}_e^*, \boldsymbol{\sigma}_e^*) = \arg \min_{(\boldsymbol{\varepsilon}'_e, \boldsymbol{\sigma}'_e) \in \mathcal{D}_{\text{loc}}^{(e)}} \left(W\left(\boldsymbol{\varepsilon}_e - \boldsymbol{\varepsilon}'_e\right) + W^*\left(\boldsymbol{\sigma}_e - \boldsymbol{\sigma}'_e\right)\right). \tag{E.8}$$

- Les états mécaniques sont alors obtenus par la minimisation sous contraintes ci-dessus, où la minimisation locale sur $\mathcal{D}_{\text{loc}}^{(e)}$ disparaît au profit d'états matériau fixés $(\boldsymbol{\varepsilon}'_e, \boldsymbol{\sigma}'_e) = (\boldsymbol{\varepsilon}_e^*, \boldsymbol{\sigma}_e^*)$. Pour résoudre cette dernière, Kirchdoerfer and Ortiz (2016) emploient une méthode de multiplicateurs de Lagrange, qui fait apparaître deux systèmes pseudo-élastiques linéaires de raideur $\mathbb{C}$, faciles à résoudre.

2.1.4 Résultats numériques

La première étude se concentre sur un treillis de barres d'un même matériau, dont la représentation est un nuage de points échantillonnant une courbe (ε, σ) non linéaire. Une base de données sans bruit et une contenant un bruit Gaussien sont étudiées. Dans les deux cas, les auteurs démontrent la convergence de la solution *data-driven* vers la solution classique du treillis lorsque le nombre de points dans la base de données augmente. La convergence est linéaire et deux fois plus rapide dans le cas non-bruité. Ils démontrent également que la distance finale entre les états mécaniques et matériau (mesurée par la norme $\|\bullet\|_{\mathbb{C}}$) donne une information sur la qualité de la base de données : lorsque la distance entre deux points est grande, cela signifie que, dans la région de l'espace des phases déformation-contrainte de l'état mécanique concerné, la densité de points dans la base de données est insuffisante.

Les résultats de convergence sont corroborés par un exemple tridimensionnel. Il s'agit d'une éprouvette de traction, sous l'hypothèse des contraintes planes. La base de données est générée à partir d'un modèle de comportement linéaire élastique par la suite mis de côté dans la simulation *data-driven*. L'espace des contraintes planes est échantillonné par une grille régulière et les déformations correspondantes sont calculées par une loi de Hooke. La fonction objective décroît comme attendu avec le nombre d'itérations de la minimisation alternée ainsi qu'avec le nombre de points dans la base de données.

2.2 Perfectionnements, inspirations et extensions

La méthode *data-driven* développée par Kirchdoerfer and Ortiz (2016) et Conti et al. (2018) est donc bien établie pour les petites déformations réversibles. Cette

approche novatrice a généré un intérêt renouvelé pour recherche dans le domaine de la mécanique numérique pilotée par les données, comme en témoignent les récentes contributions, qui visent soit à améliorer le solveur, soit à s'en inspirer pour développer des méthodes inverses d'acquisition de données ou bien qui étendent l'approche à d'autres classes de problèmes mécaniques.

2.2.1 S'affranchir des minima locaux et du bruit

Une difficulté centrale de l'approche proposée est qu'elle utilise des données *brutes*. Elle peut donc être sensible au bruit. Par ailleurs, la méthode de minimisation alternée ne permet pas de trouver le minimum global du problème comme cela a été démontré par Kanno (2018). Des publications récentes ont donc cherché à améliorer le solveur pour le rendre davantage robuste au bruit et aux points aberrants :

- Kirchdoerfer and Ortiz (2017) ont tout d'abord proposé un solveur maximisant l'entropie accompagné d'un recuit simulé qui permet de pénaliser les points aberrants dans la base de données matériau.
- S'attaquant au « fléau de la dimensionalité » (la dimension de l'espace des phases est de 12, pour un couple de tenseurs symétriques), Eggersmann et al. (2020) ont adapté une méthode d'apprentissage (*tenseur voting*) pour calculer un plan tangent à chacun des points dans la base de données et minimiser la distance à ce plan plutôt qu'au point lui-même.
- Kanno (2019) a démontré que le problème est un problème de *mixed integer quadratic programming* bien posé, ce qui permet d'accéder au minimum global grâce à des solveurs efficaces mais très gourmands.
- Une autre approche consiste à approximer le nuage de points déformation-contrainte par une variété afin de régulariser le problème (Gebhardt et al., 2020; He and Chen, 2020; Kanno, 2020). Il s'agit alors de coupler une approche *sans modèle* et des techniques d'apprentissage de variété.
- Enfin, on peut mentionner les couplages avec des approches stochastiques de Ayensa-Jiménez et al. (2018); Korzeniowski and Weinberg (2019).

2.2.2 L'identification par les données

Le cadre de la micromécanique est particulièrement propice à la génération de données, comme indiqué plus haut. Ainsi, des bases de données pour la DDCM peuvent être créées par une série de simulations éléments finis sur des microstructures.

Le défi principal de l'approche *data-driven* réside encore dans la capacité à générer des bases de données tensorielles *expérimentales*. Si le tenseur des déformations peut être accessible grâce aux progrès des mesures de champs et de la corrélation d'images numériques, ce n'est pas le cas du tenseur des contraintes. Souvent on n'en connaît qu'une seule composante lors d'essais expérimentaux aux hypothèses fortes. C'est dans cette optique que Leygue et al. (2018) ont proposé une méthode inverse

à la DDCM, baptisée *data-driven identification* (DDI). Cette fois, la base de données est une *sortie* et non une entrée du solveur *data-driven*. À partir des déformations mécaniques, obtenues par dérivation des champs de déplacement résultant de la corrélation d'images numériques, et des efforts appliqués à l'échantillon, la DDI permet d'obtenir (i) les contraintes mécaniques équilibrées, et (ii) la base de données des états matériau, qui échantillonnent la réponse matériau.

Stainier et al. (2019) ont démontré que cette approche pouvait fournir des base de données pertinentes pour la DDCM, correspondant à un échantillonnage préférentiel de la réponse matériau. D'abord développée avec des données synthétiques, la méthode a été appliquée avec succès aux données expérimentales dans la thèse de Dalémat (2019), ce qui ouvre la voie pour « mesurer des champs de contrainte sans loi de comportement » (Dalémat et al., 2019). On peut également citer les travaux de Réthoré et al. (2018) dont la méthode inverse s'applique en élasto-plasticité et ceux de Leygue et al. (2019) pour l'élasticité non linéaire, la plasticité et la dynamique.

2.2.3 Extension à d'autre classes de problèmes

Enfin, la méthode a été étendue à différents types de problèmes, comme la dynamique (Kirchdoerfer and Ortiz, 2018), la diffusion (Nguyen et al., 2020) ou encore la mécanique de la rupture (Carrara et al., 2020).

En réalité, changer de classe de problème signifie change d'espace des phases : il faut déterminer quelles sont les grandeurs intrinsèques suffisantes pour décrire l'état mécanique d'un point matériel. Dans le cadre visco-élastique par exemple, il convient de prendre en compte l'histoire de déformation du matériau dans la base de données (Eggersmann et al., 2019). L'extension aux grandes transformations élastiques a été explorée par Nguyen and Keip (2018) et Conti et al. (2020).

2.3 Bilan

L'approche DDCM ne fournit pas seulement un nouveau solveur éléments finis pour les problèmes élastiques linéaires, peu à même de se comparer à la technologie mature des éléments finis, mais bien davantage un nouveau paradigme pour la mécanique numérique qui a inspiré de nombreuses contributions ces dernières années. Nous pensons que ce travail invite à un changement radical de perspective : la loi de comportement n'est plus écrite « en dur » dans le code mais on s'autorise à s'éloigner légèrement de la réponse matériau, représentée par une base de données brutes. Au contraire, les principes physiques que sont la compatibilité et l'équilibre des efforts doivent toujours être respectés. De ce point de départ, plusieurs pistes de travail se sont ouvertes pour la présente thèse. Nous avons choisies d'explorer deux thèmes, qui font naturellement l'objet de deux parties distinctes dans le manuscrit :

1. l'extension aux grandes transformations élastiques,

2. l'application aux simulations multiéchelles.

Nous nous intéressons dans un premier temps à l'extension aux grandes transformations élastiques, pour laquelle deux formulations sont étudiées :

- Une formulation dite Lagrangienne, dans laquelle l'espace des phases est celui des couples $(\boldsymbol{E}, \boldsymbol{S})$ de déformation de Green-Lagrange – second tenseur des contraintes de Piola-Kirchhoff (Nguyen and Keip, 2018). Cette publication présente une formulation éléments finis restreinte aux cas bidimensionnels et ne peut ainsi pas être comparée aux standards présentés dans le Chapitre 1. Par ailleurs, il n'existe pas de preuve mathématique de convergence similaire au cas des petites déformations, car les contraintes de minimisation (la compatibilité et l'équilibre) sont non linéaires.

- La formulation dit nominale de Conti et al. (2020) propose de résoudre ce dernier point : l'espace des phases est défini par les champs $(\boldsymbol{F}, \boldsymbol{P})$ du tenseur gradient de la transformation – premier tenseur des contraintes de Piola-Kirchhoff. Cependant, seule une formulation pour la mécanique des milieux continus est proposée ; l'implémentation dans un solveur éléments finis reste donc entièrement à faire.

Nos travaux visent à explorer et comparer les deux formulations. Les solveurs éléments finis développés pour les deux approches sont présentés dans le Chapitre 3. Le Chapitre 4 est consacré à la génération et à l'analyse de base de données nécessaires aux simulations DDCM en grandes transformations. Enfin, le Chapitre 5 évalue les approches développées dans les deux précédents chapitres à l'aide d'un problème tridimensionnel complexe. En dernier lieu, le Chapitre 6 constitue une ouverture vers la simulation multiéchelle de structure à matériaux hétérogènes. La base de données est constituée des couples déformation-contrainte macroscopiques et est enrichie par des calculs d'homogénéisation de la réponse microstructurale bien choisis, au cours d'un processus itératif.

3 Chapitre 3 : Solveurs *data-driven* pour les grandes transformations élastiques

3.1 Introduction

L'approche DDCM consiste à minimiser la distance entre deux types de champs déformation-contrainte : l'un représente la réponse du matériau et prend ses valeurs dans une base de données, l'autre décrit la réponse mécanique de la structure et vérifie les principes de compatibilité et d'équilibre. Comme mentionné plus haut, étendre l'approche aux grandes transformations élastiques suppose de choisir les nouvelles variables d'état susceptibles de représenter les réponses matériau et mécanique.

Dans le cadre des matériaux capable de subir de grandes déformations réversibles, dont le comportement est indépendant de l'histoire de déformation, le couple déformation-contrainte suffit. En mécanique non linéaire, le Chapitre 1 nous a montré que plusieurs couples de déformation-contrainte conjugués existent : (i) le couple dit Lagrangien $(\boldsymbol{E}, \boldsymbol{S})$, (ii) le couple dit nominal $(\boldsymbol{F}, \boldsymbol{P})$, (iii) le couple dit Eulérien $(\boldsymbol{d}, \boldsymbol{\sigma})$.

Dans ce chapitre nous explorons les deux premières formulations, qui ont été respectivement proposées par Nguyen and Keip (2018) et Conti et al. (2020). Nous proposons une implémentation éléments finis pour chacune d'elles et discutons de la robustesse des solveurs au moyen d'exemples tests.

3.2 Mécanique numérique en grandes transformations pilotée par les données

La stratégie de résolution adoptée en grandes transformations suit l'heuristique présentée en petites déformations : une minimisation alternée sur les états mécaniques puis matériau. Si la structure reste inchangée, plusieurs ingrédients de la DDCM doivent néanmoins être reformulés :

- L'espace des phases correspond désormais soit à l'ensemble des couples de tenseurs symétriques $(\boldsymbol{E}, \boldsymbol{S})$, soit à l'ensemble des couples de tenseurs $(\boldsymbol{F}, \boldsymbol{P})$;
- La mesure de distance entre états mécaniques et matériau s'en trouve modifiée ;
- La cinématique et l'équilibre des efforts sont reformulés d'après les équations de la mécanique non linéaire rappelées au Chapitre 1 ;
- La non-linéarité des contraintes de minimisation mentionnées ci-dessus demande de déployer des stratégies de résolution spécifiques à chaque formulation.

Cette section est donc organisée comme suit : pour chaque formulation, nous présentons d'abord la formulation continu du problème *data-driven* avant d'en détailler sa discrétisation par la méthode des éléments finis. Nous développons ensuite les stratégies de résolution numérique spécifiques à chaque formulation.

3.2.1 Approche Lagrangienne

Pour Nguyen and Keip (2018), le couple déformation-contrainte Lagrangien $(\boldsymbol{E}, \boldsymbol{S})$ est particulièrement avantageux car la symétrie des tenseurs assure de fait l'objectivité et car la linéarisation vers les petites déformations est directe. Nous ajoutons que la symétrie des tenseurs assure de fait l'équilibre des moments comme en petites déformations ($\boldsymbol{\sigma} = \boldsymbol{\sigma}^T$) et limite la dimensionalité de l'espace des phases à $6+6 = 12$ au plus.

Formulation continue Nous suivons ici l'approche en petites déformations qui repose sur la définition d'espaces fonctionnels mécanique $\mathcal{E}$ et matériau $\mathcal{D}$, sous-ensembles de l'espace des phases des champs déformation-contrainte noté $\mathcal{Z}$. Le problème *data-driven* en grandes transformations est exprimé de la même manière que le problème petites déformations. Considérant un corps occupant un domaine Ω_0 dans sa configuration de référence et un domaine $\Omega \in \mathbb{R}^3$ en équilibre quasi-statique, soumis à des déplacements imposés $\boldsymbol{u}_D$ et des forces volumiques $\boldsymbol{f}$ et surfaciques $\boldsymbol{t}$, les équations de la cinématique s'écrivent en formulation Lagrangienne comme suit :

$$\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{\nabla}^T\boldsymbol{u}\boldsymbol{\nabla}\boldsymbol{u} + \boldsymbol{\nabla}\boldsymbol{u} + \boldsymbol{\nabla}^T\boldsymbol{u}\right) \qquad \text{dans } \Omega, \tag{E.9a}$$

$$\boldsymbol{u} = \boldsymbol{u}_D \qquad \text{sur } \Gamma_{D_0}. \tag{E.9b}$$

L'équilibre est imposé sous sa forme faible (principe des puissances virtuelles) :

$$\delta W = \int_{\Omega_0} \boldsymbol{S} : \delta\dot{\boldsymbol{E}}\,\mathrm{d}V - \int_{\Omega_0} \boldsymbol{f}_0 \cdot \delta\boldsymbol{v}\,\mathrm{d}V - \int_{\partial\Omega_0} \boldsymbol{t}_0 \cdot \delta\boldsymbol{v}\,\mathrm{d}A = 0, \tag{E.10}$$

où $\delta\boldsymbol{v}$ est une vitesse virtuelle cinématiquement admissible depuis la position actuelle du corps, $\delta\dot{\boldsymbol{E}} = 1/2\left(\delta\dot{\boldsymbol{F}}^T\boldsymbol{F} + \boldsymbol{F}^T\delta\dot{\boldsymbol{F}}\right)$ avec $\delta\dot{\boldsymbol{F}} = \partial(\delta\boldsymbol{v})/\partial\boldsymbol{X}$, et $\boldsymbol{f}_0 = J\boldsymbol{f}$ et $\boldsymbol{t}_0 = (\,\mathrm{d}a/\,\mathrm{d}A)\,\boldsymbol{t}$ sont les forces volumiques et surfaciques respectivement exprimées dans la configuration de référence ($J = \det\boldsymbol{F}$). Ainsi, l'ensemble contrainte s'écrit désormais :

$$\mathcal{E} = \{(\boldsymbol{E}, \boldsymbol{S}) \mid \text{Eqs. (E.9) et (E.10)}\}\,. \tag{E.11}$$

Les états matériau prennent leur valeur dans des bases de données de couples déformation-contrainte Lagrangiens

$$\mathcal{D}_{\text{loc}} = \{(\boldsymbol{E}_i, \boldsymbol{S}_i) \mid i \in [1\,..\,m]\} \tag{E.12}$$

et l'ensemble données matériau est défini sur la configuration de référence Ω_0 :

$$\mathcal{D} = \{(\boldsymbol{E}, \boldsymbol{S}) \mid (\boldsymbol{E}(\boldsymbol{x}), \boldsymbol{S}(\boldsymbol{x})) \in \mathcal{D}_{\text{loc}} \text{ pour tout } \boldsymbol{X} \text{ dans } \Omega_0\}\,. \tag{E.13}$$

Comme précédemment, la solution est déterminée comme le champ déformation-contrainte qui vérifie les équations de la mécanique et s'approche au plus près de

l'ensemble données matériau. Le problème s'écrit donc à nouveau comme une double minimisation :

$$\mathcal{S} = \arg \min_{(\boldsymbol{E},\boldsymbol{S})\in\mathcal{E}} \min_{(\boldsymbol{E}',\boldsymbol{S}')\in\mathcal{D}} \| (\boldsymbol{E},\boldsymbol{S}) - (\boldsymbol{E}',\boldsymbol{S}') \|_{\mathbb{C}}, \tag{E.14}$$

où la norme $\| \bullet \|_{\mathbb{C}}$ sur $\mathcal{Z}$ est maintenant définie par

$$\| (\boldsymbol{E},\boldsymbol{S}) \|_{\mathbb{C}}^2 = \int_{\Omega_0} \left(\frac{1}{2} (\mathbb{C} : \boldsymbol{E}) : \boldsymbol{E} + \frac{1}{2} (\mathbb{C}^{-1} : \boldsymbol{S}) : \boldsymbol{S} \right) \mathrm{d}V, \tag{E.15}$$

avec $\mathbb{C}$ un tenseur du quatrième ordre défini positif.

La formulation ci-dessus est également valable en trois ou deux dimensions, lorsque, par exemple, les hypothèses de déformations planes, ou de contraintes planes et d'incompressibilité du matériau sont faites :

- En déformations planes, le tenseur des déformations $\boldsymbol{E}$ est complètement déterminé par ses composantes dans le plan ($E_{33} = 0$). Pour le tenseur des contraintes $\boldsymbol{S}$, seule la composante S_{33} est non nulle hors plan. Cette valeur ne peut pas être déterminée *a posteriori* comme lorsque le modèle de comportement est connu. L'approche *data-driven* donne alors seulement accès à la solution en deux dimensions.

- En contraintes planes, le tenseur des contraintes $\boldsymbol{S}$ est complètement déterminé par ses composantes dans le plan ($S_{33} = 0$). La déformation hors-plan E_{33} peut être obtenue *a posteriori* lorsque le matériau est supposé incompressible.

Dans la suite, les tenseurs $\boldsymbol{E}$ et $\boldsymbol{S}$ sont donc indifféremment bi- ou tridimensionnels.

Formulation éléments finis Pour passer de la formulation continue à la formulation éléments finis il convient de définir une approximation discrète du corps et de la cinématique ainsi qu'une manière d'évaluer les contraintes : les déformations et les contraintes n'étant plus reliées par une loi de comportement, l'interpolation des contraintes ne découle plus naturellement de l'interpolation des déplacements dans le maillage.

Considérons un maillage éléments finis composés de N nœuds connectés par des éléments isoparamétriques de dimension $n \in \{2, 3\}$. Les éléments sont les supports compacts des fonctions d'interpolation nodales, notées $\{\mathcal{N}_a(\boldsymbol{X})\}_{a=1}^N$. Le champ de déplacement dans la structure est alors approché par

$$\boldsymbol{u}(\boldsymbol{X}) = \sum_{a=1}^{N} \boldsymbol{u}_a \mathcal{N}_a(\boldsymbol{X}), \ \forall \boldsymbol{X} \in \Omega_0. \tag{E.16}$$

On note $\alpha \in [1 \,..\, N]$ les indices des degrés de liberté ; ils sont reliés aux nœuds $a \in [1 \,..\, N]$ et à la direction $i \in [1 \,..\, n]$ par une fonction $\alpha(a, i)$[1].

[1] Par exemple $\alpha(a, i) = (a - 1)n + i$.

Par ailleurs, les éléments sont équipés d'une règle d'intégration, typiquement une quadrature de Gauss, de telle sorte que l'intégrale d'une fonctionnelle $\boldsymbol{\mathcal{F}}$ sur le domaine Ω_0 est évaluée par une somme pondérée des valeurs de la fonction aux points d'intégration, indexés par la lettre e :

$$\int_{\Omega_0} \boldsymbol{\mathcal{F}}(\boldsymbol{X})\,\mathrm{d}V = \sum_{e=1}^{M} w_e\,\boldsymbol{\mathcal{F}}(\boldsymbol{X}_e), \tag{E.17}$$

avec M le nombre total de point d'intégration dans le maillage et w_e le poids associé à chaque point d'intégration de coordonnées $\boldsymbol{X}_e$ dans Ω_0.

Enfin, les conditions de Dirichlet sont encodées dans un tableau de n_D déplacements imposés scalaires $\left\{u_\alpha^\mathrm{D}\right\}_{\alpha\in D}$, avec $D = \{\alpha_1, \cdots, \alpha_{n_D}\}$, tel que $\alpha_k \in [1\,..\,Nn]$, $\forall k \in [1\,..\,n_D]$.

D'après les équations cinématiques continues Eqs. (E.9) et la règle d'interpolation Eq. (E.16), les équations discrètes de la cinématique s'écrivent :

$$\boldsymbol{E}(\boldsymbol{X}_e) = \boldsymbol{E}_e = \frac{1}{2}\left(\boldsymbol{\nabla}_0^e\boldsymbol{u} + (\boldsymbol{\nabla}_0^e\boldsymbol{u})^T + (\boldsymbol{\nabla}_0^e\boldsymbol{u})^T\,\boldsymbol{\nabla}_0^e\boldsymbol{u}\right), \quad \forall e \in [1\,..\,M], \tag{E.18a}$$

$$\boldsymbol{u}_a \cdot \boldsymbol{e}_i = u_\alpha^\mathrm{D}, \qquad \forall (a,i)\ :\ \alpha(a,i) \in D, \tag{E.18b}$$

où

$$\boldsymbol{\nabla}_0^e\boldsymbol{u} = \sum_{a=1}^{N} (\boldsymbol{u}_a \otimes \boldsymbol{B}_{ea}), \tag{E.19}$$

avec $\boldsymbol{B}_{ea} = \boldsymbol{\nabla}_0\mathcal{N}_a(\boldsymbol{X}_e)$ le vecteur encodant la géométrie et la connectivité au point e et au nœud a. D'après l'Eq. (E.10), en utilisant la même interpolation pour les vitesses virtuelles que celle utilisée pour les déplacements dans l'Eq. (E.16) et la règle d'intégration Eq. (E.17), les équations discrètes de la statique s'écrivent :

$$\boldsymbol{R}_a \cdot \boldsymbol{e}_i = 0, \qquad \forall (a,i)\ :\ \alpha(a,i) \notin D \tag{E.20a}$$

avec

$$\boldsymbol{R}_a = \sum_{e=1}^{M} w_e\,\boldsymbol{F}_e\boldsymbol{S}_e\boldsymbol{B}_{ea} - \boldsymbol{T}_a^\mathrm{ext}, \qquad \forall a \in [1\,..\,N], \tag{E.20b}$$

avec $\boldsymbol{F}_e = \boldsymbol{\nabla}_0^e\boldsymbol{u} + \boldsymbol{I}$ et $\boldsymbol{S}_e = \boldsymbol{S}(\boldsymbol{X}_e)$ les tenseurs gradient de la transformation et contrainte au point d'intégration $\boldsymbol{X}_e$, et $\boldsymbol{T}_a^\mathrm{ext}$ le vecteur des efforts extérieurs généralisés au nœud a, défini par :

$$\boldsymbol{T}_a^\mathrm{ext} = \int_{\Omega_0} \mathcal{N}_a\boldsymbol{f}_0\,\mathrm{d}V + \int_{\partial\Omega_0} \mathcal{N}_a\boldsymbol{t}_0\,\mathrm{d}A. \tag{E.21}$$

Ainsi, l'interpolation des déformations découle de l'approximation des déplacements tandis que les contraintes sont simplement évaluées aux points d'intégration. L'ensemble contrainte s'écrit comme l'ensemble des champs discrets (à valeurs continues) déformation-contrainte $(\boldsymbol{E}, \boldsymbol{S}) = \{(\boldsymbol{E}_e, \boldsymbol{S}_e)\}_{e=1}^{M}$ vérifiant les équations ci-dessus :

$$\mathcal{E} = \{(\boldsymbol{E}, \boldsymbol{S}) \mid \text{Eqs. (E.18) et (E.20)}\}. \tag{E.22}$$

Il est un sous-espace de l'espace de phases qui s'écrit $\mathcal{Z} = \mathcal{Z}_{\text{loc}}^{(1)} \times \cdots \times \mathcal{Z}_{\text{loc}}^{(M)}$, avec $\mathcal{Z}_{\text{loc}}^{(e)} = \mathbb{R}_{\text{sym}}^{n\times n} \times \mathbb{R}_{\text{sym}}^{n\times n}$.

L'ensemble données matériau est quasiment inchangé ; il s'agit de l'ensemble des champs discrets, à valeurs discrètes dans la base de données locale $\mathcal{D}_{\text{loc}}^{(e)}$ attachée à chaque point d'intégration :

$$\mathcal{D} = \left\{ (\boldsymbol{E}, \boldsymbol{S}) \ \middle| \ \forall e \in [1 \,..\, M],\, (\boldsymbol{E}_e, \boldsymbol{S}_e) \in \mathcal{D}_{\text{loc}}^{(e)} \right\}, \tag{E.23a}$$

avec

$$\mathcal{D}_{\text{loc}}^{(e)} = \{ (\boldsymbol{E}_i, \boldsymbol{S}_i) \ | \ i \in [1 \,..\, m_e] \} \subset \mathcal{Z}_{\text{loc}}^{e}. \tag{E.23b}$$

La formulation éléments finis du problème *data-driven* s'écrit donc comme la double minimisation sous contrainte

$$\mathcal{S} = \arg \min_{(\boldsymbol{E},\boldsymbol{S})\in\mathcal{E}} \min_{(\boldsymbol{E}',\boldsymbol{S}')\in\mathcal{D}} \|(\boldsymbol{E}, \boldsymbol{S}) - (\boldsymbol{E}', \boldsymbol{S}')\|_{\mathbb{C}}^2, \tag{E.24}$$

avec $\mathcal{E}$ et $\mathcal{D}$ définis par l'Eq. (E.22) et l'Eq. (E.23) et où la norme $\| \bullet \|_{\mathbb{C}}$ Eq. (E.15) est désormais évaluée de façon discrète par

$$\|(\boldsymbol{E}, \boldsymbol{S})\|_{\mathbb{C}} = \sum_e \frac{w_e}{2} \left((\mathbb{C} : \boldsymbol{E}_e) : \boldsymbol{E}_e + \left(\mathbb{C}^{-1} : \boldsymbol{S}_e\right) : \boldsymbol{S}_e \right). \tag{E.25}$$

Solveur *data-driven* Lagrangien Le problème de minimisation est quasiment inchangé par rapport à la formulation en petites déformations : il s'agit d'une double minimisation sur des variables à valeurs discrètes (états matériau) et à valeurs continues (états mécaniques). Seul l'espace des phases a changé (tout en restant un espace de couples de tenseurs symétriques) ce qui induit des contraintes de minimisation non linéaires : la compatibilité Eq. (E.18a) est non linéaire en déplacement et l'équilibre Eq. (E.20) est bilinéaire en déplacement-contrainte.

Pour résoudre le problème de double minimisation, la stratégie de minimisation alternée est de nouveau employée :

- Considérons dans un premier temps l'étape où l'état mécanique de la structure $(\boldsymbol{E}, \boldsymbol{S}) = \{(\boldsymbol{E}_e, \boldsymbol{S}_e)\}_{e=1}^{M} \in \mathcal{E}$ est fixé. L'ensemble solution matériau $\mathcal{S}^{\text{mat}}$ est alors obtenu d'après l'Eq. (E.24) comme

$$\mathcal{S}^{\text{mat}} = \arg \min_{(\boldsymbol{E}',\boldsymbol{S}')\in\mathcal{D}} \|(\boldsymbol{E}, \boldsymbol{S}) - (\boldsymbol{E}', \boldsymbol{S}')\|_{\mathbb{C}}^2 \tag{E.26}$$

  Tous les termes dans la somme Eq. (E.25) étant indépendants les uns des autres, la minimisation Eq. (E.26) peut être faite séparément dans chacune des bases de données $\mathcal{D}_{\text{loc}}^{(e)}$. Ainsi, l'espace solution matériau $\mathcal{S}^{\text{mat}}$ est obtenue par M recherches de plus proches voisins dans l'espace des phases local $\mathbb{R}_{\text{sym}}^{n\times n} \times \mathbb{R}_{\text{sym}}^{n\times n}$:

$$\mathcal{S}^{\text{mat}}(z) = \left\{ \arg \min_{(\boldsymbol{E}_i,\boldsymbol{S}_i)\in\mathcal{D}_{\text{loc}}^{(e)}} d_e\left((\boldsymbol{E}_e, \boldsymbol{S}_e), (\boldsymbol{E}_i, \boldsymbol{S}_i)\right) \ | \ \forall e \in [1 \,..\, M] \right\}. \tag{E.27}$$

où d_e désigne la distance locale entre deux couples déformation-contrainte définie par

$$d_e^2\left((\boldsymbol{E}_e,\boldsymbol{S}_e),(\boldsymbol{E}_i,\boldsymbol{S}_i)\right)=\frac{1}{2}\left(\mathbb{C}:(\boldsymbol{E}_e-\boldsymbol{E}_i)\right):(\boldsymbol{E}_e-\boldsymbol{E}_i) + \frac{1}{2}\left(\mathbb{C}^{-1}:(\boldsymbol{S}_e-\boldsymbol{S}_i)\right):(\boldsymbol{S}_e-\boldsymbol{S}_i). \quad \text{(E.28)}$$

L'état matériau du point d'intégration e, sélectionné comme le plus proche de l'état mécanique considéré est noté $(\boldsymbol{E}^*_{\mathsf{i}e},\boldsymbol{S}^*_{\mathsf{i}e})$ et est défini par

$$d_e\left((\boldsymbol{E}_e,\boldsymbol{S}_e),(\boldsymbol{E}^*_{\mathsf{i}e},\boldsymbol{S}^*_{\mathsf{i}e})\right)\leq d_e\left((\boldsymbol{E}_e,\boldsymbol{S}_e),(\boldsymbol{E}_i,\boldsymbol{S}_i)\right), \quad \forall e\in[1\,..\,M],\ \forall(\boldsymbol{E}_i,\boldsymbol{S}_i)\in\mathcal{D}^{(e)}_{\text{loc}}. \quad \text{(E.29)}$$

L'indice entier $\mathsf{i}e$ est alors la véritable variable d'optimisation. Il relie le point d'intégration e au i-ème point de la base de données $\mathcal{D}^{(e)}_{\text{loc}}$. Pour effectuer efficacement les recherches dans la base de données, nous utilisons des algorithmes basés sur des structures en arbre, qui réduisent drastiquement le coût de calcul (Bentley, 1975; Pedregosa et al., 2011).

- Considérons maintenant l'étape où l'état matériau de la structure $(\boldsymbol{E}^*,\boldsymbol{S}^*)=\{(\boldsymbol{E}^*_{\mathsf{i}e},\boldsymbol{S}^*_{\mathsf{i}e})\}_{e=1}^{M}\in\mathcal{D}$ est fixé. La double minimisation est simplifiée en une simple minimisation sous contrainte. L'ensemble solution mécanique $\mathcal{S}^{\text{mec}}$ est alors déterminé comme suit :

$$\mathcal{S}^{\text{mec}}(z^*)=\arg\min_{(\boldsymbol{E},\boldsymbol{S})\in\mathcal{Z}}\sum_{e=1}^{M}w_e\ d_e\left((\boldsymbol{E}_e,\boldsymbol{S}_e),(\boldsymbol{E}^*_{\mathsf{i}e},\boldsymbol{S}^*_{\mathsf{i}e})\right) \quad \text{(E.30a)}$$

sous contrainte

$$\boldsymbol{E}_e=\frac{1}{2}\left(\boldsymbol{\nabla}^e_0\boldsymbol{u}+(\boldsymbol{\nabla}^e_0\boldsymbol{u})^T+(\boldsymbol{\nabla}^e_0\boldsymbol{u})^T\ \boldsymbol{\nabla}^e_0\boldsymbol{u}\right), \qquad \forall e\in[1\,..\,M], \quad \text{(E.30b)}$$

$$\boldsymbol{u}_a\cdot\boldsymbol{e}_i=u^D_\alpha, \qquad \forall(a,i)\ :\ \alpha(a,i)\in D, \quad \text{(E.30c)}$$

$$\boldsymbol{R}_a\cdot\boldsymbol{e}_i=\sum_{e=1}^{M}w_eF^e_{iJ}S^e_{JK}B^{ea}_K-\boldsymbol{T}^{\text{ext}}_a\cdot\boldsymbol{e}_i=0, \qquad \forall(a,i)\ :\ \alpha(a,i)\notin D. \quad \text{(E.30d)}$$

Comme dans l'approche en petites déformations, la minimisation sous contrainte Eq. (E.30) est traitée par une méthode de multiplicateurs de Lagrange. A nouveau, les équations de stationnarité conduisent à la formulation de deux problèmes pseudo-élastiques. En revanche, ces systèmes sont désormais couplés et non linéaires. Nous les résolvons, comme dans Nguyen and Keip (2018), au moyen d'un schéma de Newton-Raphson. À chaque itération de la minimisation alternée, et à chaque itération du schéma de Newton il faut donc construire et résoudre un nouveau système linéaire dont la taille est $(2n_{\text{dof}}\times 2n_{\text{dof}})$, avec $n_{\text{dof}}=Nn-n_{\text{D}}$ le nombre total de degrés de liberté dans le maillage.

3.2.2 Approche nominale

L'approche nominale a été proposée par Conti et al. (2020) comme une extension du problème *data-driven* tel que présenté dans Conti et al. (2018). C'est donc à ce jour la seule formulation qui possède les preuves mathématiques de convergence des solutions vers les résultats classiques (Conti et al., 2020). L'espace des phases est désormais l'ensemble des couples de tenseur gradient de la transformation – premier tenseur de contrainte de Piola-Kirchhoff. Ces tenseurs ne sont pas symétriques, ce qui ne permet plus d'imposer implicitement l'objectivité et l'équilibre des moments comme précédemment. Dans cette section, nous reprenons la formulation continue telle que présentée dans Conti et al. (2020) et proposons le solveur éléments finis associé, ce qui est à notre connaissance une contribution originale du présent travail.

Formulation continue On considère à nouveau un corps occupant un domaine Ω_0 dans sa configuration de référence et un domaine Ω en équilibre statique, soumis aux efforts $\boldsymbol{f}_0$ et $\boldsymbol{t}_0$, et aux déplacements imposés $\boldsymbol{u}_D$.

Pour assurer la convergence des solutions *data-driven* vers les solutions classiques (avec une loi de comportement), les champs gradient de la transformation-contrainte doivent respecter certaines conditions d'intégrabilité. L'espace des phases est ainsi défini comme

$$\mathcal{Z} = L^p(\Omega_0, \mathcal{Z}_{\text{loc}}) \times L^q(\Omega_0, \mathcal{Z}_{\text{loc}}) \tag{E.31}$$

avec (p, q) tel que $1/p+1/q = 1$ et $L^p(\Omega_0, \mathcal{Z}_{\text{loc}})$ l'ensemble des fonctions continues de Ω_0 dans $\mathcal{Z}_{\text{loc}} = \mathbb{R}^{3\times 3}$ (l'espace local des phases) dont la valeur absolue à la puissance p est Lesbegue intégrable.

L'espace contrainte est défini comme précédemment comme l'ensemble des champs d'état gradient de la transformation-contrainte vérifiant les équations mécaniques. La cinématique s'écrit

$$\boldsymbol{F} = \boldsymbol{\nabla}_0 \boldsymbol{u} + \boldsymbol{I} \qquad \text{dans } \Omega_0, \tag{E.32a}$$

$$\boldsymbol{u} = \boldsymbol{u}_D \qquad \text{sur } \Gamma_{0_N}. \tag{E.32b}$$

et l'équilibre des efforts s'écrit sous forme faible dans sa formulation nominale :

$$\delta W = \int_{\Omega_0} \boldsymbol{P} : \delta \dot{\boldsymbol{F}} \, \mathrm{d}V - \int_{\Omega_0} \boldsymbol{f}_0 \cdot \delta \boldsymbol{v} \, \mathrm{d}V - \int_{\partial\Omega_0} \boldsymbol{t}_0 \, \mathrm{d}A = 0. \tag{E.33}$$

A cela, il faut ajouter l'équilibre des moments $\boldsymbol{F}\boldsymbol{P}^T = \boldsymbol{P}^T\boldsymbol{F}$, qui peut s'écrire comme l'équation vectorielle

$$\boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}\boldsymbol{P}^T\right) = \boldsymbol{0}, \text{ dans } \Omega_0. \tag{E.34}$$

Ainsi, l'ensemble contrainte s'écrit désormais

$$\mathcal{E} = \{(\boldsymbol{F}, \boldsymbol{P}) \in \mathcal{Z} \mid \text{Eqs. (E.32), (E.33) et (E.34)}\}. \tag{E.35}$$

L'ensemble données matériau $\mathcal{D}$ est simplement redéfini par le changement des variables qui sont stockées dans la base de données matériau $\mathcal{D}_{\text{loc}}$:

$$\mathcal{D} = \{(\boldsymbol{F}, \boldsymbol{P}) \in \mathcal{Z} \mid \ (\boldsymbol{F}, \boldsymbol{P})(\boldsymbol{X}) \in \mathcal{D}_{\text{loc}}\}, \tag{E.36a}$$

avec

$$\mathcal{D}_{\text{loc}} = \{(\boldsymbol{F}_i \boldsymbol{P}_i) \in \mathcal{Z}_{\text{loc}} \mid i \in [1 \,..\, m]\}. \tag{E.36b}$$

Enfin, l'ensemble solution des états mécaniques vérifiant les principes mécaniques tout en s'approchant au plus près de la base de données est défini par la minimisation d'une nouvelle fonction distance :

$$\mathcal{S} = \arg \min_{(\boldsymbol{F},\boldsymbol{P}) \in \mathcal{E}} \min_{(\boldsymbol{F}',\boldsymbol{P}') \in \mathcal{D}} d\left((\boldsymbol{F}, \boldsymbol{P}), (\boldsymbol{F}', \boldsymbol{P}')\right) \tag{E.37}$$

avec la distance d définie par

$$d\left((\boldsymbol{F}, \boldsymbol{P}), (\boldsymbol{F}', \boldsymbol{P}')\right) = \int_{\Omega_0} \left(V\left(\boldsymbol{F} - \boldsymbol{F}'\right) + V^*\left(\boldsymbol{P} - \boldsymbol{P}'\right)\right) \mathrm{d}V \tag{E.38}$$

où V est une fonction convexe, de conjuguée convexe V^* définie par

$$V(\boldsymbol{F}) = \frac{1}{p}|\boldsymbol{F}|^p, \quad V(\boldsymbol{F}) = \frac{1}{q}|\boldsymbol{P}|^q. \tag{E.39}$$

Formulation éléments finis La formulation éléments finis est développée exactement comme pour la formulation Lagrangienne. Nous considérons à nouveau un maillage de N nœuds et M points d'intégration. Les déplacements sont interpolés pareillement à l'Eq. (E.16) et la règle d'intégration Eq. (E.17) est inchangée. Les champs tenseur gradient de la transformation et premier tenseur de contrainte de Piola-Kirchhoff sont donc à nouveau évalués seulement aux points d'intégration. Les ensembles données matériaux $\mathcal{D}$ et contrainte $\mathcal{E}$ sont évalués de façon discrète sur le maillage comme précédemment.

Le problème *data-driven* se résume à nouveau à une double minimisation sous contraintes :

$$\mathcal{S}^{\text{mec}}(z^*) = \arg \min_{(\boldsymbol{E},\boldsymbol{S}) \in \mathcal{Z}} \min_{(\boldsymbol{F}',\boldsymbol{P}') \in \mathcal{D}} \sum_{e=1}^{M} w_e \, d_e\left((\boldsymbol{F}_e, \boldsymbol{P}_e), (\boldsymbol{F}'_e, \boldsymbol{P}'_e)\right) \tag{E.40a}$$

sous contrainte

$$\boldsymbol{F}_e = \boldsymbol{\nabla}_0^e \boldsymbol{u} + \boldsymbol{I}, \qquad \forall e \in [1 \,..\, M], \tag{E.40b}$$

$$\boldsymbol{u}_a \cdot \boldsymbol{e}_i = u_\alpha^D, \qquad \forall (a,i) \,:\, \alpha(a,i) \in D, \tag{E.40c}$$

$$\boldsymbol{R}_a \cdot \boldsymbol{e}_i = \sum_{e=1}^{M} w_e P_{iJ}^e B_J^{ea} - \boldsymbol{T}_a^{\text{ext}} \cdot \boldsymbol{e}_i = 0, \qquad \forall (a,i) \,:\, \alpha(a,i) \notin D. \tag{E.40d}$$

$$\boldsymbol{\mathcal{E}} : \left(\boldsymbol{F}_e \, \boldsymbol{P}_e^T\right) = \boldsymbol{0}, \qquad \forall e \in [1 \,..\, M], \tag{E.40e}$$

où la distance locale d_e dans $\mathcal{Z}_{\text{loc}} \times \mathcal{D}_{\text{loc}}^{(e)}$ est définie par[2]

$$d_e\left((\boldsymbol{F}_e, \boldsymbol{P}_e), (\boldsymbol{F}_i, \boldsymbol{P}_i)\right) = \frac{1}{2} C |\boldsymbol{F}_e - \boldsymbol{F}_i|^2 + \frac{1}{2} C^{-1} |\boldsymbol{P}_e - \boldsymbol{P}_i|^2. \tag{E.41}$$

A noter que les contraintes de compatibilité et d'équilibre translationnel Eq. (E.30b) et Eq. (E.30d) sont désormais linéaires. En revanche, s'est ajoutée une contrainte bilinéaire supplémentaire : l'équilibre rotationnel Eq. (E.40e).

Solveur *data-driven* nominal La stratégie de minimisation alternée est à nouveau employée pour résoudre séparément les états matériau et les états mécaniques.

- Les premiers, notés $(\boldsymbol{F}_{\text{ie}}^*, \boldsymbol{P}_{\text{ie}}^*)$, sont déterminés par une recherche des plus proches voisins dans l'espace local des phases. La distance locale quadratique que nous utilisons permet encore de bénéficier des algorithmes de recherche efficaces basés sur des structures en arbre.

- Les états mécaniques résultent d'un problème de minimisation sous contrainte. Cependant, celui-ci ne peut être traité comme précédemment par une méthode de multiplicateurs de Lagrange, dont les équations de stationnarité sont inextricables, car la contrainte supplémentaire est bilinéaire et attachée aux points d'intégration plutôt qu'aux nœuds. Nous développons donc une approche de Lagrangien augmenté qui permet de réduire le nombre d'inconnues du système et d'éviter les problèmes de conditionnement bien connus des approches de pénalité (Bertsekas, 1996). La fonction Lagrangien augmenté (non linéaire) est minimisée au moyen d'un schéma de Newton-Raphson. Celui-ci nécessite que soit construit et résolu un système de taille $(Nn + n_{\text{states}}) \times (Nn + n_{\text{states}})$ à chaque itération du schéma de Newton, chaque itération du Lagrangien augmenté, chaque itération de la minimisation alternée, où $n_{\text{states}} = Mn^2$ est le nombre d'inconnues en contrainte.

Solveur nominal linéaire Une version simplifiée du solveur nominal présenté ci-dessus consiste à ne pas imposer explicitement la contrainte d'équilibre rotationnel, en supposant qu'elle soit vérifiée par les données matériau. La minimisation se fait alors sur l'ensemble affine $\mathcal{E}_0$ défini par l'ensemble des champs gradient de la transformation-contrainte vérifiant la cinématique et l'équilibre translationnel uniquement :

$$\mathcal{E}_0 = \{(\boldsymbol{F}, \boldsymbol{P}) \in \mathcal{Z} \mid \text{Eqs. (E.32) et (E.33)}\}. \tag{E.42}$$

Les équations de stationnarité du Lagrangien augmenté (qui ne comporte plus que des contraintes linéaires en déplacement et contrainte) sont alors simplifiées et aboutissent à deux systèmes pseudo-élastiques linéaires indépendants. Cela est donc plus

[2] Les fonctions V et V^* définies dans l'Eq. (E.39) ont été modifiée pour les besoins numériques : le scalaire C permet de rendre comparable les deux quantités et les paramètres p et q sont pris égaux à 2.

avantageux numériquement parlant. Il convient néanmoins de s'assurer qu'il est suffisant de vérifier l'équilibre des moments dans l'ensemble données matériau $\mathcal{D}$ uniquement. En effet, la contrainte d'équilibre rotationnel étant locale, on peut espérer que la vérifier à chaque point d'intégration à travers l'association état mécanique/état matériau suffit, plutôt que de devoir l'imposer explicitement dans la minimisation de l'état mécanique.

3.2.3 Comparaison théorique des deux approches

Avant de passer à la validation numérique des solveurs proposés nous analysons les principales différences des deux formulations.

1. Il n'existe à notre connaissance aucune preuve mathématique d'existence ou de convergence des solutions au problème *data-driven* Lagrangien, analogues à celles fournies pour la formulation nominale par Conti et al. (2020). Le caractère quadratique de l'équation de compatibilité complexifie le problème. Néanmoins, les théorèmes d'existence et de convergence donnés dans Conti et al. (2020) reposent sur des conditions topologiques particulières pour les ensembles $\mathcal{E}$ et $\mathcal{D}$. Les auteurs proposent notamment des exemples de bases de données générées à partir de modèles hyperélastiques qui respectent ces conditions et pour lesquelles des valeurs optimales des paramètres $(p, q) \neq (2, 2)$ doivent être choisies. Cela ne nous a pas permis d'évaluer numériquement ces modèles, dont nous montrons par ailleurs dans l'Annexe D qu'ils sont non physiques.

2. L'implémentation éléments finis de la formulation nominale est un peu plus complexe et fastidieuse que pour la formulation Lagrangienne : il s'agit en particulier de résoudre un problème de minimisation non contraint à l'intérieur d'une boucle de Lagrangien augmenté, ce qui ajoute une couche à l'algorithme. Par ailleurs, la taille des systèmes non linéaires à résoudre est plus grande dans le cas nominal que dans le cas Lagrangien ($(2n_{\text{dof}})^2$ contre $(Nn + n_{\text{states}})^2$). Ces difficultés sont dues à la non symétrie des tenseurs de l'espace des phases, ce qui nécessite d'imposer explicitement la contrainte d'équilibre rotationnel.

3. Le fléau de la dimensionalité risque d'être d'autant plus prégnant en formulation nominale que Lagrangienne : l'espace de recherche est de plus grande dimension car l'ensemble des composantes indépendantes des états gradient de la transformation-contrainte est de 18 pour un calcul tridimensionnel (contre 12 en formulation Lagrangienne).

En résumé, l'approche Lagrangienne peut être vue comme une « solution d'ingénieur » au problème *data-driven* en grandes transformations. L'approche nominale quant à elle présente des garanties mathématiques mais aboutit à un problème numérique plus lourd. Ce dernier peut être allégé si la contrainte d'équilibre rotationnel n'est pas imposée mais seulement prise en compte dans les données.

3.3 Résultats numériques

Cette partie a pour but de valider l'implémentation éléments finis des deux approches *data-driven* en grandes transformations et de discuter d'aspects pratiques d'utilisation des solveurs. La validation s'entend ici comme la capacité du solveur à converger vers la solution éléments finis classique du problème, lorsque la base de données est générée à partir d'un échantillonnage d'une loi de comportement.

Les bases de données utilisées ici sont en effet générées à partir d'un échantillonnage de la loi de comportement ou bien d'une simulation éléments finis utilisant une loi de comportement. On fait l'hypothèse des contraintes planes et d'un matériau incompressible. On choisit un modèle neo-Hookéen. Dans ce cas, il est possible (*cf.* Annexe C) d'exprimer le modèle entièrement à partir des tenseurs 2D de déformation et de contrainte :

$$\boldsymbol{S} = \mu \left(\boldsymbol{I} - (\det \boldsymbol{C})^{-1} \boldsymbol{C}^{-1} \right) \tag{E.43}$$

avec $\boldsymbol{S}, \boldsymbol{C} \in \mathbb{R}^{2\times 2}_{\text{sym}}$ et $\mu = 1.2\,\text{MPa}$ le module de cisaillement.

On étudie alors trois exemples bidimensionnels de complexité croissante : (i) la traction uniaxiale d'une fine membrane, (ii) la traction d'une fine membrane encastrée à l'autre extrémité et (iii) un chargement plus complexe (traction et cisaillement simultané) d'une fine membrane trouée.

3.3.1 Validation : traction uniaxiale

Dans cet exemple, la base de données est simplement générée à partir d'un échantillonnage de la courbe contrainte-extension $P(\lambda)$ du modèle neo-Hookéen. L'espaces des phases peut alors être observé dans le plan bidimensionnel (λ, P). On génère deux familles de base de données : (i) une qui contient la solution de référence, (ii) une autre qui ne contient pas la solution de référence et pour laquelle on fait varier la densité de points.

Dans ce premier exemple simple, pour lequel les champs mécanique doivent être homogènes sur la structure, on montre déjà plusieurs resultats importants :

- Les deux méthodes Lagrangienne et nominale sont capables de retrouver la solution de référence lorsque celle-ci est dans la base de données ;
- Lorsque la base de données ne contient pas la solution de référence, la solution *data-driven* converge vers celle-ci à mesure que la densité de points dans la base de données augmente, dans les deux cas. Cependant, les solutions Lagrangienne et nominale sont légèrement différentes.

3.3.2 Cas non homogène : traction contrainte

Dans cet exemple, la base de données est constituée des états déformation-contrainte de référence, obtenue par une simulation éléments finis utilisant le modèle neo-Hookéen donné par l'Eq. (E.43). On s'attend ainsi à ce que les deux solveurs retrouvent la solution de référence. On étudie par ailleurs les performances du solveur

nominal linéaire, où la contrainte d'équilibre rotationnel n'est pas imposée sur les états mécaniques (elle est naturellement vérifiée dans la base de données).

On observe tout d'abord que la minimisation alternée n'est pas capable de retrouver cette solution de référence (pourtant entièrement contenue dans la base de données) : le minimum de la fonction objectif n'atteint pas zéro, mais une valeur tout de même faible dans les deux cas. On montre néanmoins en étudiant les cartes de la norme de Von Mises des contraintes de Cauchy que les résultants sont très satisfaisants.

On s'intéresse ensuite à l'initialisation des états matériau à l'entrée du schéma de minimisation alternée : initialisation aléatoire ou initialisation forcée à l'état déformation-contrainte nul ($(\boldsymbol{E}^*_{\mathsf{ie}}, \boldsymbol{S}^*_{\mathsf{ie}}) = (\boldsymbol{0}, \boldsymbol{0})$ ou bien $(\boldsymbol{F}^*_{\mathsf{ie}}, \boldsymbol{P}^*_{\mathsf{ie}}) = (\boldsymbol{I}, \boldsymbol{0})$). On montre que la méthode la plus robuste est l'initialisation à l'état déformation-contrainte nul, dans les deux cas. À noter néanmoins que l'approche Lagrangienne converge quasiment toujours, quelle que soit la méthode d'initialisation choisie tandis que la convergence du solveur nominal est très difficile à obtenir pour des initialisations aléatoires.

On étudie enfin l'erreur à la conservation du moment angulaire lorsqu'on utilise le solveur nominal linéaire. Celle-ci atteint jusqu'à 9 % de la densité d'énergie élastique de déformation.

3.3.3 Chargement complexe

Cet exemple sert à étudier le comportement de la méthode vis-à-vis d'une structure et d'un chargement complexes qui génèrent des champs mécaniques hétérogènes sur la structure (membrane trouée, traction et cisaillement simultanés). La base de données est construite comme précédemment à partir de la solution de référence. Les deux formulations nominale et Lagrangienne sont évaluées séparément :

1. Dans l'exemple précédent, l'homogénéité des champs mécaniques sur la structure pouvait expliquer le fait que ne pas prendre en compte l'équilibre rotationnel ait un impact relativement faible. On évalue ici le solveur nominal linéaire avec un exemple plus complexe. On construit une famille de bases de données en ajoutant les orbites des tenseurs de la base de données de référence, pour étudier l'équivalence invariance par rotation/conservation du moment angulaire.

2. On étudie ensuite la convergence de la solution Lagrangienne avec le nombre de points dans la base de données, en utilisant une méthode d'enrichissement artificiel de la base de données à partir de la solution éléments finis de référence : une autre famille de bases de données est obtenue par la concaténation de plusieurs solutions de référence du même problème avec différents maillages de résolution croissante.

A nouveau, le minimum global d'optimisation n'est pas atteint. Par ailleurs, la simulation Lagrangienne prend jusqu'à 13 fois moins de temps de calcul que la simulation nominale.

On étudie à nouveau le comportement du solveur nominal linéaire. Le temps de calcul est divisé par deux mais l'erreur à la conservation du moment angulaire est cette fois plus élevée (plusieurs centaines de pourcent au bord du trou) et cela impacte également la réponse globale de la structure.

Enfin, l'enrichissement de la base de données par le raffinement du maillage utilisé pour générer la solution de référence ne permet pas d'améliorer significativement les résultats.

3.4 Conclusion

Ce chapitre constitue le cœur de la thèse. On y a développé les solveurs éléments finis pour les deux formulations de l'approche *data-driven* étendue aux grandes transformations, sur la base des travaux de Nguyen and Keip (2018) et Conti et al. (2020). Les deux approches utilisent la stratégie de minimisation alternée initialement proposée par Kirchdoerfer and Ortiz (2016). Dans les deux cas, on utilise des algorithmes efficaces de recherche des plus proches voisins basés sur des structures en arbre pour l'optimisation des états matériau. Dans le cadre Lagrangien, la minimisation sous contrainte des états mécaniques est traitée par une méthode de multiplicateurs de Lagrange. Dans le cadre nominal, elle est traitée par une méthode de Lagrangien augmenté.

Les exemples numériques ont permis de montrer que les résultats *data-driven* sont très satisfaisants en comparaison de la solution de référence, même si le minimum global de l'optimisation n'est pas atteint par le schéma de minimisation alternée. On a également montré qu'il est plus judicieux d'initialiser les états matériau à l'état de déformation-contrainte nul. Par ailleurs, il est démontré que le solveur nominal linéaire n'est pas assez robuste : l'équilibre rotationnel doit être imposé comme une contrainte sur les états mécaniques. Enfin, nous avons tenté d'enrichir la base de données par des moyens qui se sont avérés infructueux. Cela démontre que les notions de richesse ou de couverture de l'espace par les données ne sont pas évidentes. Le Chapitre 4 est donc dédié à une étude approfondie des bases de données nécessaires à la DDCM en grandes transformations.

4 Chapitre 4

4.1 Introduction

4.1.1 Motivation

La base de données matériau est un ingrédient essentiel de la DDCM. En grandes transformations, selon la formulation, elle est composé de couples tensoriels $(\boldsymbol{E}, \boldsymbol{S})$ ou bien $(\boldsymbol{F}, \boldsymbol{P})$. En pratique, ces tenseurs sont évalués dans la base canonique $\mathcal{B} = (\boldsymbol{e}_X, \boldsymbol{e}_Y, \boldsymbol{e}_Z)$. Une base de données pour un problème bidimensionnel s'écrit alors, pour la formulation Lagrangienne et nominale respectivement :

$$\mathcal{D}^{\text{Lag}}_{\text{loc}} = \left\{ \left(E^i_{XX}, E^i_{XY}, E^i_{YY}, S^i_{XX}, S^i_{XY}, S^i_{YY} \right) | i \in [1 \,..\, m] \right\} \tag{E.44a}$$

$$\mathcal{D}^{\text{nom}}_{\text{loc}} = \left\{ \left(F^i_{XX}, F^i_{XY}, F^i_{YX}, F^i_{YY}, P^i_{XX}, P^i_{XY}, P^i_{YX}, P^i_{YY} \right) | i \in [1 \,..\, m] \right\}. \tag{E.44b}$$

La première est une base de données de m points en 6 dimensions quand la seconde comporte m points en 8 dimensions. Dans ce chapitre nous étudions les questions suivantes :

1. Comment peut-on obtenir les bases de données ci-dessus pour un matériau homogène ?

2. Combien faut-il de points dans la base de données et comment doivent-ils être répartis pour obtenir une solution mécanique fiable ?

Pour y répondre on se restreint à (i) l'étude d'un problème bidimensionnel (hypothèse des contraintes planes et de l'incompressibilité du matériau), (ii) l'utilisation de la formulation Lagrangienne seulement car elle est plus mature, utilise un espace des phases de plus petite dimension et est moins coûteuse numériquement que l'approche nominale.

On répond d'abord à la première question dans le reste de cette introduction en présentant des considérations générales sur l'acquisition ou la génération de données. La seconde question est traitée dans le reste du chapitre : d'abord d'un point de vue théorique dans la Section 4.2, puis d'un point de vue numérique dans la Section 4.3.

4.1.2 Point de vue expérimental

Le développement récent des techniques de mesure de champs permet d'instrumenter et de suivre de façon précise des essais expérimentaux complexes. Avec la méthode DDI développée par Leygue et al. (2018) et appliquée avec succès sur des membranes élastomères par Dalémat (2019), on peut désormais avoir accès à des couples déformation-contrainte représentatifs de matériaux hyperélastiques, sans utiliser de modèle de comportement. Nous n'avons néanmoins pas utilisé les résultats expérimentaux de Dalémat pour plusieurs raisons :

- la formulation adoptée dans sa thèse est une formulation Eulérienne : l'espace des phases est constitué des couples déformation vraie–contrainte de Cauchy

$(\ln \boldsymbol{V}, \boldsymbol{\sigma})$, avec $\boldsymbol{V}$ la partie extension de la décomposition polaire gauche de $\boldsymbol{F} = \boldsymbol{V}\boldsymbol{R}$. Les couples d'états matériau $(\ln \boldsymbol{V}_i, \boldsymbol{\sigma}_i)$ ainsi obtenus ne sont pas immédiatement transposables dans un espace Lagrangien $(\boldsymbol{E}, \boldsymbol{S})$ sans faire d'hypothèses particulières.

- Dalémat a également montré dans sa thèse qu'il pouvait être difficile d'obtenir une grande variété de modes de déformation dans les essais expérimentaux : même en utilisant un chargement complexe sur des membranes à plusieurs trous, une grande partie des états mécaniques correspondent à la traction uniaxiale ou au cisaillement pur. Cela pourrait indiquer qu'il est peu probable que la simulation d'une structure ait besoin que l'intégralité de l'espace des phases soit couvert par les données puisqu'une partie restreinte est en réalité explorée par la solution mécanique. Il convient néanmoins de vérifier cette hypothèse avec des données aussi riches que possible, ce que les essais de Dalémat ne permettent pas pour l'instant.

- Enfin, le manque de maturité de l'approche expérimentale DDI nous a freiné dans son utilisation pour valider la DDCM en grandes transformations, elle aussi immature.

4.1.3 Point de vue numérique

De façon alternative les bases de données peuvent être générée numériquement : soit de façon synthétique en échantillonnant une loi de comportement par la suite oubliée, soit de façon numérique en réalisant des simulations aux échelles plus fines et en homogénéisant la réponse effective à l'échelle considérée. La seconde méthode s'applique pour les matériaux hétérogènes où l'hypothèse de séparation des échelles prévaut. Dans un premier temps, nous considérons des matériaux homogènes comme les élastomères, les gels ou les tissus biologiques simples. Nous générons donc des bases de données synthétiques, à partir d'une loi de comportement. Cela permet également de se comparer facilement à la solution éléments finis, prise comme solution de référence.

Dans tous les cas, la génération numérique de bases de données nécessite d'échantillonner l'espace des déformations, décrit par les composantes indépendantes du tenseur $\boldsymbol{E}$. Pour les problèmes mécaniques réduits à deux dimensions, cet espace est tridimensionnel : $\{E_{XX}, E_{YY}, E_{XY}\} \subset \mathbb{R}^3$. Jusqu'à maintenant l'espace des déformations a été échantillonnée au moyen de grilles régulières (Kirchdoerfer and Ortiz, 2016; Eggersmann et al., 2020; Nguyen et al., 2020). Si cette méthode simple peut être pertinente en petites déformations, elle n'est pas appropriée pour les grandes transformations. En effet, il est possible qu'une point de la grille corresponde à une déformation non physique, c'est-à-dire à un tenseur gradient de la transformation au Jacobien négatif.

Dans ce chapitre, nous allons donc explorer d'autres moyens de générer des bases de données synthétiques adaptées à la DDCM en grandes transformations :

1. Sur la base du travail de Kunc and Fritzen (2019a), nous générons des bases de données denses aussi riches que possible (voir 4.2.1).

2. Nous reproduisons uniquement les essais standards (traction uniaxiale, cisaillement pur, traction équi-biaxiale) classiquement utilisés pour identifier les modèles de comportement hyperélastiques (voir 4.2.2).

3. Nous simulons une base de données qui serait issue de la reformulation Lagrangienne de la méthode DDI (voir 4.2.3).

Enfin, en 4.3, nous analysons la prédiction des simulations DDCM obtenues avec chacune des familles de bases de données ci-dessus. En particulier, nous les comparons à la solution éléments finis de référence. Nous déployons également des moyens d'analyse des résultats *data-driven* pour eux-mêmes, sans référence à la solution classique.

4.2 Base de données synthétiques

Comme indiqué précédemment les bases de données synthétiques sont toutes générées au moyen d'un échantillonnage particulier d'une loi de comportement. On se restreint ici aux problèmes bidimensionnels résultant d'hypothèses de contraintes planes et d'incompressibilité du matériau. On utilise alors le modèle neo-Hookéen purement 2D du Chapitre 3 :

$$\check{\boldsymbol{S}}(\boldsymbol{C}) = \mu \left(\boldsymbol{I} - (\det \boldsymbol{C})^{-1} \boldsymbol{C}^{-1} \right) \tag{E.45}$$

pour tout $\boldsymbol{C} \in \mathbb{R}^{2\times 2}_{\text{sym}}$.

4.2.1 Méthode d'échantillonnage dense

La première méthode d'échantillonnage présentée dans cette partie consiste en un échantillonnage dense de l'espace du tenseur des déformation pure symétrique $\boldsymbol{U}$ issu de la décomposition polaire droite du tenseur gradient de la transformation $\boldsymbol{F} = \boldsymbol{R}\boldsymbol{U}$. Cette méthode a été développée par Kunc and Fritzen (2019a) pour de générer des bases de données pour des modèles réduits d'homogénéisation de la réponse d'une microstructure en grandes transformations. La représentation utilisée est dense : pour n'importe quel tenseur $\boldsymbol{U}$ dans l'espace il est possible de raffiner la méthode d'échantillonnage afin de d'obtenir un tenseur $\boldsymbol{U}'$ qui s'en approche aussi près que possible. Afin de générer des bases de données aussi denses et aussi riches que possible nous mettons donc à profit leur méthode :

1. Nous échantillonnons l'espace des tenseurs $\boldsymbol{U}$ de façon dense grâce à la méthode de Kunc and Fritzen (2019a) ;

2. Les déformations de Green-Lagrange sont obtenues par $\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{U}^2 - \boldsymbol{I}\right)$;

3. Les contraintes de Piola-Kirchhoff $\boldsymbol{S} = \check{\boldsymbol{S}}(\boldsymbol{U}^2)$ sont obtenues par le modèle neo-Hookéen $\check{\boldsymbol{S}}$ donné par l'Eq. (E.45).

Cette famille de base de données est appelée DB-DENSE.

Cette partie est organisée comme suit. Dans un premier temps on présente la méthode originale de Kunc and Fritzen (2019a) développée pour des problèmes en trois dimensions. Dans un deuxième temps, on l'adapte aux problèmes en deux dimensions (contraintes planes et incompressible). Dans un troisième temps, on donne un sens physique aux paramètres de la méthode en utilisant les invariants du tenseur de Hencky $\boldsymbol{H} = \ln \boldsymbol{V}$ introduits par Criscione et al. (2000).

Stratégie de séparation amplitude-direction La méthode de Kunc and Fritzen (2019a) repose sur deux représentations distinctes de l'espace des tenseurs $\boldsymbol{U} \in \mathbb{R}^{3\times3}_{\text{sym}}$ admissibles, noté $\mathcal{U}$.

En premier lieu la méthode repose sur la séparation multiplicative volumétrique-déviatorique DDMS du tenseur $\boldsymbol{U} = J^{1/3}\hat{\boldsymbol{U}}$ avec $J = \det \boldsymbol{F} = \det \boldsymbol{U}$ le Jacobien de la transformation et $\hat{\boldsymbol{U}}$ la partie déviatorique du tenseur $\boldsymbol{U}$, c.-à-d. $\det \hat{\boldsymbol{U}} = 1$. L'échantillonnage de l'espace $\mathcal{U}$ est ainsi séparé en l'échantillonnage de déterminant J d'une part et l'échantillonnage des tenseurs déviatoriques $\hat{\boldsymbol{U}}$ d'autre part :

$$\left\{ \left(J^{(m)}\right)^{1/3} \hat{\boldsymbol{U}}^{(j)} \right\}_{m,j=1}^{m=N_{\text{det}},j=N_{\text{dev}}} . \tag{E.46}$$

En deuxième lieu, l'espace des tenseurs $\hat{\boldsymbol{U}}$ est échantillonnée au moyen d'un passage par l'exponentielle matrice des tenseurs de Hencky Langrangien déviatoriques $\hat{\boldsymbol{Y}} = \ln \hat{\boldsymbol{U}} = \ln \operatorname{dev} \boldsymbol{U}$. On note Y la matrice de représentation des tenseurs $\hat{\boldsymbol{Y}}$ dans la base canonique $\mathcal{B} = (\boldsymbol{e}_X, \boldsymbol{e}_Y, \boldsymbol{e}_Z)$. La matrice Y prend ses valeurs dans l'espaces des matrices 3×3 symétriques à trace nulles, qui est un espace de dimension 5, dont on peut facilement écrire une base orthonormale $\mathcal{Y} = \left(Y^{(1)}, Y^{(2)}, Y^{(3)}, Y^{(4)}, Y^{(5)}\right)$. La matrice Y est alors obtenue comme la décomposition linéaire unique des matrices de la base $\mathcal{Y}$:

$$\boldsymbol{Y} = \beta \sum_{k=1}^{5} a_k \, Y^{(k)}, \tag{E.47}$$

avec β l'« amplitude déviatorique » et le vecteur unitaire $a = [a_1, \cdots, a_5]^T \in \mathbb{R}^5$ la « direction » de la matrice Y et donc du tenseur $\boldsymbol{U}$ (Kunc and Fritzen, 2019a).

L'ensemble $\mathcal{U}$ des tenseurs de déformation pure $\boldsymbol{U}$ est donc échantillonné comme suit :

1. Prendre N_{det} valeurs de Jacobien J régulièrement espacées entre $J_{\min}$ et $J_{\max}$;

2. Générer une distribution uniforme de N_{dir} directions a sur la sphère unité de $\mathbb{R}^5$;

3. Prendre N_{amp} valeurs d'amplitude déviatorique β régulièrement espacées dans $[0, \beta_{\max}]$

4. Générer ainsi la base de données des représentations matricielles de $\boldsymbol{U}$ dans la base canonique $\mathcal{B}$ par

$$\left\{ \left(J^{(i)}\right)^{1/3} \exp\left(\beta^{(j)} \sum_{k=1}^{5} \left[a^{(l)}\right]_k Y^{(k)}\right) \right\}_{i,j,l=1}^{i=N_{\text{det}},j=N_{\text{amp}},l=N_{\text{dir}}} \subset \mathcal{U}. \quad \text{(E.48)}$$

Nous proposons dans cette partie une illustration pour les tenseurs $\boldsymbol{U}$ en déformations planes qui ne comportent alors que 3 composantes indépendantes (les composantes U_{XX}, U_{XY}, U_{YY} dans le plan, car $U_{ZZ} = 1$ et $U_{XZ} = U_{YZ} = 0$), ce qui permet une visualisation des résultats. Dans ce cas l'espace des tenseurs $\boldsymbol{U}$ à déterminant $J^{(m)}$ fixé représente un cône arrondi à sa base dans l'espace 3D (U_{XX}, U_{XY}, U_{YY}).

Adaptation aux problèmes bidimensionnels, incompressibles en contraintes planes Nous nous intéressons ici à l'application de la méthode pour les problèmes bidimensionnels, sous l'hypothèse des contraintes planes selon $\boldsymbol{e}_Z$, pour les matériaux incompressibles. Dans l'hypothèse d'incompressibilité, $J = 1$, et les tenseurs complets et déviatoriques sont confondus : $\boldsymbol{U} = \hat{\boldsymbol{U}}$ et $\boldsymbol{Y} = \hat{\boldsymbol{Y}}$. De l'hypothèse des contraintes planes selon $\boldsymbol{e}_Z$, l'espace des représentations matricielles Y des tenseurs $\boldsymbol{Y}$ dans la base $\mathcal{B}$ est dorénavant tridimensionnel :

$$Y = [\boldsymbol{Y}]_{\mathcal{B}} = \begin{pmatrix} c & b & 0 \\ b & d & 0 \\ 0 & 0 & -(c+d) \end{pmatrix} \quad \text{(E.49)}$$

avec c, b, d trois composantes indépendantes. La base $\mathcal{Y}$ est donc réduite à $\mathcal{Y} = \left(Y^{(1)}, Y^{(2)}, Y^{(3)}\right)$. La méthode d'échantillonnage présentée précédemment reste valable, à ceci près que la somme sur les indices k s'arrête à $k = 3$ et que $J^{(m)} = 1$. On note $\bar{\mathcal{U}}$ l'ensemble des tenseurs de déformation pure $\boldsymbol{U}$ respectant les conditions de contraintes planes et l'incompressibilité du matériau ($\det \boldsymbol{U} = 1$).

La partie critique de cette méthode est la génération de directions uniformes sur la sphère unité de $\mathbb{R}^3$, nécessaire pour générer la combinaison linéaire des trois matrices de la base $\mathcal{Y}$. Nous faisons alors appel à une partie de la littérature très dense sur ce sujet et choisissons de générer ces directions au moyen d'une répartition régulière des points sur la sphère, initialement développée par Sloan and Womersley (2004) pour l'interpolation de polynômes sur l'hypersphère unité de $\mathbb{R}^n$ en dimension n quelconque.

Analyse mécanique de la méthode d'échantillonnage dense Dans cette section, nous faisons le pont entre la méthode employée par Kunc and Fritzen (2019a), que nous avons adaptée aux problèmes bidimensionnels incompressibles en contraintes planes, et la définition d'invariants du tenseur de Hencky $\boldsymbol{H} = \ln \boldsymbol{V}$ donnée par Criscione et al. (2000). En effet, le sens physique des paramètres β et $a = [a_1, a_2, a_3] \in \mathbb{R}^3$ n'est pas clair dans la méthode d'échantillonnage proposée

par Kunc and Fritzen (2019a). Par ailleurs, la définition des invariants proposée par Criscione et al. (2000) repose elle aussi sur la séparation volumétrique-déviatorique du tenseur $\boldsymbol{V}$.

Nous remarquons donc d'abord que les invariants de $\boldsymbol{H}$ sont les mêmes que les invariants de $\boldsymbol{Y}$. La théorie déployée par Criscione et al. (2000) s'applique donc de façon équivalentes au tenseur $\boldsymbol{Y}$. Les trois invariants ont un sens physiques bien particulier :

- Le premier invariant K_1 décrit la « quantité de dilatation » et est défini par

$$K_1 = \operatorname{tr} \boldsymbol{Y} = \ln J; \tag{E.50}$$

- Le second invariant K_2 mesure l'« amplitude de distorsion » et est défini par

$$K_2 = \sqrt{\operatorname{dev} \boldsymbol{Y} : \operatorname{dev} \boldsymbol{Y}}; \tag{E.51}$$

- Le troisième invariant K_3 décrit le « mode de distorsion » et est défini par

$$K_3 = \frac{3\sqrt{6}}{(K_2)^3} \det(\operatorname{dev} \boldsymbol{Y}), \tag{E.52}$$

  où la constante $3\sqrt{6}$ est choisie pour que $K_3 \in [-1, 1]$ (Criscione et al., 2000).

Dans le cas incompressible $K_1 = 0$ et les deux invariants restants sont reliés par des relations univoques aux valeurs principales $y_I > y_{II} > y_{III}$ du tenseur $\boldsymbol{Y}$. Il est possible d'unifier ces relations par la fonction

$$\mathrm{y}\,(K_2, K_3, z) = K_2 \sqrt{2/3} \sin\left(\varphi + z\frac{2\pi}{3}\right) \tag{E.53}$$

avec $\varphi = -\arcsin(K_3)/3 \in [-\pi/6, \pi/6]$ et z un entier ne prenant que trois valeurs possibles, qui donne l'ordre des valeurs propres. En étudiant les valeurs propres des tenseurs $\boldsymbol{Y}$ générés par la combinaison linéaire des matrices de la base $\mathcal{Y}$, on peut donc donner un sens physique aux paramètres β et a par l'intermédiaire des invariants K_2 et K_3.

Néanmoins, dans le cas de contraintes planes, l'ordre des trois valeurs principales n'est pas nécessairement respectées dans le plan. On note $\theta = (\boldsymbol{p}_1, \boldsymbol{e}_X)$ l'angle entre le vecteur de la plus grande valeur propre dans le plan $(\boldsymbol{e}_X, \boldsymbol{e}_Y)$ du tenseur bidimensionnel $\boldsymbol{Y}$ et le vecteur $\boldsymbol{e}_X$. Cette angle est appelé l'*angle d'orientation* de l'extension principale dans le plan. Grâce à ces définitions, nous montrons que l'« amplitude déviatorique » β n'est autre que l'invariant K_2, ce qui confirme l'intuition de Kunc and Fritzen (2019a) d'une amplitude de distorsion. Nous montrons également que les coefficients a_1, a_2, a_3 sont reliés par une relation univoque à l'invariant K_3, l'angle d'orientation θ et l'ordre z_1, z_2 des valeurs propres y_1, y_2 dans le plan.

Ainsi la signification mécanique de la méthode d'échantillonnage dense est entièrement maîtrisée.

4.2.2 Méthode d'échantillonnage standard

Grâce l'analyse mécanique précédente il est possible de générer des bases de données sur mesure, en fonction du contenu mécanique désiré. L'échantillonnage de K_2 permet de choisir l'amplitude et donc l'étendue des déformations désirées. L'échantillonnage de K_3 et θ permet de choisir le mode de déformation et l'angle de l'axe selon lequel il s'applique dans le plan. Avec cette approche, il est donc aisé de générer une base de données qui contient les trois essais standards classiquement utilisés pour identifier les modèles de comportement hyperélastiques : $K_3 = 1$ pour la traction uniaxiale, $K_3 = 0$ pour le cisaillement pur et $K_3 = -1$ pour la traction équibiaxiale. Cette famille de base de données vise à répondre à la question suivante : peut-on faire de la simulation DDCM avec les données les plus simples disponibles actuellement (pour lesquelles le tenseur des contraintes est connu notamment) ?

Ce type de base de données est appelé DB-STD dans la suite.

4.2.3 Méthode d'échantillonnage préférentiel

La notion d'« échantillonnage préférentiel » (*importance sampling*) pour les bases de données de la DDCM a été utilisée par Stainier et al. (2019), dans une boucle d'identification-simulation utilisant la méthode DDI pour générer une base de données matériau à partir d'une structure, utilisée ensuite pour la simulation DDCM d'une autre structure, en petites déformations élastiques. Les auteurs ont alors montré que cette base de données donnait de meilleurs résultats qu'une base de données générée à partir d'une grille régulière dans l'espace des tenseurs de déformation linéarisée $\boldsymbol{\varepsilon}$. L'explication avancée est la suivante : la base de données DDI résultant d'une expérience (ici synthétique) sur une structure, elle couvre de façon *préférentielle* la région de l'espace déformation-contrainte susceptible d'être explorée par la solution mécanique de la simulation DDCM. C'est-à-dire que, contrairement à la grille régulière, la répartition des points dans l'espace des phases est telle qu'il existe une concentration de points matériau là où il est susceptible d'exister une concentration d'états mécaniques.

Pour reprendre cette analyse en grandes transformations, nous avons simulé la sortie d'une méthode DDI, telle qu'elle résulterait d'une formulation Lagrangienne du problème. Pour ce faire nous avons opéré un *clustering* par un algorithme standard de *k-means* (recherche de k centroïdes) sur les états de déformation-contrainte $(\boldsymbol{E}, \boldsymbol{S})$ issus de plusieurs simulations éléments finis sur une membrane élastomère à deux trous, sollicitée en traction, cisaillement et compression. Les simulations éléments finis font également appel au modèle de comportement neo-Hookéen donné par l'Eq. (E.45).

4.3 Application numérique

4.3.1 Exemple test et base de données matériau

Nous étudions ici le réponse d'une fine membrane trouée rectangulaire de dimension $l \times h = 20\,\text{mm} \times 50\,\text{mm}$, encastrée à une extrémité et soumise à un déplacement

imposé $\boldsymbol{u}_D = 30\,\text{mm}\,\boldsymbol{e}_X + 5\,\text{mm}\,\boldsymbol{e}_Y$ à l'autre extrémité. L'hypothèse des contraintes planes est adoptée et le matériau est supposé incompressible. Deux maillages sont étudiés : le maillage « grossier » (resp. « fin ») comporte $N = 1092$ (resp. 2495) nœuds et $M = 4060$ (resp. 9500) points d'intégration.

Le modèle de comportement utilisé pour la génération des bases de données et pour produire la solution éléments finis de référence est le modèle neo-Hookéen donné par l'Eq. (E.45), avec $\mu = 1.2\,\text{MPa}$.

On génère des bases de données des trois familles DB-DENSE, DB-STD et DB-DDI selon les méthodes précédemment présentées. Pour chacune des deux premières familles, on fait varier les paramètres d'échantillonnage pour générer des bases de données de différentes tailles :

- 12 bases de données DB-DENSE qui comporte chacune $m = N_{\text{dir}} \times N_{\text{amp}}$ points avec $N_{\text{dir}} \in \{25, 256, 2500, 25\,600\}$ et $N_{\text{amp}} \in \{10, 1000, 1000\}$;
- 3 bases de données DB-STD qui comporte chacune $m = 3N_{K_3} \times N_\theta \times N_{\text{amp}}$ points avec $N_{K_3} = 3$ fixé, $N_\theta = 60$ fixé et $N_{\text{amp}} =\in \{10, 1000, 1000\}$.

La base de données DB-DDI est générée par le *clustering* d'un total de 429 520 couples déformation-contrainte et comprend $m = 2200$ points. Nous remarquons dans ce cas que la base de données DB-DDI comprend moins de points qu'il n'y a de points d'intégration dans le maillage. Nous insistons également sur le fait que celle-ci ne contient pas la solution de référence : les états matériau ont été obtenus par l'analyse de la réponse d'une structure différente (bien que similaire) à celle étudiée ici. Mais surtout, ils ne respectent ni la compatibilité ni l'équilibre car aucune contrainte n'a été imposée dans le *k-means*.

4.3.2 Comparaison avec les solution éléments finis de référence

Dans un premier temps, nous évaluons les performances des bases de données à l'aune de la solution éléments finis obtenue avec le même modèle de comportement, considérée comme solution de référence.

Nous ainsi avons étudié plusieurs critères :

- Les résultats de la minimisation sont évalués en comparant la valeur finale de la fonction objectif des simulations DDCM (distance états mécaniques – états matériau) avec la distance entre la solution de référence et la base de données ;
- L'erreur relative entre les états mécaniques et la solution de référence, en déformation et en contrainte séparément ;
- La réponse globale de la structure (courbe réaction-déplacement)

Ces différents indicateurs nous ont conduits aux mêmes conclusions :

- Les bases de données les plus performantes sont, comme attendu, la plus dense des DB-DENSE et la DB-DDI.

- À l'inverse, les bases DB-STD ne fournissent pas de résultats satisfaisants.
- Pour les bases DB-DENSE, la solution *data-driven* converge clairement vers la solution de référence à mesure que le nombre N_{dir} de directions augmente. À l'inverse, l'impact du nombre N_{amp} d'amplitude déviatorique est très faible. Si l'on se réfère à l'analyse mécanique des directions a, on se souvient qu'elles contiennent en réalité le mode de déformation et l'orientation de cette déformation principale dans le plan. Ainsi, ajouter des modes de déformation à la base de données est plus judicieux que de raffiner les amplitudes de distorsion.
- La base de données DB-DDI présente des performances comparables aux bases DB-DENSE pour un nombre de point très inférieur, ce qui montre l'intérêt de l'échantillonnage préférentiel mentionné plus haut.

4.3.3 Analyse des résultats *data-driven*

Dans cette section, nous développons des outils pour analyser les résultats *data-driven pour eux-mêmes* et non plus en référence avec une solution éléments finis. En effet, l'approche a été introduite pour les cas où il n'est pas possible de représenter fidèlement la réponse du matériau avec un modèle de comportement (complexité de l'élaboration du modèle ou de l'identification de ses paramètres). Dans ces cas, il n'existe pas de « solution de référence ». Il faut donc pouvoir analyser les résultats de la DDCM pour ce qu'ils sont : la réponse mécaniquement admissible de la structure la plus proche possible de ce qui a été mesuré de la réponse matériau à travers seulement un nuage de points discrets.

Les résultats obtenus pour chacune des familles des bases de données sont comparés entre eux, au moyen des indicateurs suivants :

- la convergence de la fonction objectif avec la taille de la base de données ;
- les résultats en contrainte (carte des composantes du tenseur des contraintes de Cauchy) ;
- les résultats en déformation (projection des états de déformation dans un espace (K_2, K_3, θ)) ;
- l'écart à l'isotropie du matériau, non explicitement imposée dans la simulation (angle de désalignement entre les vecteurs propres des tenseurs $\boldsymbol{E}$ et $\boldsymbol{S}$ (Leygue et al., 2018; Dalémat et al., 2019))

Les résultats précédents sont confirmés et on ajoute également les conclusions suivantes :

- Il apparaît tout d'abord que le mode de traction uniaxiale est massivement représenté dans la structure (malgré un chargement assez complexe et la présence du trou). On retrouve en ce sens les résultats expérimentaux de Dalémat (2019).

- Les deux bases de données DB-DENSE et DB-DDI donnent des résultats très comparables, alors que le nombre de points dans la DB-DDI est très inférieur ($2200 \ll 25\,600\,000$). Ceci peut notamment s'expliquer par le fait que peu de points de la DB-DENSE ont finalement été utilisés au cours de la simulation : seulement 131 149 (resp. 259 257) points ont été sélectionnés au moins une fois comme état matériau au cours des pas de chargement et de la minimisation alternée pour le maillage grossier (resp. fin), sur les 25 600 000 disponibles.

- On remarque que les états mécaniques obtenus avec les bases DB-STD sont plus riches (en termes de modes de déformation) que les états matériau présents dans la base de données. La capacité d'« extrapolation » du solveur reste néanmoins très limitée car les autres résultats sont nettement moins satisfaisants en comparaison avec les autres bases de données.

Le résultat principal de cette étude est que la qualité de la solution mécanique est d'autant plus fine que la base de données est riche, au sens de la variété des modes, plutôt que des amplitudes, de déformation.

4.4 Conclusion

Ce chapitre a été consacré à la génération et l'étude des bases de données pour la DDCM en grandes transformations. On s'est restreint à la formulation Lagrangienne et à un exemple bidimensionnel pour des raisons pratiques mais une analyse similaire peut être conduite pour la formulation nominale et des exemples tridimensionnels. Pour ce faire, nous avons généré des bases de données synthétiques, c.-à-d. par l'intermédiaire d'un modèle de comportement par la suite oublié dans les simulations DDCM. Cela a permis de se comparer à la solution éléments finis classique obtenue avec le même modèle. Les bases de données ont été générées par un échantillonnage de l'espace des déformations. Trois méthodes ont été étudiées :

- Une méthode d'échantillonnage dit dense, qui permet de créer des bases aussi riches que possible. Nous nous sommes inspirées de la méthode présentée dans Kunc and Fritzen (2019a) et l'avons reliée au travail Criscione et al. (2000) pour en fournir une analyse mécanique pertinente.

- Une méthode d'échantillonnage dit standard nous a permis de simuler les résultats expérimentaux des essais de traction uniaxiale, cisaillement pur et traction équi-biaxiale, communément utilisés pour identifier les modèles de comportement hyperélastiques.

- Une méthode d'échantillonnage dit préférentiel qui simule la méthode d'identification DDI proposée par Leygue et al. (2018); Dalémat (2019).

Les performances des trois familles de bases de données ont été évaluées en comparant les solution *data-driven* et éléments finis. Comme attendu la plus riche DB-DENSE est la plus performante, au prix d'un nombre très important de points néanmoins. Juste derrière, la DB-DDI a démontré la pertinence de l'échantillonnage

préférentiel, qui nécessite beaucoup moins de points. Enfin, la DB-STD a démontré ses limites pour le problème complexe considéré.

Dans la dernière partie du chapitre nous avons développé des outils d'analyse de la solution *data-driven pour elle-même*. Nous avons étudié en particulier quatre indicateurs : (i) la convergence de la fonction objectif avec la taille de la base de données, (ii) la répartition des composantes des contraintes de Cauchy dans la structure, (iii) l'analyse mécanique des états de déformation, (iv) l'écart à l'isotropie. A nouveau la plus riche DB-DENSE a surpassé les autres base de données, bien que les résultats DB-DDI soient très satisfaisants. Il est néanmoins important de noter qu'une partie très réduite seulement du nombre de points est utilisé par la DDCM : cela indique que les états mécaniques d'un problème donné ne sont susceptibles de couvrir qu'une région réduite de l'espace des phases. Il convient alors que cette région soit concentrée en points matériau : le nombre de points matériau dans la base de données importe peu, c'est leur répartition qui compte.

Grâce au travail présenté dans ce chapitre, nous sommes maintenant capables de (i) générer des bases de données matériau aussi riches et denses que nécessaire, et (ii) d'analyser les résultats de la DDCM au moyen d'outils mécaniquement motivés. Le prochain chapitre est donc consacré à appliquer ces contributions dans le cadre d'un problème tridimensionnel complexe.

5 Chapitre 5 : Application à un problème tridimensionnel

5.1 Introduction

Ce court chapitre est dédié à l'application la DDCM en grandes transformations à un problème tridimensionnel, utilisant les solveurs éléments finis développés dans le Chapitre 3. De plus, nous utilisons la méthode de génération dite dense de la base de données développée au Chapitre 4. Le chapitre est organisé comme suit : le problème mécanique est décrit en 5.2.1 et la génération de la base de données est exposée en 5.2.2. Les résultats obtenus pour les deux formulations sont présentés 6.3 et discutés 5.4.

5.2 Méthodes

5.2.1 Problème mécanique

Nous étudions dans ce chapitre la torsion d'une colonne en forme de T, inspiré de l'exemple présenté dans Bonet et al. (2016, Chapitre 10, page 302). La géométrie et le maillage (éléments hexahèdres à 8 points d'intégration) du problème sont donnés sur la Fig. E.1. Le pied de la colonne est encastrée tandis que les surfaces latérales

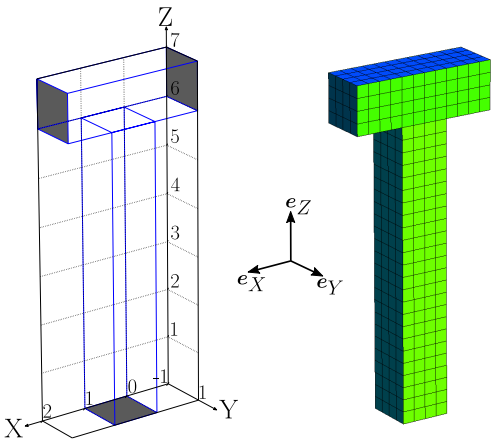


FIGURE E.1 – Géométrie et maillage du problème en trois dimensions.

de la partie supérieure du T (grisées sur la Fig. E.1) sont soumises à un déplacement imposé entraînant une rotation d'angle θ autour de l'axe Z ($X = Y = 0$). Il faut noter que l'axe de rotation n'est pas l'axe de symétrie de la géométrie ($X = Y = 0.5$). La simulation est conduite en 45 pas de chargement réguliers.

5.2.2 Base de données matériau

Défi en trois dimensions Jusqu'à présent nous n'avons réalisé que des simulations en deux dimensions. L'espace des phases était donc de dimension $3 + 3 = 6$

pour la formulation Lagrangienne et $4+4=8$ pour la formulation nominale. La dimension de l'espace des phases est doublée lorsqu'on passe en trois dimensions : $6+6=12$ et $9+9=18$ pour les deux formulations respectivement. Cette explosion de la dimensionalité pose un double problème : (i) la définition de distance et la recherche dans des espaces de grandes dimensions sont des problèmes complexes, (ii) pour générer une base de données, il faut échantillonner un espace des déformations de plus grande dimension. Le fléau de la dimensionalité est alors réellement en jeu : il va falloir énormément de point pour « remplir » l'espace des phases.

Par ailleurs, nous souhaitons dans notre étude comparer les deux formulations en grandes transformations afin de les discriminer. Les solutions doivent donc être calculées avec des bases de données comparables : les deux types de couples déformation-contrainte $(\boldsymbol{E}, \boldsymbol{S})$ et $(\boldsymbol{F}, \boldsymbol{S})$ doivent représenter la même réponse matériau, c.-à-d. échantillonner la même région de l'espace dans leur espace des phases respectif.

Nous capitalisons sur le travail précédent pour générer *d'abord* une base de données Lagrangienne :

$$\mathcal{D}_{\text{loc}}^{\text{Lag}} = \left\{ (\boldsymbol{E}_i, \boldsymbol{S}_i) \in \mathbb{R}^{3\times3}_{\text{sym}} \times \mathbb{R}^{3\times3}_{\text{sym}} \mid i \in [1\,..\,m] \right\}. \tag{E.54}$$

Nous utilisons la méthode d'échantillonnage dense en trois dimensions, en reprenant exactement les travaux de Kunc and Fritzen (2019a).

La base de données nominale

$$\mathcal{D}_{\text{loc}}^{\text{nom}} = \left\{ (\boldsymbol{F}_i, \boldsymbol{P}_i) \in \mathbb{R}^{3\times3} \times \mathbb{R}^{3\times3} \mid i \in [1\,..\,m] \right\} \tag{E.55}$$

est ensuite obtenue à partir de $\mathcal{D}_{\text{loc}}^{\text{Lag}}$ d'après les relations entre les différents tenseurs :

$$\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{F}^T\boldsymbol{F} - \boldsymbol{I}\right) \tag{E.56}$$

$$\boldsymbol{S} = \boldsymbol{F}^{-1}\boldsymbol{P}. \tag{E.57}$$

Formulation Lagrangienne On reprend ici la méthode originale de Kunc and Fritzen (2019a) en trois dimensions. Les tenseurs de déformation pure $\boldsymbol{U}$ sont obtenu selon l'Eq. (E.48). Le modèle de comportement utilisé ici est un modèle neo-Hookéen isotrope compressible :

$$\check{\boldsymbol{S}}(\boldsymbol{C}) = \mu\left(\boldsymbol{I} - \boldsymbol{C}^{-1}\right) + \lambda \ln(J)\boldsymbol{C}^{-1}, \tag{E.58}$$

avec $\lambda = \mu = 100$ tels que le module d'Young est $E = 250$ (sans unité) et le cœfficient de Poisson est $\nu = 0.25$. La base de données Lagrangienne s'écrit donc

$$\mathcal{D}_{\text{loc}}^{\text{Lag}} = \left\{ \left(\boldsymbol{E}^{(i,j,l)}, \boldsymbol{S}^{(i,j,l)} \right) \mid i \in [1\,..\,N_{\text{det}}],\, j \in [1\,..\,N_{\text{amp}}],\, l \in [1\,..\,N_{\text{dir}}] \right\}, \tag{E.59}$$

avec $N_{\text{det}} = 49$ le nombre de valeurs de Jacobien $J \in [0,8; 1,2]$ dans l'échantillon, $N_{\text{amp}} = 100$ le nombre de valeurs d'amplitude déviatorique $\beta \in]0; 0,5]$ et $N_{\text{dir}} = 2047$ le nombre de directions $a \in \mathbb{R}^5$. Le nombre total de points est donc de $m = 10\,035\,200$.

Formulation nominale La base de données nominale est obtenue à partir de la base de données Lagrangienne. Pour ce faire on réécrit les relations Eqs. (E.56) comme

$$\boldsymbol{F} = \boldsymbol{R}\boldsymbol{U} \quad \text{with } \boldsymbol{U}^2 = 2\boldsymbol{E} + \boldsymbol{I} \tag{E.60}$$

$$\boldsymbol{P} = \boldsymbol{F}\boldsymbol{S} = \boldsymbol{R}\boldsymbol{U}\boldsymbol{S}, \tag{E.61}$$

avec $\boldsymbol{R} \in SO(3)$ un tenseur de rotation du groupe spécial orthogonal en trois dimensions $SO(3) = \left\{R \in \mathbb{R}^{3\times3} \mid R^T R = RR^T = I, \det R = 1\right\}$. Il est donc nécessaire d'échantillonner en plus l'espace $SO(3)$ pour obtenir une base de données nominale. Le nombre de points dans l'échantillon de $SO(3)$ est noté N_R. Pour s'assurer que n'importe quelle rotation $\boldsymbol{R}$ soit prise en compte, il faut que N_R soit très grand. Le nombre de points dans la base de données nominale $\mathcal{D}_{\text{loc}}^{\text{nom}}$ est alors $m \times N_R$, ce qui peut vite devenir énorme. Pourtant, il est peu probable que l'intégralité des m points Lagrangiens soient nécessaire à la simulation DDCM du problème étudié. En effet, on a vu au chapitre précédent que peu de points de la base de données DB-DENSE étaient réellement utilisés pour le problème bidimensionnel et on s'attend à un comportement similaire ici. Pour réduire le nombre de points dans la base de données nominale on utilise une première simulation DDCM Lagrangienne comme un moyen de sélectionner les points pertinents dans la base de données $\mathcal{D}_{\text{loc}}^{\text{Lag}}$: on ne conserve dans une base de données notée $\mathcal{D}_{\text{loc}}^{\text{Lagred}}$ que les points qui sont sélectionnés au moins une fois comme état matériau, au cours des pas de chargement t et des itérations k du schéma de minimisation alternée. La base de données nominale $\mathcal{D}_{\text{loc}}^{\text{nom}}$ est alors obtenue comme le produit de l'échantillonnage de $SO(3)$ et de $\mathcal{D}_{\text{loc}}^{\text{Lagred}}$.

Bilan La stratégie de génération de base de données comparables pour les deux formulations du problème tridimensionnel en grandes transformations est donnée sur la Fig E.2.

5.3 Résultats

5.3.1 Formulation nominale

Les tests conduits avec la formulation nominale n'ont pas abouti : la minimisation n'a pas convergé pour le problème considéré. Des études préliminaires avec un maillage plus grossier et une base de données entièrement constituée de la solution de référence conduisent également à des résultats non satisfaisants, ce qui montre que la méthode de génération de la base de données employée ici n'est pas la seule cause de non convergence. Nous pensons que la dimensionalité de l'espace des phases peut jouer mais nous n'excluons pas que l'immaturité du solveur nominal (développé dans les derniers mois de la thèse) explique qu'une solution n'ait pas pu être trouvée. Il convient d'explorer plus avant ce problème dans des travaux futurs.

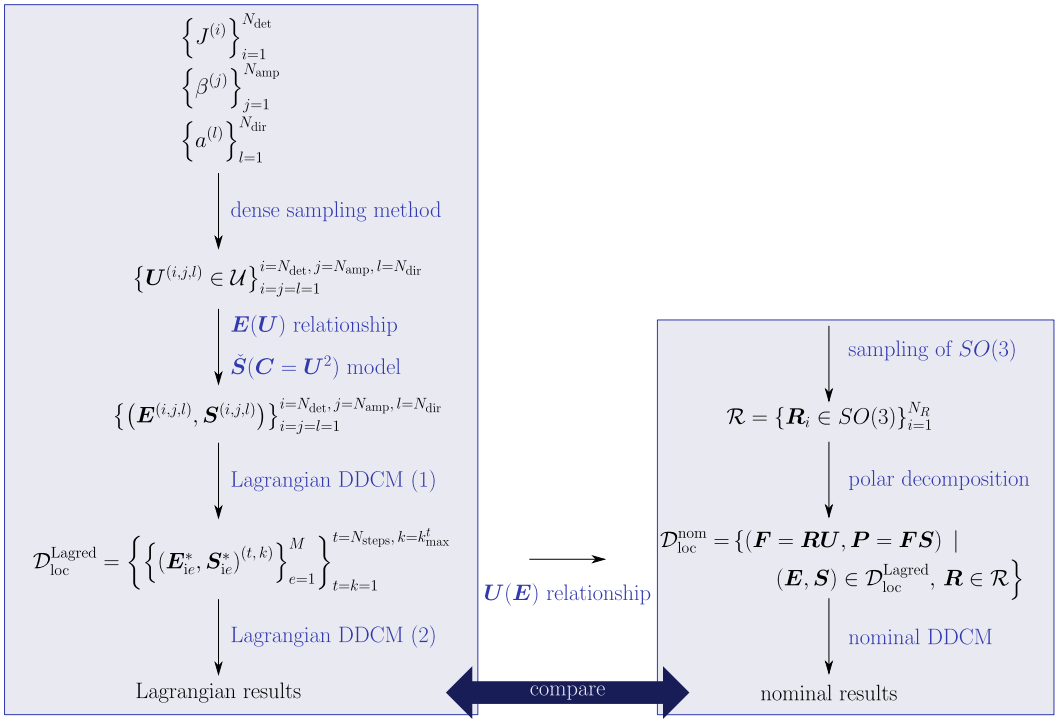


FIGURE E.2 – Stratégie de génération de bases de données Lagrangiennes (gauche) et nominales (droite) pour effectuer des calculs DDCM en trois dimensions comparables.

5.3.2 Formulation Lagrangienne

Nous nous concentrons donc dorénavant sur les résultats de la formulation Lagrangienne. Nous étudions les résultats issus de la deuxième simulation utilisant la base de données réduites $\mathcal{D}_{\text{loc}}^{\text{Lagred}}$ qui comporte 75 077 sur les $m = 10\,035\,200$ de la base de données originale.

Déformée et résultante Nous étudions d'abord la déformée du maillage : comme on peut le voir sur la Fig. E.3 les résultats sont en concordance avec le chargement imposé.

Nous étudions ensuite l'évolution du torseur (force et moment) résultant sur la face encastrée en bas avec l'angle de rotation imposé. L'analyse qualitative des résultats montre un comportement attendu et les résultats éléments finis sont fournis à titre indicatif.

Résultats de la minimisation Nous étudions ensuite la convergence de l'algorithme de minimisation alternée au sein de chaque pas de chargement. On constate que plus le déplacement imposé est grand, plus la convergence est lente et plus la valeur finale de la fonction objectif est grande. On sait d'après la Fig. E.3 que plus la rotation imposée est grande, plus les déformations dans la structure sont grandes. On en conclut que la minimisation est d'autant plus difficile à résoudre en grandes

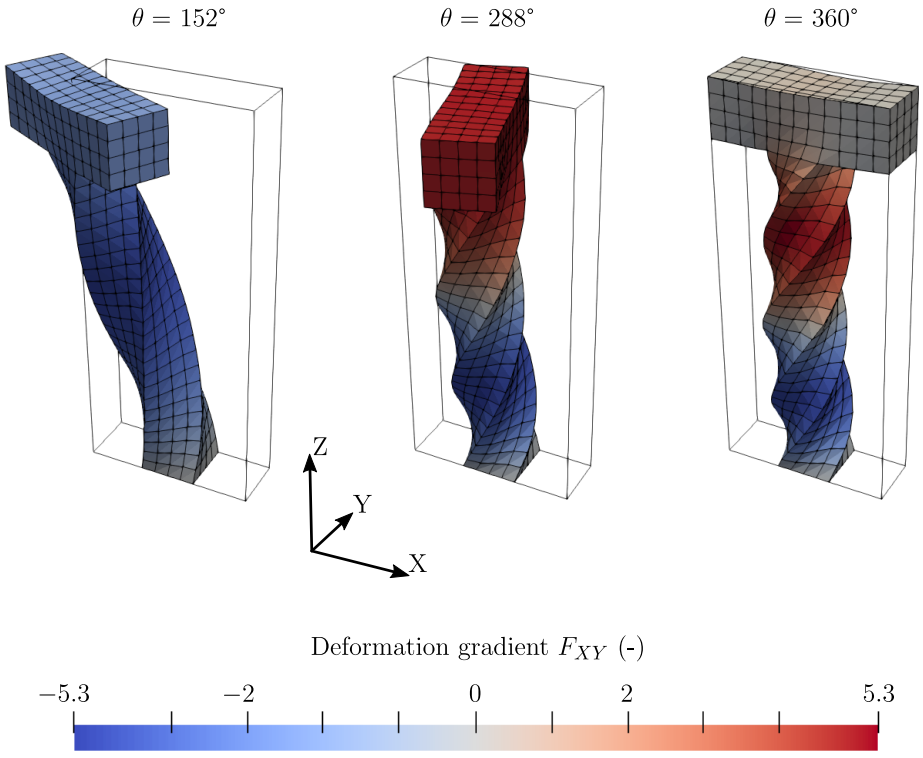


FIGURE E.3 – Maillage déformé et glissement F_{XY} pour différents pas de chargement.

transformations.

5.4 Discussion

5.4.1 Retrouver l'isotropie

Dans cette section, on s'intéresse à nouveau à l'écart à l'isotropie du matériau, vérifiée par la base de données mais non imposée sur les états mécaniques. On mesure les statistiques des angles entre les vecteurs propres des tenseurs $\boldsymbol{E}_e$ et $\boldsymbol{S}_e$ pour chaque pas de chargement. Pour 60 % des points d'intégrations du maillage, l'angle d'écart à l'isotropie est inférieur à 12°. Pour des valeurs proches de 90°, on montre par une analyse détaillée qu'il s'agit majoritairement d'une incertitude sur la détermination de la plus grande valeur propre de chacun des tenseurs. Ceci permet de conclure que la base de données réduite $\mathcal{D}_{\text{loc}}^{\text{Lagred}}$ conduit à des résultats assez satisfaisants. Le moyen de l'enrichir de modes de déformation pertinents pour améliorer ce critère reste cependant un sujet ouvert.

5.4.2 Modes de déformation

Finalement, on analyse l'amplitude de dilatation, l'amplitude de distorsion et le mode de déformation de chaque état mécanique au moyen des invariants (K_1, K_2, K_3) du tenseur de déformation pure $\boldsymbol{U}$ présentés au Chapitre 4. Il apparaît que les répartition des états mécaniques et matériau dans cet espace à trois dimensions sont très différentes. On voit notamment que la diversité d'amplitudes de distorsion K_2 est plus élevée parmi les états mécaniques que matériau. De même pour les modes de déformation K_3. Enfin, en accord avec le chargement imposé, une grande partie des points d'intégration se trouve en contraction volumétrique : $K_1 < 0$.

5.5 Conclusion

Ce chapitre nous a permis de tester les solveurs DDCM en grandes transformations pour un problème complexe en trois dimensions. L'enjeu de la dimensionalité de l'espace des phases et donc de la taille de la base de données est évidemment plus prégnant qu'en deux dimensions. Nous avons notamment présenté une stratégie pour générer des bases de données adaptées à chaque formulation (Lagrangienne et nominale). Pour la formulation Lagrangienne, il s'agit essentiellement d'appliquer la méthode d'échantillonnage de l'espace des tenseurs $\boldsymbol{U}$ développée par Kunc and Fritzen (2019a). Pour la formulation nominale, il s'agit de sélectionner les points pertinents dans la base de données Lagrangienne pour les multiplier avec un échantillonnage dense des tenseurs $\boldsymbol{R}$ dans $SO(3)$.

Malheureusement, il n'a pas été possible d'obtenir de résultat avec l'approche nominale en raison de problèmes de convergence du solveur, encore non élucidés à ce jour. Néanmoins, la formulation Lagrangienne a fourni des résultats très satisfaisants pour une première étude. Dans une première simulation, la base de données comprend 10 035 200 points mais seulement 75 077 sont réellement utilisés dans la simulation. Ce résultat était attendu : la méthode d'échantillonnage dense couvre l'ensemble des amplitudes et modes de déformation possibles, tandis que les amplitudes et modes de déformation présents dans une structure particulière sont nécessairement en nombre restreints, c.-à-d. que seule une certaine région de l'espace des phases est parcourue par la solution mécanique. Tous les points de la base de données matériau en dehors de cette région semblent donc inutiles.

Cette étude est un premier pas encourageant vers des simulations DDCM en grandes transformations en trois dimensions. Nous souhaitons en effet insister sur le fait que cet étude est, à notre connaissance, la première du genre pour les solveurs DDCM (en grandes transformations) : dans l'exemple 3D présenté dans Kirchdoerfer and Ortiz (2016), la base de données était en réalité réduite à une base 2D par l'hypothèse des contraintes planes et les maillages ne comportaient qu'un ou deux éléments dans l'épaisseur. Bien sûr, un étude systématique de l'influence des différents paramètres N_{det} ou N_{dir} devrait être conduite dans des travaux futurs. L'adaptation du solveur nominal devra également être maîtrisée.

Pour conclure sur cette deuxième partie de la thèse, nous souhaitons mettre en avant une question qui reste non résolue. Dans les exemples 2D et 3D présentés jusqu'ici, la réponse mécaniquement admissible obtenue par la DDCM est aussi proche que possible que ce qui est connu de la réponse matériau, c.-à-d. la base de données. Néanmoins, on a vu que cette connaissance est parfois incomplète ou non adaptée au problème mécanique considéré. Il conviendrait donc de *compléter* de façon adaptative la base de données. C'est notamment ce que nous avons tenté de faire en ajoutant des amplitudes ou modes de déformation en augmentant les nombres N_{amp} et N_{dir}. Il serait cependant plus intéressant de trouver un moyen systématique d'enrichir la base de données. Dans la dernière partie de cette thèse, au Chapitre 6, nous proposons une première stratégie adaptative d'enrichissement de la base de données pour le cas particulier des matériaux hétérogènes.

6 Chapitre 6 : une première approche pour des simulations multiéchelles pilotée par les données

6.1 Introduction

Jusqu'à maintenant, les bases de données utilisées dans les exemples présentés étaient artificielles : elles ont été générées par l'échantillonnage d'un modèle de comportement. Nous avons ainsi pu les comparer à la solution éléments finis correspondante pour valider nos approches DDCM en grandes transformations.

Cependant, la méthode DDCM a été introduite dans l'objectif d'utiliser des données *brutes*, sans passer pas des lois de comportement. Ces données sur la réponse matériau peuvent provenir de deux sources :

- Les mesures expérimentales, par exemple en utilisant les mesures de champs et la DDI. Il reste encore des défis à relever pour pouvoir utiliser ces approches dans un cadre DDCM, comme nous l'avons montrés au Chapitre 2 et au Chapitre 4.
- L'homogénéisation numérique, que ce soit par des simulations éléments finis sur une microstructure (polycristal, composite plastique renforcé fibres courtes, élastomère chargé, *etc.*) ou même des calculs aux échelles très fines (*ab initio*, dynamique moléculaire), peut fournir plus aisément (moyennant un coût de calcul parfois important) des couples déformation-contrainte représentatifs de la réponse homogénéisée d'un matériau hétérogène.

Dans ce chapitre, nous explorons la deuxième possibilité dans le cadre de matériaux hétérogènes où l'hypothèse de séparation des échelles est vérifiée. Nous nous limitons aux petites déformations élastiques et nous utilisons l'approche originale de la DDCM (Kirchdoerfer and Ortiz, 2016).

6.1.1 Principes et défis de l'homogénéisation numérique

Dans cette section, nous revenons rapidement sur les aspects essentiels de l'homogénéisation numérique.

On considère un matériau multiphasé, c.-à-d. qui contient des hétérogénéités de petites tailles, incorporées dans une matrice. La taille de ces hétérogénéités est supposées beaucoup plus petite que la taille de la structure composée du matériau hétérogène considéré (séparation des échelles). On présente alors le problème aux limites à résoudre sur le Volume Élémentaire Représentatif (VER) de la microstructure (compatibilité, équilibre, loi de comportement de chacune des hétérogénéités) en chaque point de la macrostructure. Les conditions aux limites sont prescrites par le tenseur de déformation macroscopique $\bar{\boldsymbol{\varepsilon}}$ et le tenseur de contrainte macroscopique correspondant à ce chargement est obtenu par une moyenne volumique du champ de contrainte sur le VER. La loi de comportement effective du matériau n'est alors plus une relation mathématique explicite mais est obtenue par l'homogénéisation de la réponse microstructurale qui se fait en trois étapes : (i) localisation (application

des conditions aux limites), (ii) résolution du problème mécanique microstructural, (iii) homogénéisation (moyenne volumique des champs). Ce processus est illustré sur la Fig. E.4.

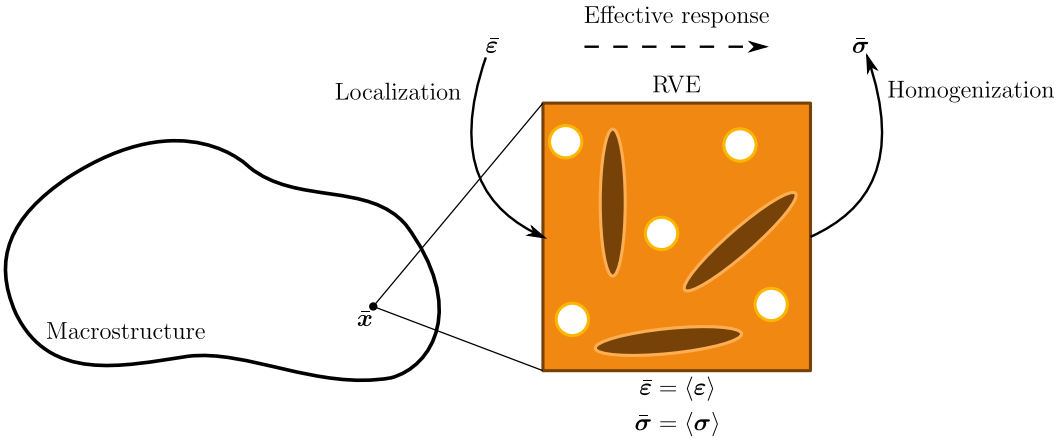


FIGURE E.4 – Loi de comportement effective d'un composite, obtenue par la résolution d'un problème aux limites sur le VER de la microstructure. Inspiré de Brassart (2011).

Après une courte revue des différentes méthodes de la littérature pour résoudre ce problème multiéchelle (théorie des champs moyens, théorie des modules effectifs en élasticité linéaire, méthode champs complets par homogénéisation numérique), on s'intéresse à la méthode de calcul multiéchelle popularisée sous le nom de FE^2 (pour « éléments finis au carré ») par Feyel (1999). Cette méthode multiéchelle consiste à réaliser deux calculs éléments finis emboîtés : le premier est conduit sur la structure (macroéchelle), le deuxième est conduit en chaque point d'intégration macroscopique sur le VER d'une microstructure (microéchelle). Les conditions aux limites du calcul microéchelle sont déterminées par l'état mécanique du point d'intégration correspondant à la macroéchelle. Cette approche permet d'obtenir, sans hypothèse forte sur le comportement effectif du matériau, la réponse macroscopique d'une structure à moindre coût, comparé à un calcul éléments finis unique dont la résolution serait suffisamment fine pour mailler toutes les hétérogénéités. Elle reste néanmoins assez coûteuse car (i) il faut conduire de façon systématique un calcul microstructural à chaque point d'intégration du maillage macroscopique, (ii) pour les comportements non linéaires, il est nécessaire de connaître également le module tangent effectif de la microstructure, qui est typiquement obtenu par perturbation numérique, ce qui entraîne des évaluations microscopiques supplémentaires. Le maillage du VER de la microstructure étant souvent très raffiné, la facteur limitant en terme de coût calcul est le nombre d'évaluations microscopiques :

$$N_{\text{éval. micro.}} = N_{\text{it.}} \times M \times (1 + n(n-1)), \tag{E.62}$$

avec N_{it} le nombre d'itérations nécessaires pour atteindre l'équilibre de la macrostructure (typiquement dans un schéma de Newton-Raphson), M le nombre de

points d'intégrations sur maillage macroscopique, $n = 2, 3$ la dimension du problème. Cependant, il est probable que certaines de ces évaluations soient redondantes et donc inutiles : au cours de la simulation, deux points macroscopiques peuvent très certainement partager le même état déformation-contrainte mécanique et donc conduire à deux évaluations microscopiques similaires. Des stratégies d'accélération des schémas FE^2 ont donc été mises en place pour réduire le temps de calcul en évitant ces redondance. Par exemple, Klusemann and Ortiz (2015) stockent les états de déformation-contrainte macroscopiques dans un espace des phases et interpolent, au moyen d'une subdivision simpliciale de l'espace, la réponse microscopique entre deux points voisins déjà calculés. Le nombre d'évaluations est ainsi fortement réduit.

6.1.2 Vers l'homogénéisation multiéchelle pilotée par les données

Une autre approche pour réduire les coûts des calculs multiéchelles consiste à générer une base de données représentative du comportement effectif, à partir de nombreux calculs microstructuraux, dans un processus « hors-ligne ». Cette base de données peut typiquement être utilisée pour alimenter un métamodèle de la réponse macroscopique, allant de la réduction de modèle au réseau de neurones (voir la revue de Matouš et al. (2017) pour une liste exhaustive de ces approches). Cependant, ces méthodes reposent toujours sur l'existence d'un modèle de comportement : il est soit interpolé, réduit ou identifié à partir des données. En rupture avec cette vision, Xu et al. (2020) ont proposé une première utilisation de la DDCM pour les matériaux hétérogènes : la base de données des couples déformation-contrainte macroscopiques est générée en amont puis utilisée dans un calcul DDCM sur la structure. Cette méthode requiert de définir *a priori* les conditions aux limites appropriées qu'il faut appliquer aux calculs microstructuraux, c.-à-d. déterminer à l'avance la région de l'espace des phases susceptible d'être pertinente pour le calcul macroéchelle.

Nous proposons plutôt dans ce chapitre une première stratégie pour réaliser des calculs FE^2 utilisant la DDCM : la résolution du problème macroéchelle est faite par DDCM et la résolution du problème microéchelle est faite par un calcul éléments finis classique. Entre les deux échelles, la base de données est enrichie de façon adaptative en fonction de la réponse mécanique de la macrostructure, dans un processus « en-ligne ». De cette façon, il n'est pas besoin de connaître *a priori* la réponse de la macrostructure. L'autre avantage de cette approche (déjà acquis par Xu et al. (2020)) est que la DDCM ne nécessite pas de module tangent effectif. Notre méthodologie a donc pour objectif de réduire le nombre d'évaluations microscopiques : grâce à la formulation du problème *data-driven*, seuls les points pertinents pour la simulation macroéchelle sont calculés.

Le chapitre est organisé comme suit : le solveur *data-driven* FE^2 est présenté dans la Section 6.2, la méthode est évaluée au moyen d'exemples en deux dimensions présentés à la Section 6.3, les résultats sont discutés dans la conclusion 6.4.

6.2 Un solveur *data-driven* FE^2

Nous développons une méthode itérative qui consiste à conduire successivement des calculs DDCM macroéchelles avec une base de données enrichie entre chaque itération des points jugés nécessaires par des calculs microéchelles. Le processus général de la méthode est comparé à celui de la méthode FE^2 sur la Fig. E.5.

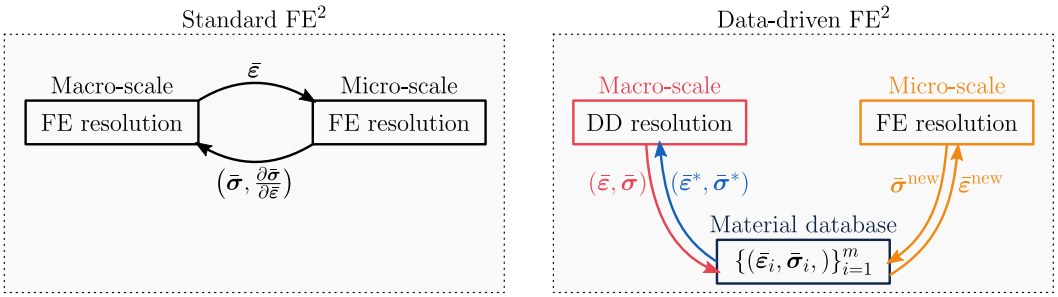


FIGURE E.5 – Schéma de fonctionnement de la méthode FE^2 standard et *data-driven*.

Dans un cette partie, nous rappelons dans un premier temps la structure et les ingrédients du solveur DDCM en petites déformations élastiques, appliqué au calcul macrostructural. Nous exposons ensuite notre méthodologie pour un calcul à deux échelles avec mise à jour adaptative de la base de données.

6.2.1 Solveur *data-driven* macroscopique

Nous rappelons ici rapidement la méthode DDCM présentée au Chapitre 2, qui consiste en une minimisation alternée des états mécaniques macroscopiques $(\bar{\boldsymbol{\varepsilon}}, \bar{\boldsymbol{\sigma}})$ (vérifiant la compatibilité et l'équilibre) et des états matériau $(\bar{\boldsymbol{\varepsilon}}^*, \bar{\boldsymbol{\sigma}}^*)$ (sélectionnés comme les plus proches dans la base de données). La base de données est alors constituée d'états déformation-contrainte macroscopiques :

$$\mathcal{D}_{\text{loc}} = \{(\bar{\boldsymbol{\varepsilon}}_i, \bar{\boldsymbol{\sigma}}_i) \mid i \in [1 \,..\, m]\}\,. \tag{E.63}$$

6.2.2 Solveur *data-driven* à deux échelles

Motivation Comme mentionné plus haut, l'approche de Xu et al. (2020) nécessite de déterminer *a priori* une base de données pour le calcul. Dans leurs travaux, Xu et al. (2020) utilisent une grille régulière pour échantillonner l'espace des déformations macroscopiques qui prescrivent les conditions de chargement de la microstructure. À notre connaissance, il n'existe pourtant pas de moyens pour déterminer efficacement les bornes et la densité nécessaire de cette grille. Par ailleurs, comme nous l'avons vu précédemment, une bonne partie des points risquent d'avoir été précalculés en vain car la réponse mécanique de la macrostructure ne couvre qu'une région réduite de l'espace de phases.

Idée générale et algorithme La méthode proposée est un processus itératif où, entre chaque itération, la base de données $\mathcal{D}_{\text{loc}}$ Eq. (E.63) est enrichie de façon adaptative. Pour choisir les points pertinents à lui ajouter, on utilise une des informations très importante donnée par la DDCM : la distance finale entre les états mécaniques et matériau. Si la distance $d_e\left((\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e), (\bar{\boldsymbol{\varepsilon}}^*_{\text{ie}}, \bar{\boldsymbol{\sigma}}^*_{\text{ie}})\right)$ est grande (supérieure à une distance critique d_c fixée par l'utilisateur), cela signifie qu'il manque des points dans la base de données dans la région de l'espace local des phases entourant l'état mécanique $(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)$ considéré. Ce type d'états mécaniques est appelé *état mécanique isolé*. On ajoute alors à la base de données l'état déformation-contrainte obtenu par un calcul microstructural dont les conditions aux limites sont déterminées par l'état mécanique $(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)$. Un exemple de cet procédure pour des conditions aux limites déterminées par la déformation macroscopique est donné sur la Fig E.6.

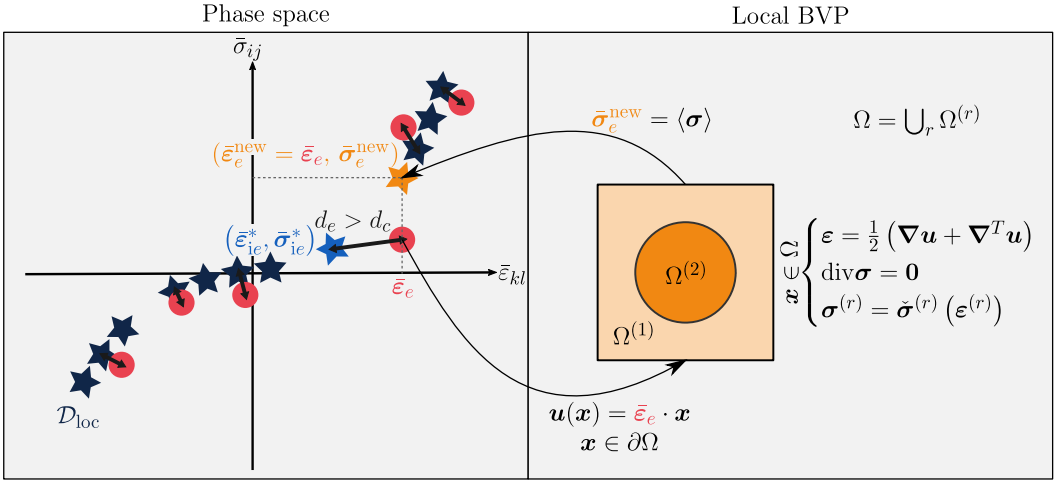


FIGURE E.6 – Stratégie de mise à jour de la base de données à partir des déformations. Le nouveau point matériau est obtenu à partir de la résolution d'un problème mécanique sur le VER de la microstructure, dont les conditions aux limites sont déterminées par la déformation mécanique $\bar{\boldsymbol{\varepsilon}}_e$, sélectionnée parmi les états mécaniques isolés (trop loin de leur état matériau associé).

L'algorithme de la méthode est fourni à l'Algorithme E.1.

Base de données matériau initiale Nous proposons d'initialiser la base de données $\mathcal{D}^{(0)}_{\text{loc}}$ le plus simplement possible et en ne faisant aucune hypothèse : la base de données initiale ne contient que le point de déformation-contrainte nulles : $\mathcal{D}^{(0)}_{\text{loc}} = \{(\boldsymbol{0}, \boldsymbol{0})\}$.

Dans ce paragraphe, nous discutons les conséquences de ce choix et montrons que cela revient à faire de la première itération du processus itératif Algorithme E.1 un estimateur linéaire élastique de raideur $\mathbb{C}$ de la solution du problème mécanique macroscopique à partir des conditions aux limites sur la macrostructure. La base de données $\mathcal{D}^{(1)}_{\text{loc}}$ est donc obtenue par l'homogénéisation de la réponse mécanique de la microstructure, à partir de la solution élastique linéaire en déformation sur

Algorithme E.1 Solveur *data-driven* à deux échelles.

ENTRÉES :

- Macroéchelle : modèle éléments finis de N nœuds et M points d'intégration, conditions aux limites et chargement $\left\{\bar{u}_\alpha^{\mathrm{D}}\right\}_{\alpha\in D}$ et $\left\{\bar{\boldsymbol{f}}_a\right\}_{a=1}^{N}$.
- Microéchelle : modèle éléments finis du VER de la microstructure, modèle de comportement des constituants.

SORTIES :

- Macroéchelle : états mécaniques $\{(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e)\}_{e=1}^{M}$, base de données matériau $\mathcal{D}_{\mathrm{loc}}$.
- Microéchelle : réponse mécanique locale et homogénéisée pour chaque point de chargement macroscopic dans $\mathcal{D}_{\mathrm{loc}}$.

Étape 1. Fixer $j = 0$. Initialiser la base de données $\mathcal{D}_{\mathrm{loc}}^{(0)}$.

Étape 2. Calculer les états mécaniques macroscopiques avec le solveur `DDSOLVER` :

$$\left(\bar{\boldsymbol{\varepsilon}}^{(j)}, \bar{\boldsymbol{\sigma}}^{(j)}\right) = \texttt{DDSOLVER}\left(\mathcal{D}_{\mathrm{loc}}^{(j)}\right).$$

Étape 3. Enrichir la base de données matériau d'après l'homogénéisation des états mécaniques isolés :

$$\mathcal{D}_{\mathrm{loc}}^{(j+1)} = \mathcal{D}_{\mathrm{loc}}^{(j)} \bigcup_{e\in I} \{(\bar{\boldsymbol{\varepsilon}}_e, \bar{\boldsymbol{\sigma}}_e^{\mathrm{new}})\},$$

avec $I = \{e \in [1\,..\,M] \mid d_e > d_c\}$ et $\bar{\boldsymbol{\sigma}}_e^{\mathrm{new}}$ obtenu d'après la stratégie présentée sur la Fig. E.6.

Étape 4. Tester la convergence :

- si $d_e \leq d_c$, $\forall e \in [1\,..\,M]$, alors fin.
- sinon, incrémenter $j \leftarrow j + 1$ et retourner à l'Étape 2.

la macrostructure. Ce premier remplissage de la base de données est donc plus pertinent qu'utiliser une grille régulière dans l'espace des déformations.

Paramètre $\mathbb{C}$ et modification de la stratégie de mise à jour Dans ce paragraphe, on discute de l'influence du paramètre $\mathbb{C}$ sur l'association état mécanique/état matériau. On montre qu'il est préférable d'avoir un grand (resp. petit) module de $\mathbb{C}$ pour une simulation pilotée en déplacements (resp. efforts), où la base de données est enrichie à partir des déformations (resp. contraintes) macroscopiques.

On modifie également légèrement le nombre de points à ajouter dans la base de données. Au lieu d'ajouter tous les états mécaniques isolés, on n'en sélectionne

qu'une partie, de telle sorte que le nombre de points ajoutés à la base de données ne dépassent pas une certaine limite M_{lim}. Typiquement on choisit $M_{\text{lim}} = M/2$ avec M le nombre de points d'intégration dans le maillage macroéchelle.

Le nombre d'évaluations microscopiques effectuées dans notre méthode de calcul multiéchelle est donc borné par

$$N_{\text{éval. micro}} \leq N_{\text{multi. iter.}} \times M_{\text{lim}}. \tag{E.64}$$

avec $N_{\text{multi. iter.}}$ le nombre d'itérations du processus multiéchelle de l'Algorithme E.1. Si ce nombre reste raisonnable, on peut donc espérer gagner beaucoup par rapport à l'évaluation systématique de la méthode FE^2 standard.

6.3 Résultats numériques

On étudie à nouveau la fine membrane rectangulaire trouée encastrée à une extrémité et soumise à un déplacement imposé de traction-cisaillement à l'autre extrémité. Dans un premier temps, on valide notre approche avec un matériau microscopiquement homogène : l'ajout de points à la base de données passe alors simplement par l'évaluation d'un modèle de comportement non linéaire. Dans un deuxième temps, on applique l'approche à un matériau composite biphasé, dont la microstructure est isotrope : il s'agit d'une inclusion circulaire rigide (élastique linéaire) dans une matrice plus souple (élastique non linéaire).

6.3.1 Cas test homogène

Dans cet exemple, on démontre à l'aide d'une comparaison entre les solutions *data-driven* et éléments finis de référence que la méthode produit des résultats très satisfaisants avec la base de données initialisée à zéro. Le nombre d'évaluations du modèle de comportement est également bien inférieur à celui de la méthode éléments finis classique.

6.3.2 Cas test sur un composite biphasé

Dans cette étude nous étudions deux maillages de la macrostructure (membrane) :

- Une première simulation multiéchelle est effectuée pour le maillage grossier ($M = 4060$ points d'intégration), en partant de la base de données $\mathcal{D}^{(0)}_{\text{loc}} = \{(\mathbf{0}, \mathbf{0})\}$;
- Une deuxième simulation multiéchelle identique est effectuée pour le maillage fin ($M = 9500$ point d'intégration), en partant de la base de données finale du précédent calcul : $\mathcal{D}^{(0)}_{\text{maillage fin}} = \mathcal{D}^{(j_{\max})}_{\text{maillage grossier}}$;

On mesure ainsi la pertinence de stocker les bases de données ainsi enrichies pour de futurs calculs similaires.

Dans une première analyse de la convergence des deux calculs multiéchelle, on montre que la fonction objectif des calculs DDCM successifs décroît très rapidement

à mesure que la base de données s'enrichit. Dans la première simulation 17 271 évaluations microscopiques ont été effectuées. Partant de ces calculs déjà effectués, 34 094 évaluations supplémentaires ont été nécessaires pour le calcul sur le maillage raffiné. En tout, ces 51 365 évaluations sont en nombre très nettement inférieur au besoin théorique d'une méthode FE^2 standard, comme le montre la Fig. E.7.

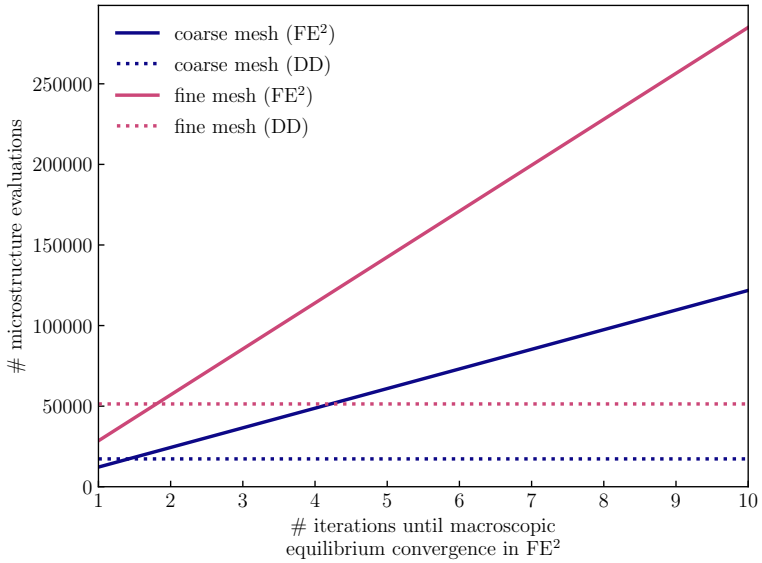


FIGURE E.7 – Nombre d'évaluations microscopiques effectuées par un calcul multiéchelle suivant la méthode FE^2 standard, en fonction du nombre d'itérations nécessaires pour atteindre l'équilibre macroscopique (schéma de Newton-Raphson par exemple).

On remarque que le nombre d'évaluations nécessaire au calcul sur le maillage raffiné est plus grand que pour le maillage grossier. Cela démontre que plus le maillage est fin, plus la base de données doit être raffinée. C'est notamment nécessaire car avec le raffinement du maillage, les zones de concentration des déformations et des contraintes dans la macrostructure agrandissent la région parcourue par la solution mécanique dans l'espace local des phases. On le démontre sur la Fig. E.8 où l'on reporte le nombre d'évaluations microscopiques effectuées en chaque point d'intégration du maillage fin (maximum de 13). Les zones au bord des trous nécessitent en effet plus d'évaluations.

6.4 Conclusion

Dans ce chapitre final de la thèse nous avons proposé une première stratégie de calcul adaptative pour les simulations multiéchelles, basée sur la DDCM. La base de données est alors constituée de couples déformation-contrainte macroscopiques. Typiquement, comme dans le schéma FE^2 classique, la contrainte macroscopique $\bar{\boldsymbol{\sigma}}$ est obtenue à partir de la moyenne volumique du champ microscopique $\boldsymbol{\sigma}$, solution

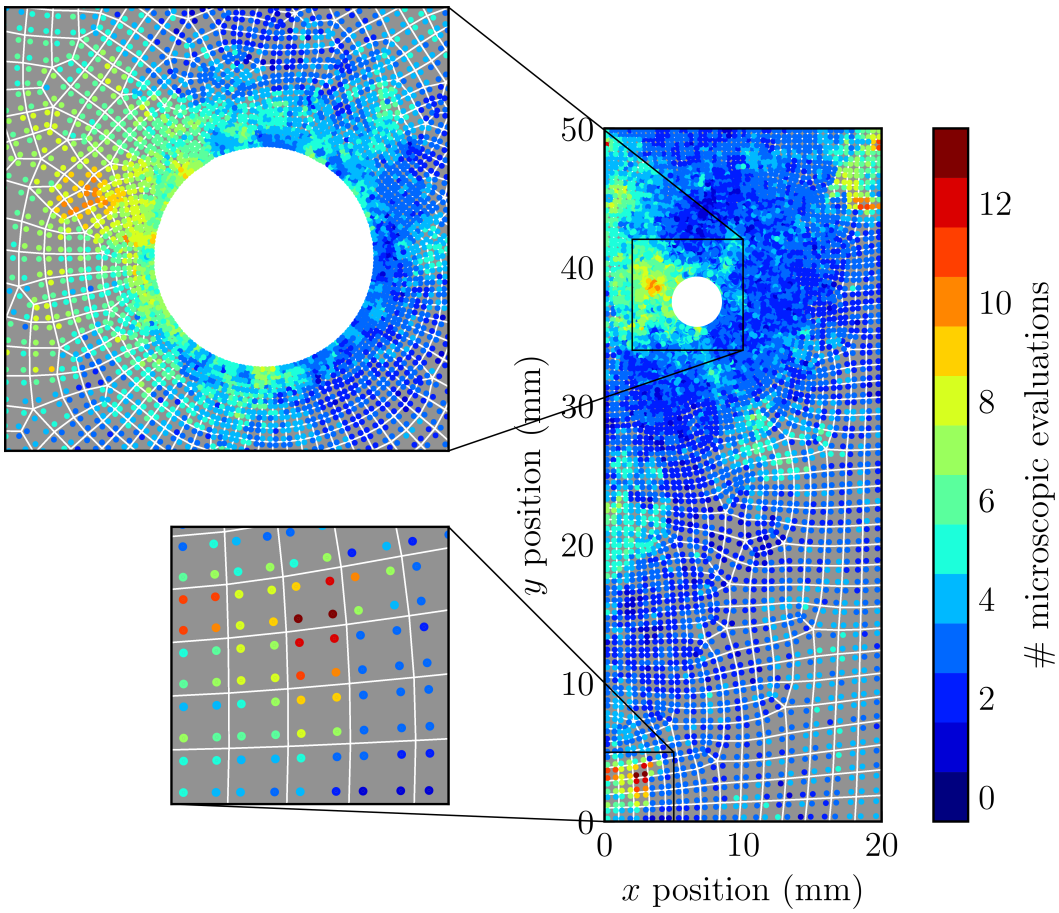


FIGURE E.8 – Nombre d'évaluations microscopiques en chaque point d'intégration du maillage macroscopique.

d'un problème microstructural dont les conditions aux limites sont déterminées par la déformation macroscopique $\bar{\boldsymbol{\varepsilon}}$ (Feyel, 1999). En échantillonnant l'espace des déformations macroscopiques par une grille régulière, Xu et al. (2020) ont proposé une première manière de réunir DDCM et homogénéisation numérique. Néanmoins, leur approche nécessite de déterminer la taille et la résolution de cette grille *a priori* et une bonne partie des points pré-calculés est susceptible d'être inutile pour la simulation macrostructurale considérée.

Notre méthodologie propose donc d'utiliser la DDCM comme un moyen de sélectionner les points à calculer. C'est un processus itératif dans lequel la base de données est successivement enrichie par les calculs microstructuraux correspondant seulement aux états mécaniques isolés : la distance entre les états mécaniques et matériau dans l'approche DDCM fournit en effet une information sur la qualité de la base de données actuelle. Tous les points dont la distance est jugée trop grande font l'objet d'un ajout à la base de données. Nous avons démontré que nous pouvons ainsi partir d'une base de données quasi-vide : l'unique point déformation-contrainte

nulles.

Nous avons d'abord validé notre approche sur un matériau microscopiquement homogène. Nous l'avons ensuite appliquée à un matériau hétérogène : un composite isotrope biphasé non linéaire. Notre méthodologie réduit drastiquement le nombre d'évaluations microscopiques nécessaires, par rapport à une méthode FE^2 standard. De plus, la base de données enrichie par un premier calcul peut être efficacement réutilisée comme point de départ d'un second calcul adaptatif. Les réponses microscopique et macroscopique se sont alors révélées qualitativement satisfaisantes.

Pour conclure, les résultats de cette première étude pour une approche de calcul adaptatif utilisant la DDCM et l'homogénéisation numérique sont très encourageants. De nombreuses questions restent ouvertes et les paramètres de la méthode peuvent certainement être améliorés (par exemple le paramètre $\mathbb{C}$ ou le nombre de point à ajouter). Néanmoins, nous pensons que ce type d'approche offre une application très intéressante du paradigme DDCM. En perspective, le même raisonnement peut être appliqué pour développer une méthode analogue en grandes transformations élastiques et ainsi offrir des solutions pour l'homogénéisation des matériaux hétérogènes non linéaires. Enfin, il est aussi possible d'imaginer une approche multiéchelle entièrement *data-driven* : les deux calculs (macro- et microéchelle) seraient effectués en utilisant la DDCM, au lieu de la méthode des éléments finis standard comme ce fut le cas ici pour le calcul microéchelle.

References

Avril, S., M. Bonnet, A.-S. Bretelle, M. Grédiac, F. Hild, P. Ienny, F. Latourte, D. Lemosse, S. Pagano, E. Pagnacco, & F. Pierron (2008). Overview of identification methods of mechanical parameters based on full-field measurements. *Experimental Mechanics 48*(4), 381–402.
cited on page 32.

Ayensa-Jiménez, J., M. H. Doweidar, J. A. Sanz-Herrera, & M. Doblaré (2018). A new reliability-based data-driven approach for noisy experimental data with physical constraints. *Computer Methods in Applied Mechanics and Engineering 328*, 752–774.
cited on pages 58 and 281.

Ayensa-Jiménez, J., M. H. Doweidar, J. A. Sanz-Herrera, & M. Doblaré (2019). An unsupervised data completion method for physically-based data-driven models. *Computer Methods in Applied Mechanics and Engineering 344*, 120–143.
cited on page 203.

Badel, P.-B. & J.-B. Leblond (2004). A note on integration schemes for the microplane model of the mechanical behaviour of concrete. *Communications in Numerical Methods in Engineering 20*(1), 75–81.
cited on page 137.

Bažant, P. & B. H. Oh (1986). Efficient Numerical Integration on the Surface of a Sphere. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik 66*(1), 37–49.
cited on page 137.

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM 18*(9), 509–517.
cited on pages 80 and 289.

Bertsekas, D. P. (1996). *Constrained Optimization and Lagrange Multiplier Methods*. Belmont, Massachusetts: Athena Scientific.
cited on pages 93, 96, 100, 101 and 292.

Bonet, J., A. J. Gil, & R. D. Wood (2016). *Nonlinear Solid Mechanics for Finite Element Analysis: Statics*. Cambridge: Cambridge University Press.
cited on pages 177, 180, 253 and 308.

Bonet, J. & R. D. Wood (2008). *Nonlinear Continuum Mechanics for Finite Element Analysis* (2nd edition ed.). Cambridge, UK ; New York: Cambridge University Press.
cited on pages 10, 12, 14, 18, 19, 20, 22, 23, 24, 61 and 272.

Bornert, M., T. Bretheau, & P. Gilormini (2001). *Homogénéisation en mécanique des matériaux, Tome 1 : Matériaux aléatoires élastiques et milieux périodiques*. Hermes science.
cited on pages 199, 201 and 224.

Boyce, M. C. & E. M. Arruda (2000). Constitutive models of rubber elasticity: A review. *Rubber Chemistry and Technology 73*(3), 504–523.
cited on page 30.

Brannon, R. M. (2018). *Rotation, Reflection, and Frame Changes*. Orthogonal Tensors in Computational Engineering Mechanics. IOP Publishing.
cited on pages 80, 149 and 251.

Brassart, L. (2011). *Homogenization of elasto-(visco)plastic composites: history-dependent incremental and variational approaches*. Phd thesis, Université Catholique de Louvain.
cited on pages 200 and 316.

Buffiere, J. Y., E. Ferrie, H. Proudhon, & W. Ludwig (2006). Three-dimensional visualisation of fatigue cracks in metals using high resolution synchrotron X-ray micro-tomography. *Materials Science and Technology 22*(9), 1019–1024.
cited on page 34.

Carrara, P., L. D. Lorenzis, L. Stainier, & M. Ortiz (2020). Data-driven fracture mechanics. *arXiv:2006.03133v1 [cs.CE]*, 39.
cited on pages 60 and 282.

Chaboche, J. L. & G. Cailletaud (1996). Integration methods for complex plastic constitutive equations. *Computer Methods in Applied Mechanics and Engineering 133*(1), 125–155.
cited on page 30.

Coleman, B. D. & W. Noll (1974). The Thermodynamics of Elastic Materials with Heat Conduction and Viscosity. In W. Noll (Ed.), *The Foundations of Mechanics and Thermodynamics: Selected Papers*, pp. 145–156. Berlin, Heidelberg: Springer.
cited on page 29.

Conti, S., S. Müller, & M. Ortiz (2018). Data-driven problems in elasticity. *Archive for Rational Mechanics and Analysis 229*(1), 79–123.
cited on pages 3, 4, 37, 38, 39, 40, 42, 43, 44, 45, 48, 57, 69, 84, 103, 235, 277, 278, 280 and 290.

Conti, S., S. Müller, & M. Ortiz (2020). Data-driven finite elasticity. *Archive for Rational Mechanics and Analysis 237*(1), 1–33.
cited on pages 60, 61, 69, 84, 86, 88, 89, 103, 111, 123, 235, 236, 237, 246, 249, 255, 265, 266, 267, 282, 283, 284, 290, 293 and 296.

Crespo, J., M. Latorre, & F. J. Montáns (2017). WYPIWYG hyperelasticity for isotropic, compressible materials. *Computational Mechanics 59*(1), 73–92.
cited on page 34.

Criscione, J. C., J. D. Humphrey, A. S. Douglas, & W. C. Hunter (2000). An invariant basis for natural strain which yields orthogonal stress response terms in isotropic hyperelasticity. *Journal of the Mechanics and Physics of Solids 48*(12), 2445–2465.
cited on pages 131, 138, 139, 141, 142, 173, 191, 300, 301, 302 and 306.

Dalémat, M. (2019). *Une expérimentation réussie pour l'identification de la réponse mécanique sans loi de comportement: approche data-driven appliquée aux membranes élastomères*. Phd thesis, Ecole centrale de Nantes.
cited on pages 60, 127, 128, 213, 282, 297, 305 and 306.

Dalémat, M., M. Coret, A. Leygue, & E. Verron (2019). Measuring stress field without constitutive equation. *Mechanics of Materials 136*, 103087.
cited on pages 60, 128, 130, 168, 238, 282 and 305.

Dhar, V. (2013). Data science and prediction. *Communications of the ACM 56*(12), 64–73.
cited on pages 33 and 275.

Doll, S. & K. Schweizerhof (2000). On the development of volumetric strain energy functions. *Journal of Applied Mechanics 67*(1), 17–21.
cited on page 264.

Eggersmann, R., T. Kirchdoerfer, S. Reese, L. Stainier, & M. Ortiz (2019). Model-Free Data-Driven inelasticity. *Computer Methods in Applied Mechanics and Engineering 350*, 81–99.
cited on pages 60, 61 and 282.

Eggersmann, R., L. Stainier, M. Ortiz, & S. Reese (2020). Model-free data-driven computational mechanics enhanced by tensor voting. *arXiv:2004.02503 [cs]*.
cited on pages 58, 129, 281 and 298.

Eshelby, J. D. & R. E. Peierls (1957). The determination of the elastic field of an ellipsoidal inclusion, and related problems. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences 241*(1226), 376–396.
cited on pages 31 and 201.

Feyel, F. (1999). Multiscale FE2 elastoviscoplastic analysis of composite structures. *Computational Materials Science 16*(1), 344–354.
cited on pages 5, 31, 202, 231, 316 and 323.

Feyel, F. & J.-L. Chaboche (2000). FE2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials. *Computer*

Methods in Applied Mechanics and Engineering 183(3), 309–330.
cited on page 31.

Fritzen, F. & O. Kunc (2018a). GitHub repository MinimumEnergyPoints. https://github.com/EMMA-Group/MinimumEnergyPoints.
cited on page 179.

Fritzen, F. & O. Kunc (2018b). Two-stage data-driven homogenization for nonlinear solids using a reduced order model. *European Journal of Mechanics - A/Solids 69*, 201–220.
cited on page 138.

Gebhardt, C. G., D. Schillinger, M. C. Steinbach, & R. Rolfes (2020). A framework for data-driven structural analysis in general elasticity based on nonlinear optimization: The static case. *Computer Methods in Applied Mechanics and Engineering 365*, 112993.
cited on pages 58 and 281.

Geers, M. G. D., V. G. Kouznetsova, & W. A. M. Brekelmans (2010). Multi-scale computational homogenization: Trends and challenges. *Journal of Computational and Applied Mathematics 234*(7), 2175–2182.
cited on pages 31 and 199.

Germain, P., P. Suquet, & Q. S. Nguyen (1983). Continuum thermodynamics. *ASME Journal of Applied Mechanics 50*, 1010–1020.
cited on page 29.

Geuzaine, C. & J.-F. Remacle (2009). Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering 79*(11), 1309–1331.
cited on page 151.

Ghaboussi J., Garrett J. H., & Wu X. (1991). Knowledge-based modeling of material behavior with neural networks. *Journal of Engineering Mechanics 117*(1), 132–153.
cited on page 33.

Halphen, B. & Q. Son Nguyen (1975). Sur les matériaux standard généralisés. *Journal de Mécanique 14*, 39–63.
cited on page 29.

Hashash, Y. M. A., S. Jung, & J. Ghaboussi (2004). Numerical implementation of a neural network based material model in finite element analysis. *International Journal for Numerical Methods in Engineering 59*(7), 989–1005.
cited on page 33.

He, Q. & J.-S. Chen (2020). A physics-constrained data-driven approach based on locally convex reconstruction for noisy database. *Computer Methods in Applied*

Mechanics and Engineering 363, 112791.
cited on pages 58 and 281.

Herbig, M., A. King, P. Reischig, H. Proudhon, E. M. Lauridsen, J. Marrow, J.-Y. Buffière, & W. Ludwig (2011). 3-D growth of a short fatigue crack within a polycrystalline microstructure studied using combined diffraction and phase-contrast X-ray tomography. *Acta Materialia 59*(2), 590–601.
cited on page 34.

Hesse, K., I. H. Sloan, & R. S. Womersley (2010). Numerical integration on the sphere. In *Handbook of Geomathematics*, pp. 1185–1219. Springer Berlin Heidelberg.
cited on pages 136 and 138.

Hill, R. (1963). Elastic properties of reinforced solids: Some theoretical principles. *Journal of the Mechanics and Physics of Solids 11*(5), 357–372.
cited on page 199.

Hill, R. (1965). Continuum micro-mechanics of elastoplastic polycrystals. *Journal of the Mechanics and Physics of Solids 13*(2), 89–101.
cited on pages 31 and 201.

Hill, R. (1967). The essential structure of constitutive laws for metal composites and polycrystals. *Journal of the Mechanics and Physics of Solids 15*(2), 79–95.
cited on page 201.

Holzapfel, G. A. (2000). *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. John Wiley & Sons.
cited on pages 18, 29 and 274.

Ibañez, R., E. Abisset-Chavanne, J. V. Aguado, D. Gonzalez, E. Cueto, & F. Chinesta (2018). A manifold learning approach to data-driven computational elasticity and inelasticity. *Archives of Computational Methods in Engineering 25*(1), 47–57.
cited on page 34.

Ibañez, R., D. Borzacchiello, J. V. Aguado, E. Abisset-Chavanne, E. Cueto, P. Ladeveze, & F. Chinesta (2017). Data-driven non-linear elasticity: Constitutive manifold construction and problem discretization. *Computational Mechanics 60*(5), 813–826.
cited on page 34.

Itskov, M. (2000). On the theory of fourth-order tensors and their applications in computational mechanics. *Computer Methods in Applied Mechanics and Engineering 189*(2), 419–438.
cited on page 19.

Itskov, M. (2015). *Tensor Algebra and Tensor Analysis for Engineers: With Applications to Continuum Mechanics* (4th ed ed.). Mathematical Engineering. Cham: Springer.
cited on page 19.

Kanno, Y. (2018). Data-driven computing in elasticity via kernel regression. *Theoretical and Applied Mechanics Letters 8*(6), 361–365.
cited on pages 57 and 281.

Kanno, Y. (2019). Mixed-integer programming formulation of a data-driven solver in computational elasticity. *Optimization Letters 13*(7), 1505–1514.
cited on pages 58, 78, 238 and 281.

Kanno, Y. (2020). A kernel method for learning constitutive relation in data-driven computational elasticity. *Japan Journal of Industrial and Applied Mathematics*.
cited on pages 58 and 281.

Kanouté, P., D. P. Boso, J. L. Chaboche, & B. A. Schrefler (2009). Multiscale methods for composites: A review. *Archives of Computational Methods in Engineering 16*(1), 31–75.
cited on page 199.

Kirchdoerfer, T. & M. Ortiz (2016). Data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering 304*, 81–101.
cited on pages 3, 5, 34, 37, 38, 41, 42, 45, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 60, 61, 63, 70, 80, 82, 123, 124, 127, 129, 194, 199, 204, 205, 235, 236, 242, 249, 276, 277, 279, 280, 296, 298, 313 and 315.

Kirchdoerfer, T. & M. Ortiz (2017). Data-driven computing with noisy material data sets. *Computer Methods in Applied Mechanics and Engineering 326*, 622–641.
cited on pages 42, 57, 78, 238, 242 and 281.

Kirchdoerfer, T. & M. Ortiz (2018). Data-driven computing in dynamics. *International Journal for Numerical Methods in Engineering 113*(11), 1697–1710.
cited on pages 60 and 282.

Klusemann, B. & M. Ortiz (2015). Acceleration of material-dominated calculations via phase-space simplicial subdivision and interpolation. *International Journal for Numerical Methods in Engineering 103*(4), 256–274.
cited on pages 202 and 317.

Korzeniowski, T. F. & K. Weinberg (2019). A comparison of stochastic and data-driven FEM approaches to problems with insufficient material data. *Computer Methods in Applied Mechanics and Engineering 350*, 554–570.
cited on pages 58 and 281.

Kunc, O. & F. Fritzen (2019a). Finite strain homogenization using a reduced basis and efficient sampling. *Mathematical and Computational Applications 24*(2), 56.
cited on pages 129, 130, 131, 132, 133, 134, 136, 138, 140, 141, 173, 178, 179, 299, 300, 301, 302, 306, 309 and 313.

Kunc, O. & F. Fritzen (2019b). Generation of energy-minimizing point sets on spheres and their application in mesh-free interpolation and differentiation. *Advances in Computational Mathematics 45*(5), 3021–3056.
cited on page 179.

Ladevèze, P. (1996). *Mécanique Non Linéaire Des Structures, Nouvelle Approche et Méthodes de Calcul Non Incrémentales*. Étude En Mécanique Des Matériaux et Des Structures. Paris: Hermès.
cited on page 30.

Latorre, M. & F. J. Montáns (2014). What-You-Prescribe-Is-What-You-Get orthotropic hyperelasticity. *Computational Mechanics 53*(6), 1279–1298.
cited on page 34.

Le, B. A., J. Yvonnet, & Q.-C. He (2015). Computational homogenization of nonlinear elastic materials using neural networks. *International Journal for Numerical Methods in Engineering 104*(12), 1061–1084.
cited on page 33.

Leygue, A., M. Coret, J. Réthoré, L. Stainier, & E. Verron (2018). Data-based derivation of material response. *Computer Methods in Applied Mechanics and Engineering 331*, 184–196.
cited on pages 59, 60, 128, 130, 168, 238, 281, 297, 305 and 306.

Leygue, A., R. Seghir, J. Réthoré, M. Coret, E. Verron, & L. Stainier (2019). Non-parametric material state field extraction from full field measurements. *Computational Mechanics 64*(2), 501–509.
cited on pages 60 and 282.

Ling, J., R. Jones, & J. Templeton (2016). Machine learning strategies for systems with invariance properties. *Journal of Computational Physics 318*, 22–35.
cited on page 33.

Marckmann, G. & E. Verron (2006). Comparison of hyperelastic models for rubber-like materials. *Rubber Chemistry and Technology 79*(5), 835–858.
cited on page 30.

Marimont, R. B. & M. B. Shapiro (1979). Nearest neighbour searches and the curse of dimensionality. *IMA Journal of Applied Mathematics 24*(1), 59–70.
cited on page 178.

Matouš, K., M. G. Geers, V. G. Kouznetsova, & A. Gillman (2017). A review of predictive nonlinear theories for multiscale modeling of heterogeneous materials.

Journal of Computational Physics 330, 192–220.
cited on pages 32, 33, 199, 202, 203, 224 and 317.

Mihai, L. A. & A. Goriely (2017). How to characterize a nonlinear elastic material? A review on nonlinear constitutive parameters in isotropic finite elasticity. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 473*(2207), 20170607.
cited on page 30.

Mora-Macías, J., J. Ayensa-Jiménez, E. Reina-Romo, M. H. Doweidar, J. Domínguez, M. Doblaré, & J. A. Sanz-Herrera (2020). A multiscale data-driven approach for bone tissue biomechanics. *Computer Methods in Applied Mechanics and Engineering 368*, 113136.
cited on page 203.

Mosby, M. & K. Matouš (2016). Computational homogenization at extreme scales. *Extreme Mechanics Letters 6*, 68–74.
cited on page 32.

Moulinec, H. & P. Suquet (1998). A numerical method for computing the overall response of nonlinear composites with complex microstructure. *Computer Methods in Applied Mechanics and Engineering 157*(1), 69–94.
cited on page 201.

Nguyen, L. T. K. & M.-A. Keip (2018). A data-driven approach to nonlinear elasticity. *Computers & Structures 194*, 97–115.
cited on pages 4, 60, 61, 69, 71, 80, 82, 84, 123, 235, 236, 245, 249, 282, 283, 284, 285, 289 and 296.

Nguyen, L. T. K., M. Rambausek, & M.-A. Keip (2020). Variational framework for distance-minimizing method in data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering 365*, 112898.
cited on pages 38, 60, 81, 129, 277, 282 and 298.

Ogden, R. W. (1972). Large deformation isotropic elasticity – on the correlation of theory and experiment for incompressible rubberlike solids. *Proc. R. Soc. Lond. A. Mathematical and Physical Sciences. 326*, 565–584.
cited on page 145.

Ortiz, M. (2015). "Data-Driven Computational Mechanics" in XIII International Conference on Computational Plasticity. Fundamentals and Applications. Barcelona, Spain. http://cimnemultimediachannel.com/vnews/2790/michael-ortiz-data-driven-computational-mechanics-.
cited on page 32.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos,

D. Cournapeau, M. Brucher, M. Perrot, & É. Duchesnay (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research 12*, 2825–2830.
cited on pages 80, 238, 251 and 289.

Platzer, A., A. Leygue, & L. Stainier (2019). Assessment of data-driven computational mechanics in finite strain elasticity. In *Constitutive Models for Rubber XI: Proceedings of the 11th European Conference on Constitutive Models for Rubber (ECCMR 2019), June 25-27, 2019, Nantes, France*, pp. 230. CRC Press.
cited on page 82.

Réthoré, J., A. Leygue, M. Coret, L. Stainier, & E. Verron (2018). Computational measurements of stress fields from digital images. *International Journal for Numerical Methods in Engineering 113*(12), 1810–1826.
cited on pages 60 and 282.

Rovinelli, A., M. D. Sangid, H. Proudhon, & W. Ludwig (2018). Using machine learning and a data-driven approach to identify the small fatigue crack driving force in polycrystalline materials. *npj Computational Materials 4*(1), 35.
cited on page 34.

Roweis, S. T. & L. K. Saul (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science 290*(5500), 2323–2326.
cited on page 33.

Simo, J. C. & T. J. R. Hughes (1998). *Computational Inelasticity* (First ed.). Number 7 in Interdisciplinary Applied Mathematics. New York: Springer-Verlag.
cited on page 30.

Sloan, I. H. & R. S. Womersley (2004). Extremal Systems of Points and Numerical Integration on the Sphere. *Advances in Computational Mathematics 21*(1), 107–125.
cited on pages 137, 138, 139, 182 and 301.

Stainier, L., A. Leygue, & M. Ortiz (2019). Model-free data-driven methods in mechanics: Material data identification and solvers. *Computational Mechanics 64*(2), 381–393.
cited on pages 60, 128, 129, 145, 282 and 303.

Steinmann, P., M. Hossain, & G. Possart (2012). Hyperelastic models for rubber-like materials: Consistent tangent operators and suitability for Treloar's data. *Archive of Applied Mechanics 82*(9), 1183–1217.
cited on page 30.

Terada, K., T. Miura, & N. Kikuchi (1997). Digital image-based modeling applied to the homogenization analysis of composite materials. *Computational Mechanics 20*(4), 331–346.
cited on page 201.

Treloar, L. R. G. (1944). Stress-strain data for vulcanised rubber under various types of deformation. *Transactions of the Faraday Society 40*(0), 59–70.
cited on page 145.

Verron, E. (2015). Questioning numerical integration methods for microsphere (and microplane) constitutive equations. *Mechanics of Materials 89*, 216–228.
cited on pages 136 and 137.

Verron, E. (2018). Modèles hyperélastiques pour le comportement mécanique des élastomères. *Ref : TIP100WEB - "Plastiques et composites"*.
cited on page 30.

Womersley, R. S. (2007). Extremal points on the sphere. https://web.maths.unsw.edu.au/~rsw/Sphere/Extremal/New/extremal1.html.
cited on pages 138 and 139.

Xu, R., J. Yang, W. Yan, Q. Huang, G. Giunta, S. Belouettar, H. Zahrouni, T. B. Zineb, & H. Hu (2020). Data-driven multiscale finite element method: From concurrence to separation. *Computer Methods in Applied Mechanics and Engineering 363*, 112893.
cited on pages 203, 206, 227, 231, 317, 318 and 323.

Yeoh, O. H. (1993). Some forms of the strain energy function for rubber. *Rubber Chemistry and Technology 66*(5), 754–771.
cited on page 258.

Yvonnet, J. & Q. C. He (2007). The reduced model multiscale method (R3M) for the non-linear homogenization of hyperelastic media at finite strains. *Journal of Computational Physics 223*(1), 341–368.
cited on page 33.

Zakon, E. (2017). *Mathematical Analysis. Volume I.* West Lafayette, Indiana, USA: The Trillia Group.
cited on page 73.

Zaoui, A. (2002). Continuum micromechanics: Survey. *Journal of Engineering Mechanics 128*(8), 808–816.
cited on pages 199 and 201.

Zohdi, T. I. & P. Wriggers (2008). *An Introduction to Computational Micromechanics.* Springer, Berlin, Heidelberg.
cited on page 199.

CENTRALE NANTES

Titre : Mécanique numérique en grandes transformations pilotée par les données. De la génération de données sur mesure à une stratégie adaptative de calcul multiéchelle.

Mots clés : Méthodes data-driven ; Grandes déformations ; Hyperélasticité ; Multiéchelle.

Résumé : La mécanique numérique est aujourd'hui au cœur d'un important flux de données. D'un côté, l'identification des lois de comportement utilisées dans les simulations éléments finis repose sur de riches données expérimentales (mesures de champs). D'un autre côté, les calculs multiéchelles fournissent un très grand nombre de valeurs discrètes de champs de déplacement, déformation et contrainte, dont on extrait des connaissances sur la réponse effective du matériau. Entre ces données, la loi de comportement apparaît comme un goulot contraignant le champ des possibles.

En rupture avec cette approche, Kirchdoerfer et Ortiz (*Computer Methods in Applied Mechanics and Engineering*, 304, 81-101) ont proposé un paradigme de mécanique numérique sans modèle, appelé *data-driven computational mechanics*. La réponse matériau y est uniquement représentée par une base de données (couples déformation-contrainte). Le problème mécanique est alors reformulé comme une minimisation sous contrainte de la distance entre (i) l'état déformation-contrainte mécanique de la structure, et (ii) la base de données matériau.

Ces travaux de thèse se concentrent sur la question de la couverture de l'espace par les données matériau, notamment dans le cadre des grandes transformations. Ainsi, l'approche *data-driven* est d'abord étendue à la mécanique non linéaire : nous considérons deux formulations différentes et proposons pour chacune d'elles un solveur éléments finis. Nous explorons ensuite la génération de base de données sur mesure, grâce à une méthode d'échantillonnage mécaniquement motivée. Nous évaluons l'approche au moyen d'analyses éléments finis de structures complexes en grandes déformations. Enfin, nous proposons une première stratégie de calcul multiéchelle pilotée par les données, qui permet d'enrichir de façon adaptative la base de données matériau.

Title : Finite strain data-driven computational mechanics. From tailored data to adaptive solvers for multiscale simulations.

Keywords : Data-driven computing; Finite strain; Hyperelasticity; Multiscale.

Abstract : Computational mechanics is a field in which a large amount of data is both consumed and produced. On the one hand, the recent developments of experimental measurement techniques have provided rich data for the identification process of constitutive models used in finite element simulations. On the other hand, multiscale analysis produces a huge amount of discrete values of displacements, strains and stresses from which knowledge is extracted on the overall material behavior. The constitutive model then acts as a bottleneck between upstream and downstream material data.

In contrast, Kirchdoerfer and Ortiz (*Computer Methods in Applied Mechanics and Engineering*, 304, 81-101) proposed a model-free computing paradigm, called data-driven computational mechanics. The material response is then only represented by a database of raw material data (strain-stress pairs). The boundary value problem is thus reformulated as a constrained distance minimization between (i) the mechanical strain-stress state of the body, and (ii) the material database.

In this thesis, we investigate the question of material data coverage, especially in the finite strain framework. The data-driven approach is first extended to a geometrically nonlinear setting: two alternative formulations are considered and a finite element solver is proposed for both. Second, we explore the generation of tailored databases using a mechanically meaningful sampling method. The approach is assessed by means of finite element analyses of complex structures exhibiting large deformations. Finally, we propose a prototype multiscale data-driven solver, in which the material database is adaptively enriched.