Graph-based representation of history-dependent material response in the Data-Driven Computational Mechanics framework

Héloïse Dandin, Adrien Leygue, Laurent Stainier

Graph-based representation of history-dependent material response in the Data-Driven Computational Mechanics framework

Héloïse Dandin, Adrien Leygue, Laurent Stainier

Nantes Université, Ecole Centrale Nantes, CNRS, GeM, UMR 6183, F-44000 Nantes

Abstract

This work develops an extension of the Data-Driven Computational Mechanics approach introduced by Kirchdoerfer and Ortiz (2016) to history-dependent behaviour. The material database is replaced by a material directed graph, where vertices represent strain-stress pairs and arcs encode thermomechanically admissible transitions between them. A graph search algorithm is used to select local material databases representative of the incremental material behaviour. This study thus gives the mathematical tools for a structured data-driven representation of the material behaviour, designed to minimise the increase of the problem dimensionality. The performances of the proposed approach applied to elastoplasticity are studied on a single-element structure and a larger truss structure. A two-step solver is proposed to address numerical accuracy issues in the latter case.

Keywords: Data-Driven Computational Mechanics, computational plasticity, graph theory

Contents

1 Introduction **1**

2 Data-Driven Computational Mechanics framework for inelastic material response **2**

3 Directed graphs for DDCM simulations of history-dependent materials **4**

3.1 Representation of the discrete material behaviour with a material digraph 5

3.2 Distinction between identical states in the constitutive space 6

3.3 Local database selection 7

3.4 Local database reduction 8

3.5 Numerical representation of the material digraph 9

4 Numerical implementation and investigation of a single element problem **10**

4.1 Material digraph construction 10

4.2 Evaluation of results quality 13

4.3 Resolution of a data-driven problem 13

5 Numerical experiment on a truss problem **15**

5.1 Truss structure and related challenges 15

5.2 Predictor-corrector algorithm 18

6 Conclusion **20**

Bibliography **22**

1 Introduction

Model-free Data-Driven Computational Mechanics (DDCM), first introduced by Kirchdoerfer and Ortiz [2016], is a method for solving boundary value problems in solid mechanics without constitutive models. In this approach the material behaviour is represented by a discrete set of points, instead of a set of explicit mathematical expressions as in classical constitutive modelling (see, *e.g.*, Lubliner [2008]) or machine learning methods (see, *e.g.*, Ghavamian and Simone [2019], Flaschel et al. [2022]). In DDCM, the solution is then formulated as a distance minimisation problem, which becomes a mixed-integer quadratic problem (MIQP) exhibiting combinatorial complexity [Kanno, 2019].

The data sets that describe the material behaviour, also referred to as material databases, can be obtained in different ways. On the one hand, multiscale modelling has been used to generate data with constitutive models identified at the microscopic scale [Karapiperis et al., 2021, Korzeniowski and Weinberg, 2021, Platzer, 2020]. On the other hand, identification methods such as Data-Driven Identification (DDI) allow estimating stresses from experimental strain measures without constitutive models [Leygue et al., 2019, Dalémat et al., 2019, Vinel, 2022]. In that case, DDI can be seen as an inverse problem of DDCM for material properties identification.

These material databases are likely to evolve during an incremental simulation, for example to locally enrich a sparse database [Platzer, 2020] or due to the irreversibility of the behaviour [Eggersmann et al., 2019]. In the latter case, the solver must, at each point of the structure and for each loading increment, monitor irreversibility and propose a set of admissible data points. Eggersmann et al. [2019] review typical representational paradigms for history-dependent materials in continuum mechanics: general materials with memory, differential representation, internal and history variables. All these methods, especially the first one, imply to increase significantly the dimensionality of the problem. For instance, the work by Valdés-Alonzo [2022] develops an extension of DDI to dissipative behaviours within the framework of materials with memory, while Pham et al. [2023], following Ciftci and Hackl [2022], and Langlois et al. [2022] use differential representations. Poelstra et al. [2022] and Bartel et al. [2023] adopted a different strategy based on history surrogates that also requires a consequent dimensionality augmentation.

In this work, we develop a graph-based representation of history-dependent material databases. Graph theory is a branch of discrete mathematics interested in networks and pairwise relations between objects, that has already been applied in solid mechanics, for example to granular materials [Goddard, 2008] and dislocation kinematics [Starkey et al., 2022]. It offers a wide range of tools and methods designed to handle large amounts of data and solve optimisation problems. We show that the history dependence can be encoded in a *material directed graph* (*abbr.* digraph). This is first achieved by expanding the usual strain-stress constitutive space to other thermodynamical quantities like internal variables, and second by

making use of graph arcs to record dissipation levels. Standard efficient graph algorithms can then be used to extract the material database from the material digraph and perform the simulation in strain-stress space.

To illustrate the concepts, we present numerical results obtained for one-dimensional elements, in which case all quantities of interest are scalar. We first put aside the combinatorial issues arising from the alternating minimisation solver by modelling the behaviour of a single element to demonstrate the ability of the method to generate a material database from the material digraph allowing us to solve a simple problem. For larger structures, we show the accuracy issues and challenges coming both from the combinatorial complexity of DDCM and the richer material behaviour. An enhanced solver that takes advantage of the thermodynamical information encoded in the material digraph is proposed. We show that it is possible to retrieve a good response from the system both at global and local scales.

2 Data-Driven Computational Mechanics framework for inelastic material response

This section recalls the mathematical framework for model-free data-driven inelasticity introduced by Kirchdoerfer and Ortiz [2016] and Eggersmann et al. [2019].

We consider the discrete representation of an inelastic body composed of N nodes and M integration points, which undergoes displacements $\boldsymbol{u} = \{\boldsymbol{u}_a\}_{a=1}^{N}$ and loads $\boldsymbol{f} = \{\boldsymbol{f}_a\}_{a=1}^{N}$. In a time-discrete setting, we seek to approximate solutions at times $\{t_0, \dots, t_k, t_{k+1}, \dots\}$.

The internal state of the system can be described at each time step by strain and stress pairs $\{(\boldsymbol{\varepsilon}_{e,k+1}, \boldsymbol{\sigma}_{e,k+1})\}_{e=1}^{M}$, with $\boldsymbol{\varepsilon}_{e,k+1}, \boldsymbol{\sigma}_{e,k+1} \in \mathbb{R}^{m_e}$ and m_e the dimension of the tensors at integration point e for linearised kinematics. In a three-dimensional setting, $m_e = 6$ in Voigt notation, while $m_e = 1$ for bar elements. A local state is thus defined as $z_{e,k+1} = (\boldsymbol{\varepsilon}_{e,k+1}, \boldsymbol{\sigma}_{e,k+1}) \in \mathcal{Z}_{e,k+1}$ with $\mathcal{Z}_{e,k+1} = \mathbb{R}^{m_e \times m_e}$ the local phase space[1], while $z_{k+1} = \{z_e\}_{e=1}^{M}$ is a point in the global phase space $\mathcal{Z}_{k+1} = \bigotimes_{e=1}^{M} \mathcal{Z}_{e,k+1}$, with $\otimes$ meaning the tensor product of spaces.

The mechanical admissibility of a state is material-independent and is given by the compatibility and equilibrium constraints at time step t_{k+1}:

$$\boldsymbol{\varepsilon}_{e,k+1} = \boldsymbol{B}_e \boldsymbol{u}_{k+1}, \forall e = 1 \dots M, \tag{1a}$$

$$\sum_{e=1}^{M} w_e \boldsymbol{B}_e^{\mathsf{T}} \boldsymbol{\sigma}_{e,k+1} = \boldsymbol{f}_{k+1}, \tag{1b}$$

with w_e the weights of integration points and $\boldsymbol{B}_e$ the discrete kinematic operator matrix related

[1]The notation $\mathcal{Z}_{e,k+1}$ is introduced here for convenience as it will be used in the following. However, note that the local phase space is independent of k.

to the integration point e. These relations define the set of mechanical constraints,

$$\mathcal{E}_{k+1} = \{z \in \mathcal{Z} \mid \text{ Eqs. (1a) and (1b)}\} \subset \mathcal{Z}_{k+1}. \tag{2}$$

In the model-free data-driven approach the material response is described by a collection of states obtained through experiments or numerical simulations at a finer material scale. These local material data sets denoted

$$\mathcal{D}_{e,k+1} = \{y_i = (\boldsymbol{\varepsilon}_i, \boldsymbol{\sigma}_i) \in \mathcal{Z}_{e,k+1},\ i = 1 \dots N^* \mid \text{ past local history}\} \subset \mathcal{Z}_{e,k+1}, \tag{3}$$

for inelastic behaviours, with N^* the number of material strain-stress pairs, also extend to the global material database at time t_{k+1} as $\mathcal{D}_{k+1} = \mathcal{D}_{1,k+1} \times \cdots \times \mathcal{D}_{M,k+1} \subset \mathcal{Z}_{k+1}$.

The following norm can be defined in the local phase space:

$$\|z_{e,k+1}\|_{\mathbb{C}} = \left[\frac{1}{2}(\mathbb{C}_{e,k+1} : \boldsymbol{\varepsilon}_{e,k+1}) : \boldsymbol{\varepsilon}_{e,k+1} + \frac{1}{2}(\mathbb{C}^{-1}_{e,k+1} : \boldsymbol{\sigma}_{e,k+1}) : \boldsymbol{\sigma}_{e,k+1}\right]^{1/2}, \tag{4}$$

with $\mathbb{C}$ a positive-definite 4th-order tensor that is not related to any material property and can be updated during the simulation. Hence, the (squared) distance between two states $z_{e,k+1}, y_{e,k+1} \in \mathcal{Z}_{e,k+1}$ is:

$$d^2_{\mathbb{C}}(z_{e,k+1}, y_{e,k+1}) = \|z_{e,k+1} - y_{e,k+1}\|^2_{\mathbb{C}}\ . \tag{5}$$

Finally, the extension to the global phase space $\mathcal{Z}_{k+1}$ gives the distance

$$d^2_{\mathbb{C}}(z_{k+1}, y_{k+1}) = \sum_{e=1}^{M} w_e d^2_{\mathbb{C}}(z_{e,k+1}, y_{e,k+1})\ . \tag{6}$$

The data-driven problem is then reformulated into a double minimisation problem, whose solution at step t_{k+1} is given by:

$$\mathcal{S} = \arg \min_{z_{k+1} \in \mathcal{E}_{k+1}} \min_{y_{k+1} \in \mathcal{D}_{k+1}} d^2_{\mathbb{C}}(z_{k+1}, y_{k+1})\ , \tag{7}$$

i.e. the pair (z_{k+1}, y_{k+1}) of states, respectively mechanically admissible and from the material database constrained by history, which are closest to each other according to distance $d_{\mathbb{C}}$. The solution is typically obtained by alternating minimisation over continuous (z) and discrete (y) variables, as illustrated in Fig. 1, that consists in the fixed point iteration:

$$z^{(i+1)}_{k+1} = P_{\mathcal{E}_{k+1}} P_{\mathcal{D}_{k+1}} z^{(i)}_{k+1}, \tag{8}$$

with i the iteration number, $y^{(i)}_{k+1} = P_{\mathcal{D}_{k+1}} z^{(i)}_{k+1}$ the closest point projection onto $\mathcal{D}_{k+1}$ and $P_{\mathcal{E}_{k+1}} y^{(i)}_{k+1}$ the projection of a material state onto $\mathcal{E}_{k+1}$. The discrete nature of the material

database leads to high combinatorial complexity as M and N^* increase. The proportion of local minima also increases, implying a greater probability that the solver will converge on local minima that are far from the global minima [Kanno, 2019].

The main challenge thus lies in the selection of the local material databases $\mathcal{D}_{e,k+1}$ subject to the history of strain and stress as defined in Eq. 3.

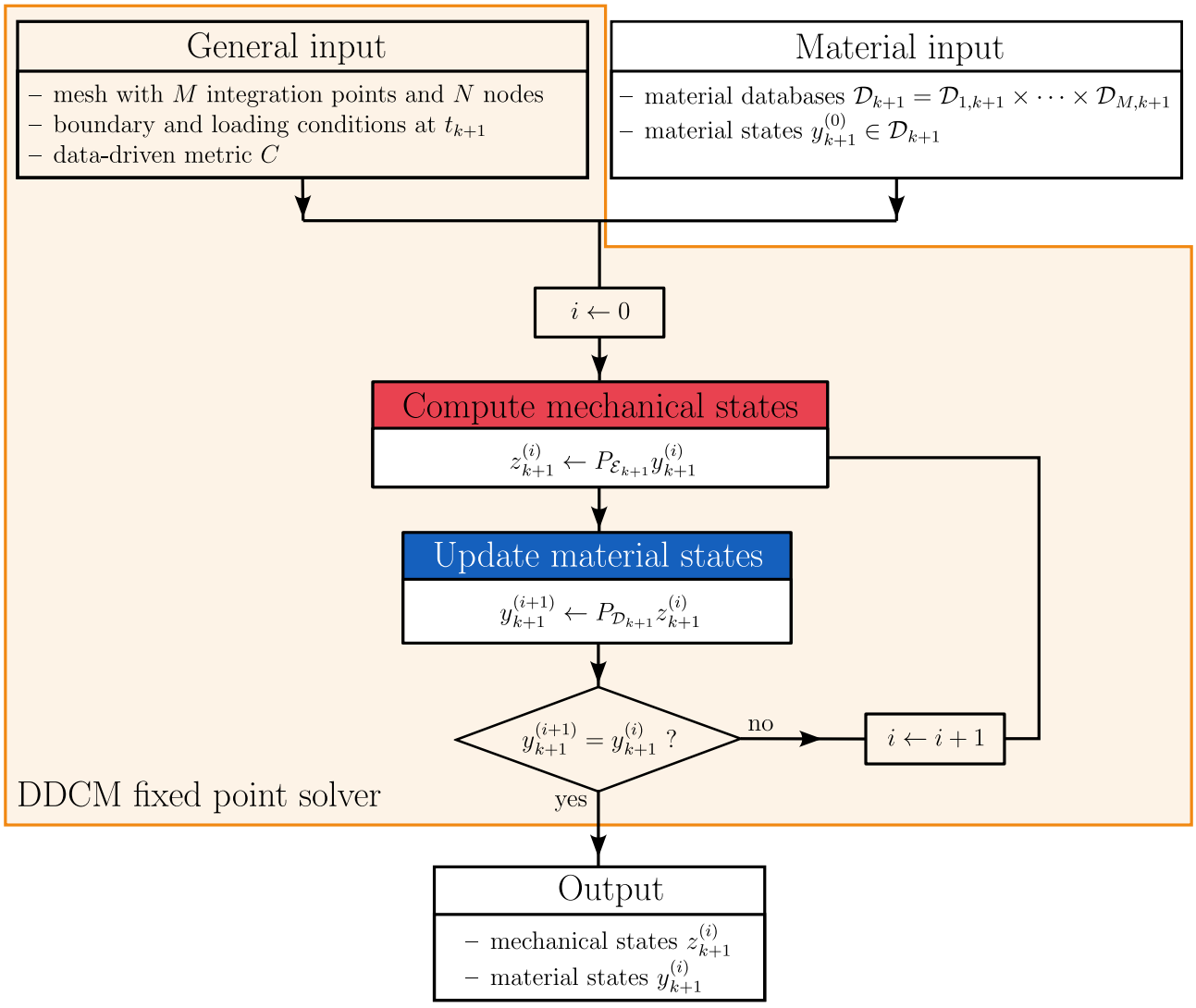

Figure 1: DDCM algorithm for elastic material response at loading step $k+1$

3 Directed graphs for DDCM simulations of history-dependent materials

In this section, we generalise the material database to the material digraph concept. We also show how to construct this digraph and extract the local database $\mathcal{D}_{e,k+1}$ from it.

3.1 Representation of the discrete material behaviour with a material digraph

The definitions and concepts of graph theory provided here can be found in Harary et al. [1965], Bondy and Murty [2008] and Rigo [2016].

A directed graph G (*abbr.* digraph) consists of a pair $(V(G), A(G))$, or shortly, (V, A), with V a set of vertices and A a set of arcs disjoint from V. An incidence function ψ_G links each arc to an ordered pair of vertices in V:

$$\begin{aligned} \psi_G : A &\mapsto (V, V) \\ a &\to (uv), \end{aligned} \tag{9}$$

with u the tail and v the head of a. A cost $c(a)$ is assigned to each arc $a \in A$. G is connected if, for every partition of its vertex set into two non-empty subsets X and Y, there is an arc with one end in X and the other in Y; otherwise, it is disconnected.

We note $G = (V, A)$ the digraph encoding the material behaviour, illustrated on Fig. 2. Its vertices are the material states and its arcs are thermomechanically consistent transitions between these states such that G contains no isolated vertex. Given an arc $(y_i y_j) \in A$ between vertices $y_i, y_j \in V$, the cost $c_D(y_i y_j)$ is the dissipative cost of the transition from states y_i to y_j:

$$c_D(y_i y_j) = D_{i \to j}, \tag{10}$$

with $D_{i \to j}$ the dissipation level of the transition. In this way, a non-dissipative or reversible transition is such that $c_D(y_i y_j) = 0$ and is encoded with two symmetric arcs in the digraph, *i.e.* $(y_i y_j)$ and $(y_j y_i)$, while an irreversible transition is represented with one arc directed such that $c_D(y_i y_j) > 0$.

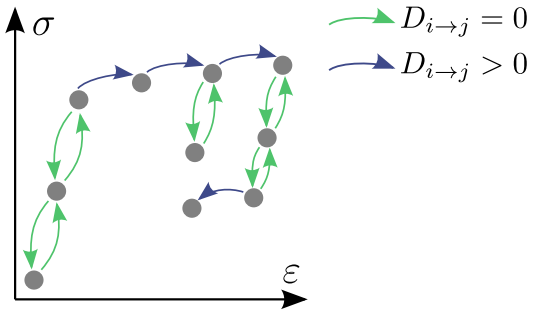

Figure 2: A digraph G for an elastoplastic material response with loading and unloading.

We define a non-dissipative directed subgraph $E = (V(E), A(E)) \subseteq G$ such that $A(E)$ is the set of non-dissipative arcs of $A(G)$ and $V(E)$ the set of vertices joined by arcs in $A(E)$. E might be disconnected and all the vertices belonging to the same connected component can be linked together via zero-cost arcs only (see Fig. 3 (a)). As a result, each connected component is

associated with an elastic domain, which fundamentally corresponds to a clique, *i.e.* a digraph in which every pair of vertices is joined by exactly two arcs, one in each direction.

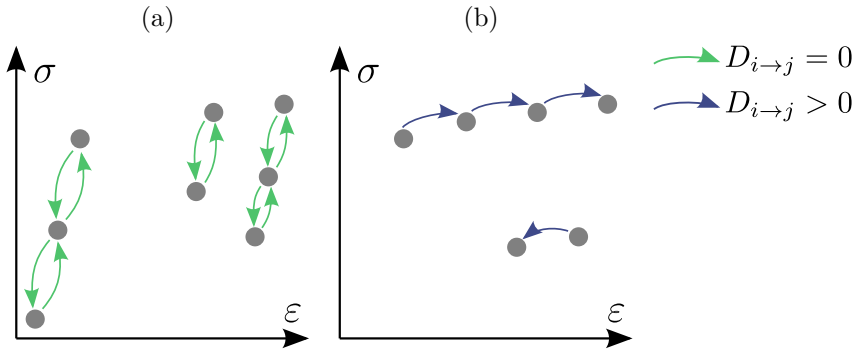

Figure 3: Subgraphs of G (Fig. 2): (a) non-dissipative subgraph E, (b) dissipative subgraph P.

Furthermore, we define the dissipative directed subgraph $P = (V(P), A(P)) \subseteq G$ such that $A(P) = A(G) \setminus A(E)$ and $V(P)$ has no isolated vertex. $A(P)$ thus contains all thermodynamically irreversible transitions: all arcs in $A(P)$ have strictly positive dissipative cost, as in Fig. 3 (b).

For a database representing a purely elastic material behaviour, the digraph G is a clique and is equal to the non-dissipative subgraph E. Therefore, the local database $\mathcal{D}_{e,k+1}$ contains all material states.

From this point on, we will use the terms of computational mechanics and graph theory interchangeably to refer to a state of the material database or the corresponding vertex in the digraph and a thermomechanical transition or the equivalent arc. We also call *material digraph* and write G the digraph associated with the material data set.

3.2 Distinction between identical states in the constitutive space

Let us now consider two identical states in the constitutive space $y_k = (\boldsymbol{\varepsilon}, \boldsymbol{\sigma})$ and $y_\varkappa = (\boldsymbol{\varepsilon}, \boldsymbol{\sigma})$ but with different histories of strain and stress, denoted $\{y_l\}_{l\leqslant k}$ and $\{y_\lambda\}_{\lambda\leqslant\varkappa}$ respectively.

A path is a special type of graph made of an ordered sequence of arcs such that the tail of each arc of the sequence coincides with the head of the preceding and where all the visited vertices are pairwise distinct. A path with the same first and last vertices is called a cycle.

Histories $\{y_l\}_{l\leqslant k}$ and $\{y_\lambda\}_{\lambda\leqslant\varkappa}$ exist in the digraph as paths whose vertices are the states of histories. Yet according to its definition, P is acyclic. Thus, if either $\{y_l\}_{l\leqslant k}$ or $\{y_\lambda\}_{\lambda\leqslant\varkappa}$ or both contain any arcs $a_k \in A(P)$ and $a_\varkappa \in A(P)$ that belong to different connected components of P, then y_k and $y_\varkappa$ are different vertices of G (see Fig. 4). Otherwise, y_k and $y_\varkappa$ belong to the same elastic domain and are strictly equal (same characteristics in the constitutive space and same loading histories).

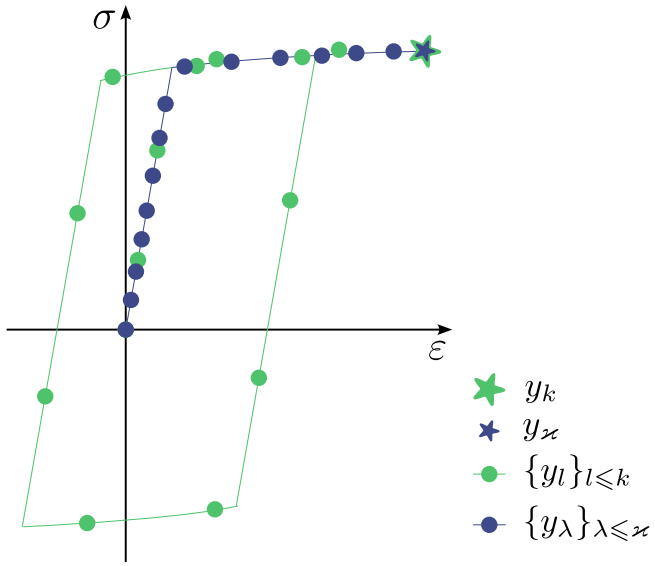

Figure 4: Identical states y_k and $y_\varkappa$ in constitutive space $(\boldsymbol{\varepsilon}, \boldsymbol{\sigma})$ but with different histories.

In conclusion, the digraph framework makes it possible to distinguish between states with the same localisation in the constitutive space but different past histories and thus to represent complete and repeated loading cycles with various loading directions without any modification in the formalism. In that case, simulating the material response is only achievable in the sampled areas of the constitutive space with adequate dissipation levels. For example, to predict the mechanical behaviour for a loading-unloading-reloading cycle, the material digraph must encode the material response for at least one cycle and for the same strain range as implied by the applied loading.

3.3 Local database selection

The solution of the data-driven problem at time step $k+1$ and integration point e is given by the pair $(z_{e,k+1}, y_{e,k+1})$ of respectively mechanical and material states computed with Eq. 7. This first requires knowing the local material database $\mathcal{D}_{e,k+1}$, which contains all the admissible states in the data set given the history of e.

As shown in Section 3.1, the history of a state is encoded in the global material database digraph. Therefore, selecting $\mathcal{D}_{e,k+1}$ now comes down to searching for admissible futures in the material database or equivalently to searching for existing paths in the digraph.

To do that, we use the concept of tree as defined in graph theory. A tree is a connected graph that contains no cycles. As for digraphs, trees can be written in terms of coupled sets of arcs and vertices as $T = (V(T), A(T))$. Starting from a root $r \in V(G)$, a tree $T(r)$ contains all vertices of G that can be reached from r, and the arcs leading to them. Hence a vertex $v \in V(G)$ belongs to $V(T)$ if there is a path in G leading from r to v.

Knowing that digraph G encodes the thermodynamically admissible transitions between states of material database $\mathcal{D}$ and given a root $y_{e,k}$, we build a rooted tree from this vertex and

denote it $T(y_{e,k})$. Finally, the local material database at $k+1$, illustrated in Fig. 5, contains the local states corresponding to the vertices of $T(y_{e,k})$:

$$\mathcal{D}_{e,k+1} = \{y_i, \forall y_i \in V(T(y_{e,k}))\}\,. \tag{11}$$

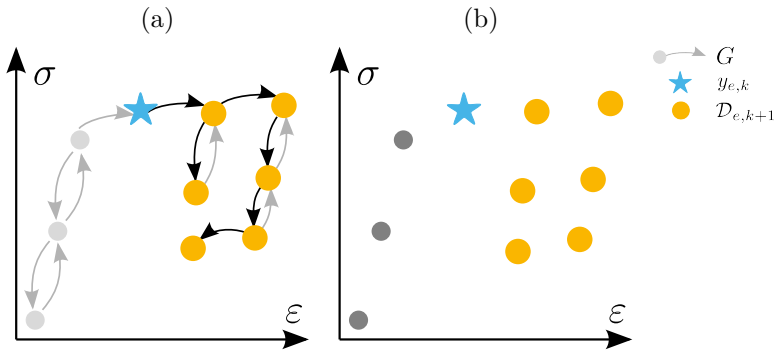

Figure 5: (a) Tree $T(y_{e,k})$ built from G (Fig 2). (b) Material database, root $y_{e,k}$ and local database $\mathcal{D}_{e,k+1}$.

3.4 Local database reduction

The alternating minimisation solver used for elastic problems is highly sensitive to local minima [Kirchdoerfer and Ortiz, 2017], the number of which increases in inelasticity. In the incremental problem developed above, we usually consider small loading increments. We therefore assume small increments of the material solution at a material point e both in terms of distance in the constitutive space and dissipative cost of the transition.

To this end, we introduce the concept of path cost defined as the sum of every arc's cost:

$$c(p) = \sum_{a \in A(p)} c(a), \tag{12}$$

with p a specific path and $A(p)$ the set of its arcs. A minimum cost path is the shortest path with respect to cost $c(\bullet)$ and is denoted $[u, v]_c$ with u and v the tail and head vertices respectively. As part of an elastoplastic local database, u is always a tree's root and v is a vertex of $T(u)$, which ensures that there is at least one path from u to v.

Finally, three conditions on the states belonging to $\mathcal{D}_{e,k+1}$ as defined in Eq. 11 are applied to reduce local material databases:

1. squared data-driven distance:

$$d^2_{\mathbb{C}}(y_k, y_i) \leqslant \texttt{TOL1}, \tag{13}$$

2. cumulated squared data-driven distance along the path:

$$c_\delta(p = [y_k, y_i]_\delta) = \sum_{a \in A(p)} c_\delta(a) \leqslant \texttt{TOL2}, \tag{14}$$

with $c_\delta(a) = d^2_{\mathbb{C}}(\text{tail}(a), \text{head}(a))$,

3. path dissipative cost:

$$c_D(p = [y_k, y_i]_D) = \sum_{a \in A(p)} c_D(a) \leqslant \texttt{TOL3}, \tag{15}$$

with $c_D(a) = D_{\text{head}(a) \to \text{tail}(a)}$.

It should be noted that the first two conditions involve algorithmic criteria while the latter is a material criterion. Furthermore, for any path from y_k to y_i, the data-driven distance is always smaller or equal to the distance along the path used in condition 2. The effect of these criteria and their combinations will be discussed in more detail in Section 4.3.

Fig. 6 details the new data-driven procedure for the incremental inelastic problem. As stated by Eggersmann et al. [2019], the difference with the elastic case lies in the definition of local databases while the following algorithmic steps remain the same.

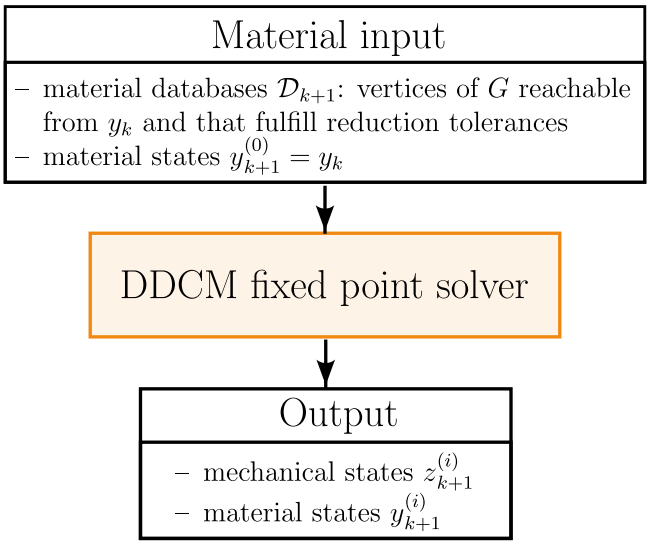

Figure 6: DDCM algorithm for rate-independent elastoplastic material response at loading step $k + 1$.

3.5 Numerical representation of the material digraph

In practice, material data can be generated in two different ways. On the one hand, it can come from numerical simulations, for example in the context of multiscale modelling [Karapiperis et al., 2021, Gorgogianni et al., 2023]. In this case, a constitutive model provides all the information needed to build the material digraph, *i.e.* strains, stresses and dissipation. On the other hand, experimental tests can also be used to generate this data, using a full-field strain measurement technique coupled with DDI for strains and stresses. Dissipation can be inferred from calorimetric measurements with assumptions on potential heat exchanges [Chrysochoos et al., 2010, Seghir et al., 2013, Vinel, 2022].

A graph is usually encoded by an adjacency matrix, *i.e.* a square matrix A that represents adjacent vertices of the graph. A component A_{uv} usually equals the number of arcs from vertex u to v.

In the present work, the numerical representation of the material digraph G is an $N^* \times N^*$ adjacency matrix $A_{ij} = \exp(c(y_i y_j))$ where N^* is the number of points in the data set and $c(y_i y_j)$ is the cost of arc $(y_i y_j)$. The exponential function allows to distinguish between the absence of arc from a zero-cost arc. We use two matrices to encode the costs defined above:

- data-driven distance matrix: $A^{\delta}_{ij} = \exp(c_{\delta}(y_i y_j))$,
- dissipative cost matrix: $A^{D}_{ij} = \exp(c_{D}(y_i y_j))$.

Both matrices represent the same material digraph (same vertices and same arcs) but with different arcs' costs. Therefore they have the same sparsity pattern that might be stored only once.

It is to be noted that the material digraph should be designed in such a way that it contains a sufficient number of arcs to provide enough information about the material behaviour and thus improve the solver's precision, and yet as few arcs as possible to ensure adjacency matrices' sparsity and speed up computations (*e.g.* graph search). In particular, cliques should not be fully encoded as such as long as at least one path exists between all vertices of the corresponding elastic domain. For instance, an enhanced representation of cliques could be developed to speed up computations.

Finally, local databases can be obtained thanks to generic graph search algorithms [Bondy and Murty, 2008] or shortest-path algorithms like Dijkstra's [Dijkstra, 1959]. We implement the reduction criteria introduced in Section 3.4 within the chosen routine as a boundary or limit of path cost.

4 Numerical implementation and investigation of a single element problem

4.1 Material digraph construction

We generate an artificial database thanks to an elastoplastic material model with linear kinematic hardening. The yield function is given by:

$$f(\sigma, \varepsilon_{\mathrm{p}}) = |\sigma - H\varepsilon_{\mathrm{p}}| - \sigma_y \tag{16}$$

with hardening modulus H, yield limit σ_y and plastic strain ε_{p}.

We build a regular data set representing the material response under elastoplastic loading, elastic unloading and plastic reloading. The strain increment in the non-hardened elastic domain and the first dissipative part is 0.01 %. Unloading and reloading paths are generated with a plastic strain increment of 0.2 % and an elastic strain increment of 0.04 %.

We define the arcs of the material digraph as follows:

- elastic domains, or sets of states that could be represented by cliques, are encoded as minimum spanning subgraphs with respect to the data-driven distance, *i.e.* such that all vertices are connected in both directions and with arcs that minimise the total (data-driven) cost of each connected component,
- elastic domains are linked together with dissipative arcs that encode irreversibility, *i.e.* such that

$$\int_0^t |\dot{\varepsilon}_{\mathrm{p}}| dt > 0.$$

We rely on the order of the generated sequence of material points and on the dissipation level to determine the arcs' direction and whether they should be encoded. The final digraph is similar to the one presented in Fig. 2. According to this definition, superimposed states in the (ε, σ) configuration (see Fig. 7 (a)) might have different histories and dissipation levels (see Fig. 7 (b)) and are different vertices of the material digraph.

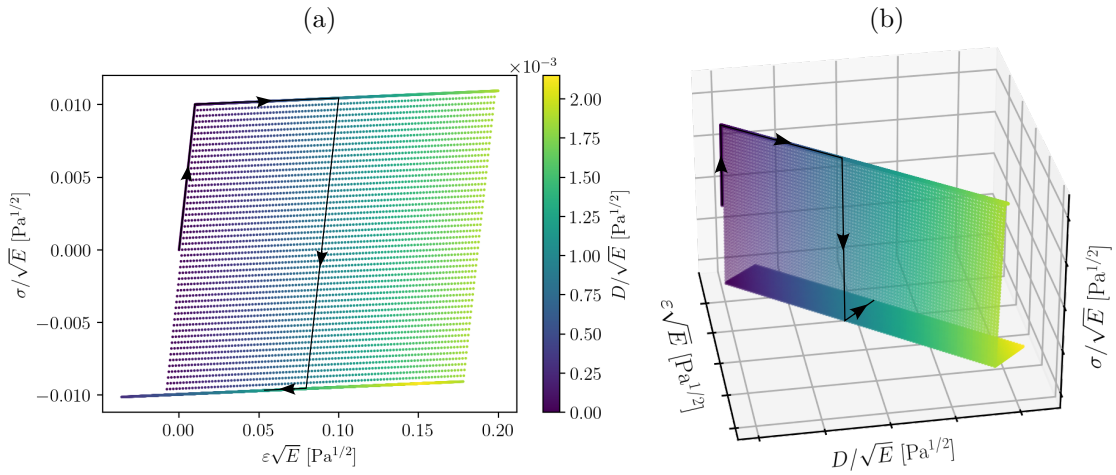

Figure 7: Material data set under loading, unloading and reloading conditions, the black solid line represents an example history: strain and stress reach a maximum before decreasing, while dissipation only increases. (a) In stress-strain space, colouring represents the dissipation state, (b) 3d representation.

A local database contains potential states for a given loading increment. The criteria defined in Section 3.4 tighten eligibility conditions and therefore downsize the database. The data-driven distance criterion is implemented with `sklearn`'s Nearest Neighbor algorithm and a maximum radius equal to $\overline{\texttt{TOL1}}$. The path distance and cost conditions are computed with Dijkstra's algorithm implementation from Python library `scipy.sparse.csgraph` and graph adjacency matrices A^{δ} and A^{D} encoding respectively data-driven distance and transition dissipation as arcs' cost. Again, $\overline{\texttt{TOL2}}$ and $\overline{\texttt{TOL3}}$ represent the maximum admissible path cost for the last node of the sequence to be in the local database. Note that $\overline{\texttt{TOL}}$ represent relative tolerances taken with respect to the chosen metric C.

Fig. 8 shows local databases for different reduction criteria in space $(\varepsilon\sqrt{E}, \sigma/\sqrt{E})$, which has the same dimensions as the constitutive space. In this representation and because we set the DDCM parameter C equal to Young's modulus, data-driven and euclidean distance are equivalent. Criterion 1 expresses a maximum data-driven radius centred on the root and is represented by a circle in these coordinates (see Fig. 8 (b)).

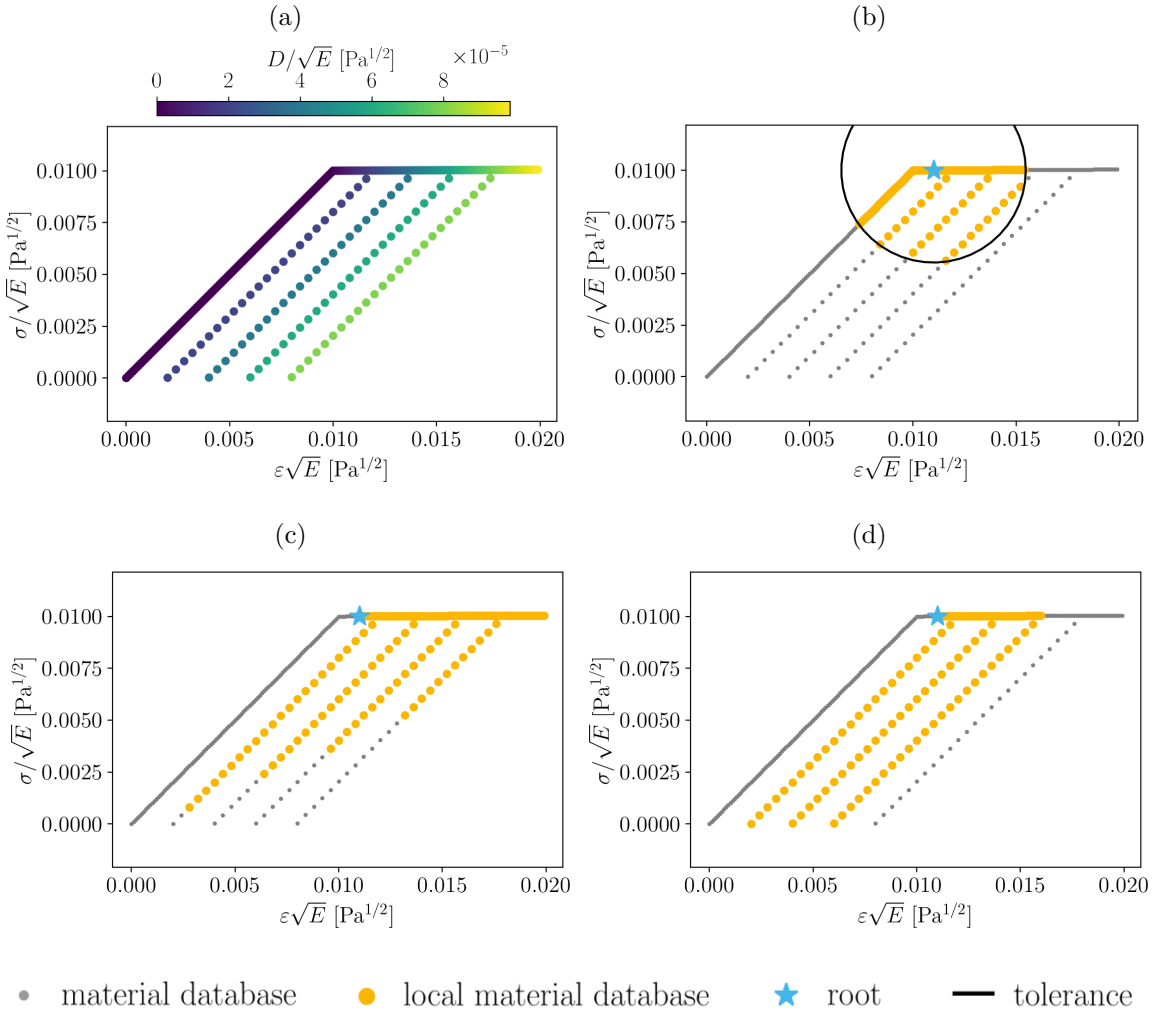

Figure 8: Material database and effect of reduction criteria on the local database. (a) Material database. (b) Data-driven distance, TOL1=$10^{-5} \times C$. (c) Cumulated data-driven distance along the path, TOL2=$10^{-2} \times C$. (d) Path dissipative cost, TOL3=$5 \cdot 10^{-5} \times C$.

The second criterion limits the data-driven distance along the shortest path from the root to a point in the local database (see Fig. 8 (c)). Note that the notion of path included in this cost excludes points corresponding plastic unloading, contrarily to previous criterion. It is equivalent to a radius in the graph representation and seems more suitable to our study as it better reflects the effective loading path leading to a given transition.

Finally, the dissipative criterion in Fig. 8 (d) adds all vertices in elastic domains obtained with dissipative increments lower than the chosen tolerance to the database. It is used in the following simulations in association with the second criterion. Other criteria could also be

developed, for instance, to explicitly forbid non-monotonic increments (in dissipation). This assumption is for example used in the classical return-mapping algorithm for elastoplastic constitutive modelling.

4.2 Evaluation of results quality

DDCM solution strongly depends on database sparsity, especially when DDCM metric C is not optimised [Kirchdoerfer and Ortiz, 2017, Eggersmann et al., 2019]. This is the case here as we choose a fixed value for this parameter throughout the simulation despite slope discontinuities in the material data. This work aims to demonstrate the ability of the present graph-based approach to represent the mechanical solution independently of the convergence study, which should be improved by an enhanced solver, *e.g.* through game theory as suggested by Weinberg et al. [2023]. To decouple the influence of these parameters, we seek to limit the data set's density effect by filtering the FE reference solution with the data set. We therefore introduce a new indicator: the data-driven projection of the FE reference solution onto the material database. This value is computed in two steps:

1. projecting the FE solution z^{ref} onto the material database (*reference material states*),

$$y^{\text{ref}} = P_{\mathcal{D}} z^{\text{ref}} \ ,$$

2. re-projecting the reference material states onto equilibrium (*projected reference mechanical states*),

$$z^{\text{proj}} = P_{\mathcal{E}_{k+1}} y^{\text{ref}}.$$

The projected states represent the best solution achievable by a data-driven solver with given data and chosen metric. This is particularly important as the material data set might contain a non-uniform discretisation of the constitutive space with possibly very different sampling rates in the elastic and dissipative domains. Even for a regular data set as in Fig. 9, gaps appear in the projected data when the database is too sparse in the neighbourhood of the FE solution. Hence the distance between the DDCM solution and the projected reference solution provides a neutral indicator regarding data sparsity and is used in the following to evaluate the results accuracy.

It is noteworthy that the distance between DDCM solution and projected reference states is not totally independent of the database density, since the convergence of the fixed point algorithm depends on this density. However, this indicator is more neutral than the comparison with the FE reference solution usually used.

4.3 Resolution of a data-driven problem

We illustrate the approach with the one-dimensional spring-bar element system on Fig. 10 subject to a strain $\bar{\varepsilon}$ up to 15 % followed by elastic unloading and plastic reloading in

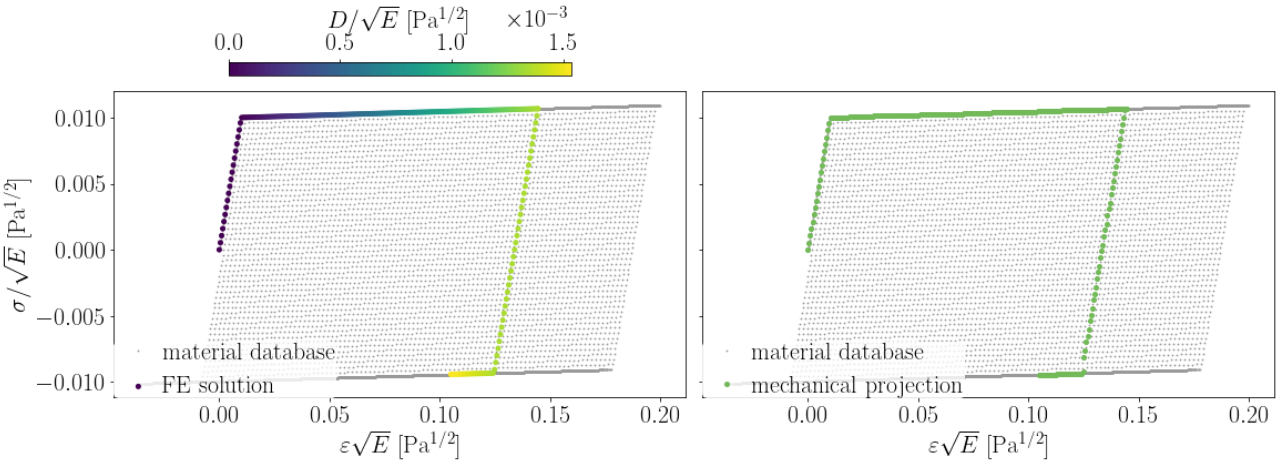

Figure 9: Material data set and (left) FE solution, (right) projected reference mechanical states.

compression with $\bar{\varepsilon}$ down to 10 %, a spring stiffness $K = 2l_0E$ and $l_0 = 1$ m the bar length. The material data set is artificially generated as described in Section 4.1 with constitutive model parameters $E = 1$ Pa, $H = E/200$ and $\sigma_y = E/100$ and the data-driven metric is set to $C = E$.

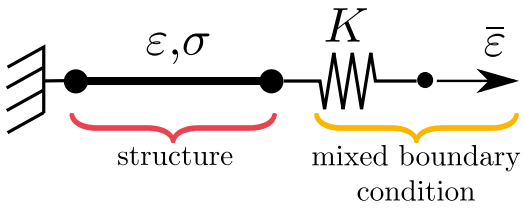

Figure 10: Boundary value problem: spring-bar element system

Fig. 11 (a) shows the mechanical and material states obtained at all time steps. The DDCM solver is able to retrieve a good approximation of the path shape even during the unloading phase, where the material database does not contain points with the exact reference dissipation level. The projected reference solution is well recovered except from the onset of unloading (time step 188) until reloading (time step 225) where the distance between DDCM mechanical states and projected reference mechanical states increases (see Fig. 11 (b) and (c)). As stated by Kanno [2019], this phenomenon is characteristic of the solver and occurs even for simple problems. Moreover, absolute strain and stress errors, computed as

$$\begin{aligned} \mathrm{err}_\varepsilon &= |\varepsilon_k - \varepsilon_k^{\mathrm{proj}}| \ , \\ \mathrm{err}_\sigma &= |\sigma_k - \sigma_k^{\mathrm{proj}}| \ . \end{aligned} \tag{17}$$

are given on Fig. 11 (c) between mechanical and projected reference mechanical states. The evolution of these values during loading is coherent with the chosen metric which affects a lower weight on stresses compared to strains and thus tends to approximate the latter better.

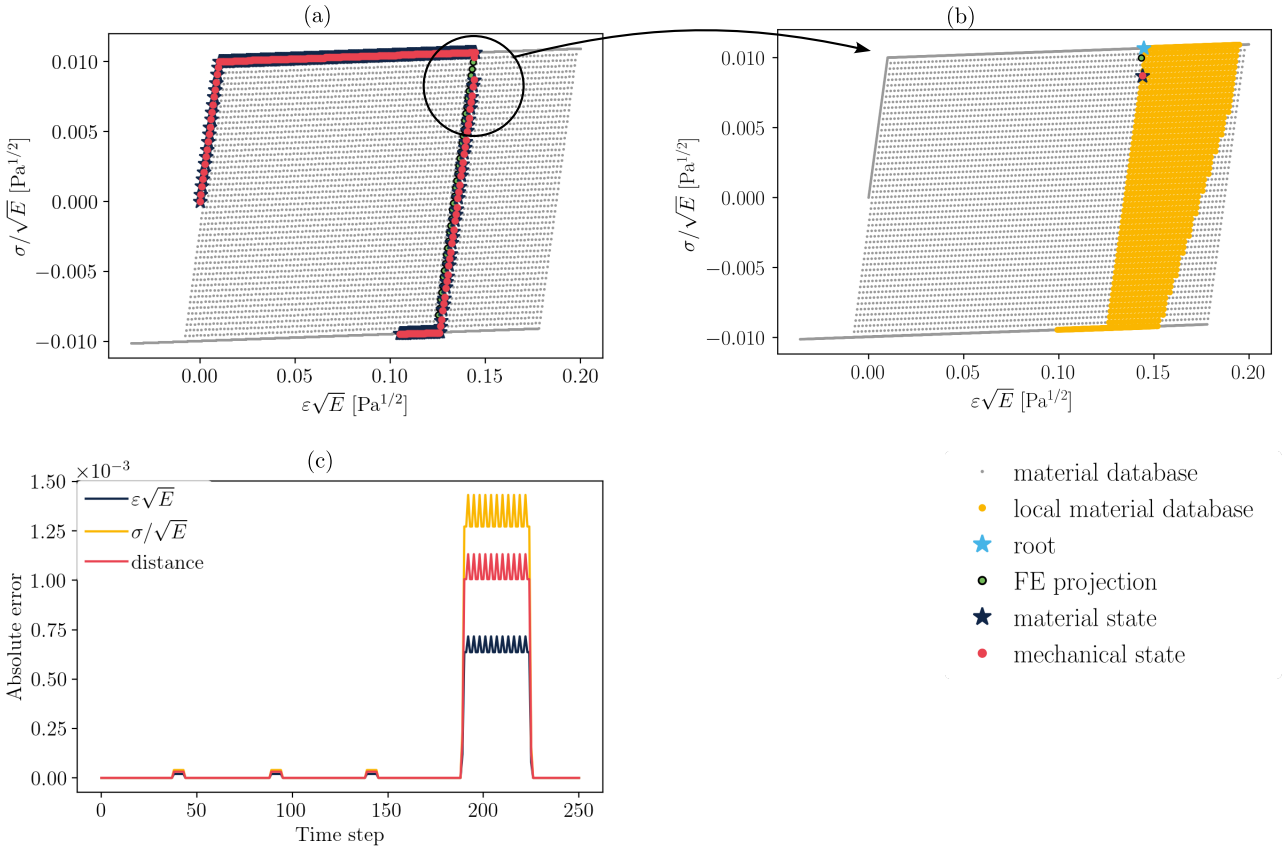

Figure 11: (a) Mechanical and material solutions for the spring-bar element model under loading, unloading and reloading. (b) local material database, DDCM solution (blue star and red point) and projected reference mechanical state (green point) at maximum loading. (c) Errors between DDCM and projected reference mechanical states.

5 Numerical experiment on a truss problem

5.1 Truss structure and related challenges

The simulation of a truss structure is a combinatorial problem that involves coupled minimisations of mechanical and material states at all integration points. The DDCM alternating minimisation solver ensures global convergence but is likely to fall into local minima, which could lead to a completely unreliable solution both at local and global scales. As an example, Fig. 12 (a) shows a 2D truss composed of 102 nodes and 252 elements, subject to displacement-driven boundary conditions. The top and left boundaries are fixed while right and bottom nodes are subject to the same displacement up to 7 mm and back to 6 mm with 135 increments, as illustrated in Fig. 12 (b). The material data set is artificially generated as described in Section 4.1 with constitutive model parameters $E = 217.5$ GPa, $H = 1$ GPa, $\sigma_y = 250$ MPa, a strain increment of 0.001 % and the metric is set to $C = E$. These parameters are representative of steel and close (in adimensional form) to those used in the previous section. The local database reduction tolerances are set to TOL2 $= 5 \times 10^3 \times C$ and TOL3 $= 10^{-5} \times C$.

DDCM mechanical and material states obtained at all loading steps for two plastically

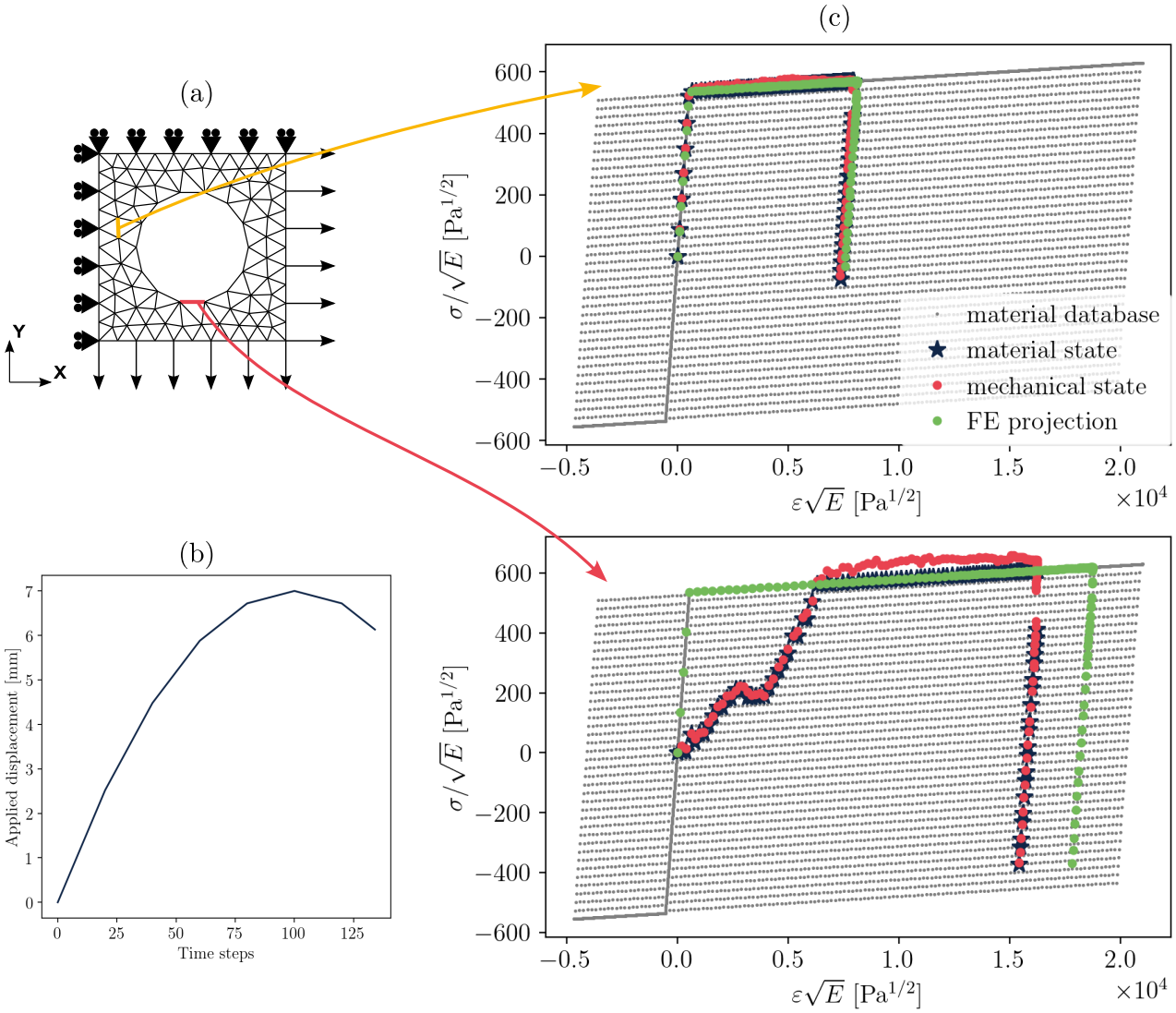

Figure 12: (a) Truss geometry and boundary conditions. (b) Applied displacement. (c) Evolution of DDCM mechanical, material states and projected reference mechanical states for yellow and red elements during loading.

deformed elements of the mesh are illustrated on Fig. 12 (c). Although prediction is satisfactory for the yellow element, the solution computed for the red element does not match the expected elastoplastic response. Strain "sliding" is observed for strains from 0 Pa$^{1/2}$ to 6×10^3 Pa$^{1/2}$ and DDCM underestimates maximum strain, even though the overall path shape is consistent. These effects are related to the constitutive space discretisation with finer strain increments along with a metric value that gives strains a higher weight in the distance calculation. Besides, Fig. 13 (a) highlights the median of relative strain and stress errors, calculated as:

$$\begin{aligned} \mathrm{err}_\varepsilon &= |\varepsilon_{e,k} - \varepsilon^{\mathrm{proj}}_{e,k}|/\varepsilon^{\mathrm{proj}}_{e,k} \ , \\ \mathrm{err}_\sigma &= |\sigma_{e,k} - \sigma^{\mathrm{proj}}_{e,k}|/\sigma^{\mathrm{proj}}_{e,k} \ . \end{aligned} \tag{18}$$

The difference in magnitude between the strain and stress medians is consistent with the strain

and stress discretisation: with the data-driven distance, which transforms into the euclidean distance in the $(\varepsilon\sqrt{E}, \sigma/\sqrt{E})$ space, the stress range is much lower than that of strains, and stress discretisation is denser.

Fig. 13 (b) shows how relative errors are distributed in the mesh with elements sorted by increasing strain. The plotted time step corresponds to the loading phase, which implies that both strain and stress should grow together. For some elements labelled in range 220-252 nonetheless, stress levels are lower than expected, which corresponds to the stress underestimation observed on Fig. 12 (c) for the red element. At this specific loading step, 19 % of the elements have strain relative errors greater than 100 %, half of which reaching strain levels smaller than 500 $\mathrm{Pa}^{1/2} \approx 0.1$ %.

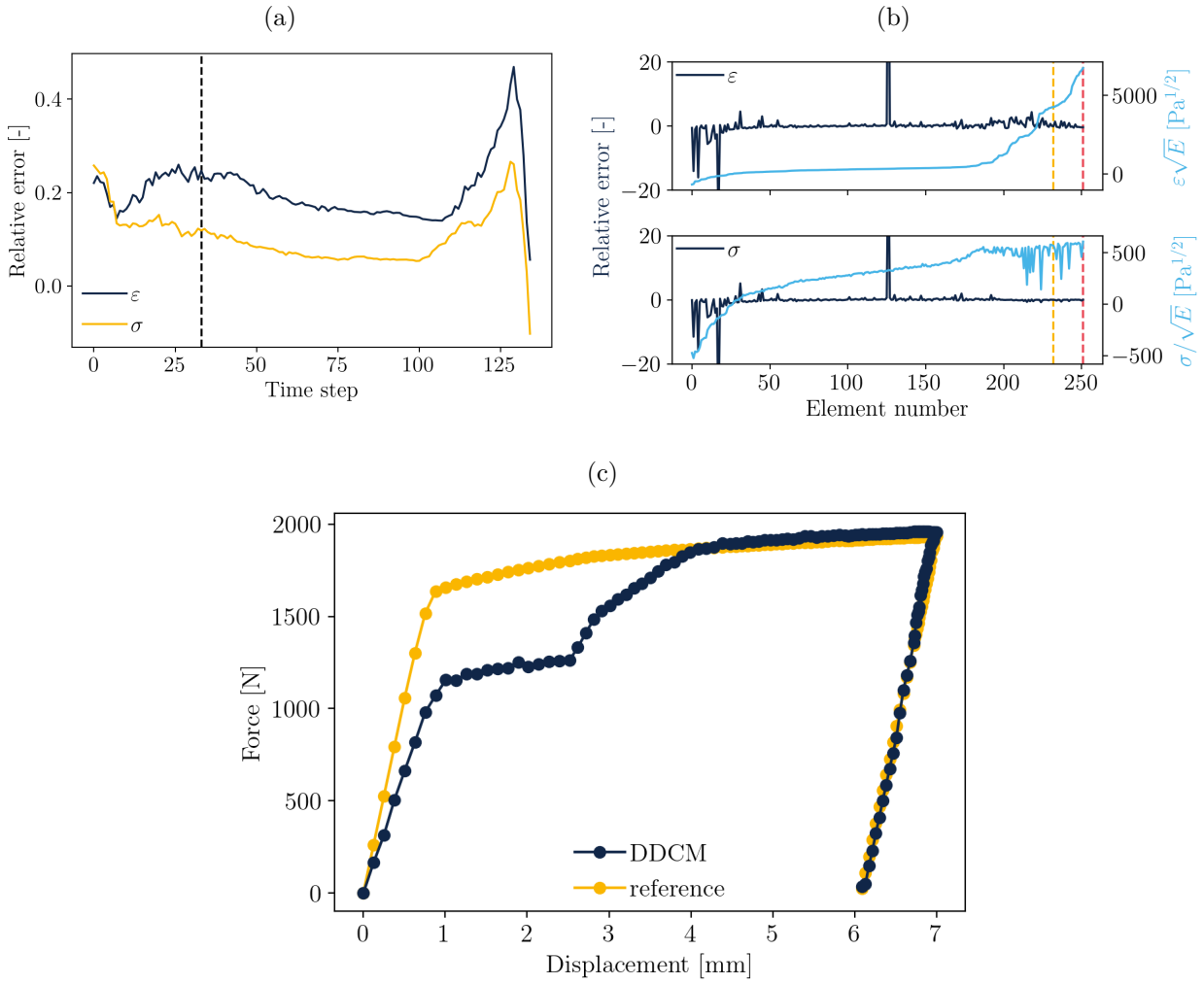

Figure 13: (a) Median relative errors between DDCM and projected reference mechanical states. (b) Relative errors at time step 33 (black dotted line on plot (a)) between DDCM and projected reference mechanical states, the yellow and red dotted lines refer to the elements highlighted on Fig. 12 (a). (c) Evolution of resulting force against displacement for the truss problem.

Additionally, combinatorial resolution optimises the global objective function to the detriment of local behaviours. The macroscopic response, illustrated on Fig. 13 (c) by the

evolution of the resulting force along the truss right boundary as a function of the applied displacement, also exhibits the consequence of local sliding for lower strains. The DDCM maximum force however is close to reference as local over- and underestimations of strains offset each other.

Even though the laws of mechanics and thermodynamics are fulfilled, respectively with the mechanical fields satisfying kinematic compatibility and equilibrium, and the material fields satisfying the condition of non-negativity of dissipation, the lack of a unique solution within local material databases combined with the sensitivity of the alternated minimisation solver to local minima leads to defective results at local and global scales. We therefore discuss a possible improvement of the solver for combinatorial elastoplastic problems.

5.2 Predictor-corrector algorithm

We suggest to implement a two-step non-dissipative predictor-dissipative corrector algorithm inspired by resolution methods for constitutive models. This first requires to define two types of local databases:

- a non-dissipative prediction local database $\mathcal{D}^{\text{pred}}_{e,k+1} \subset \mathcal{D}_{e,k+1}$ that only contains states such that the dissipative cost of any path from subgraph root $y_{e,k}$ to a state of the local database is null, or equivalently, local databases only contain states belonging to the same elastic domain as the roots,
- a dissipative correction local database $\mathcal{D}^{\text{corr}}_{e,k+1} = \mathcal{D}_{e,k+1}$ obtained as defined in Section 3.3 by taking the prediction's material state $y^{\text{pred}}_{e,k+1}$ as root.

Local databases $\mathcal{D}^{\text{pred}}_{e,k+1}$ and $\mathcal{D}^{\text{corr}}_{e,k+1}$ might also respect the reduction criteria based on cumulated data-driven distance along path and path dissipative cost, introduced in Section 3.4. In practice, $\mathcal{D}^{\text{pred}}_{e,k+1}$ is calculated by setting the tolerance on dissipative path cost $\overline{\texttt{TOL3}}$ to zero and the tolerance on path data-driven distance $\overline{\texttt{TOL2}}$ for $\mathcal{D}^{\text{corr}}_{e,k+1}$ must be chosen such that $\mathcal{D}^{\text{pred}}_{e,k+1} \subseteq \mathcal{D}^{\text{corr}}_{e,k+1}$ to allow permanent deformation of elements.

A data-driven iteration is thus performed in two steps as illustrated on Fig. 14. First, a non-dissipative step, called predictor, where the resolution is performed using $\mathcal{D}^{\text{pred}}_{e,k+1}$ with root $y_{e,k}$ until convergence. Then a dissipative computation or correction is performed using $\mathcal{D}^{\text{corr}}_{e,k+1}$, equal to the local database defined in Section 3.3 and Section 3.4, obtained by taking the predicted material state $y^{\text{pred}}_{e,k+1}$ as root vertex. In addition, the material states in the correction stage are initialised with the predicted material states. The corrected states are also the final solution of the loading increment. This modification of the DDCM algorithm does not involve any additional hypothesis or parameter but is solely based on physics.

The truss problem introduced above is now solved with the enhanced algorithm. The DDCM solution for the yellow element depicted in Fig. 12 is shown on Fig. 15 (a) and is mostly similar to the solution obtained in Section 5.1, except in the unloading part where it is slightly overestimated. Fig. 15 (b) shows the states obtained for the red element: the predicted

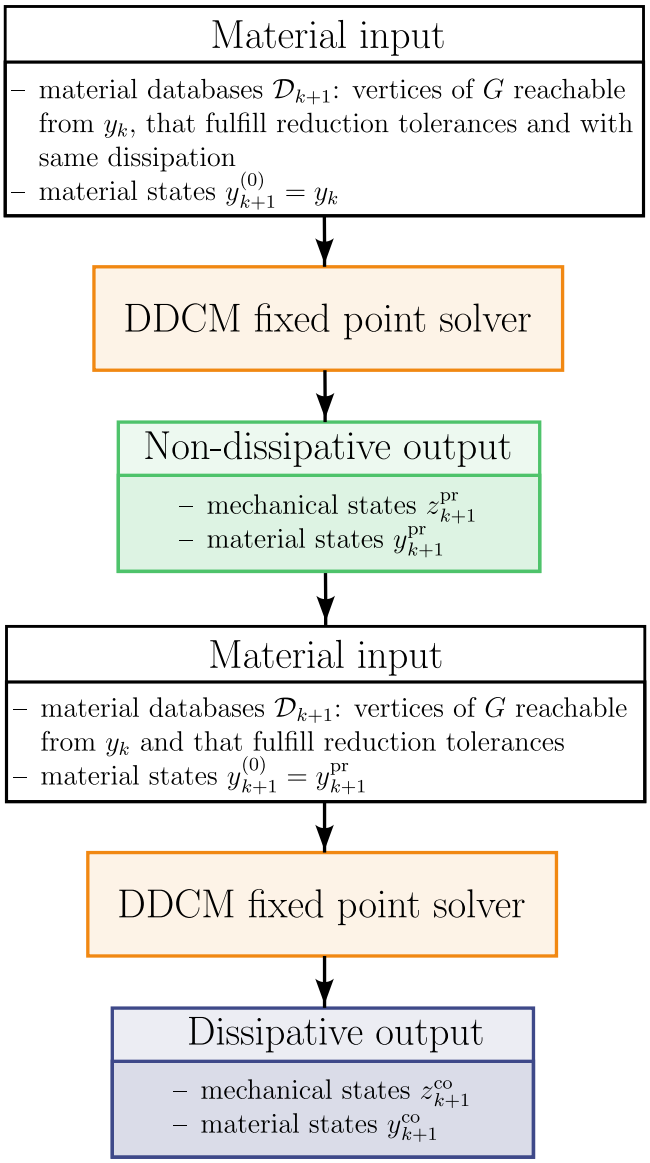

Figure 14: Illustration of the predictor-corrector elastoplastic DDCM algorithm.

solution improved significantly compared to Fig. 12. The material response is well predicted in the reversible part, although the solver still tends to underestimate strains in the dissipative domain, leading to early unloading.

Fig. 16 confirms that relative errors also improved compared to the previous simulation. The median values were reduced down to 20 % of those in the previous simulation. Additionally, at loading step 33, high relative errors mostly restrict to slightly deformed elements: only 11 % of the elements exhibit strain relative errors greater than 100 %, half of which with strain levels smaller than 500 $\mathrm{Pa}^{1/2} \approx 0.1$ %. Finally, the macroscopic response of the mesh, evaluated by the resulting force against displacement curve on Fig. 16 (c), is recovered as well.

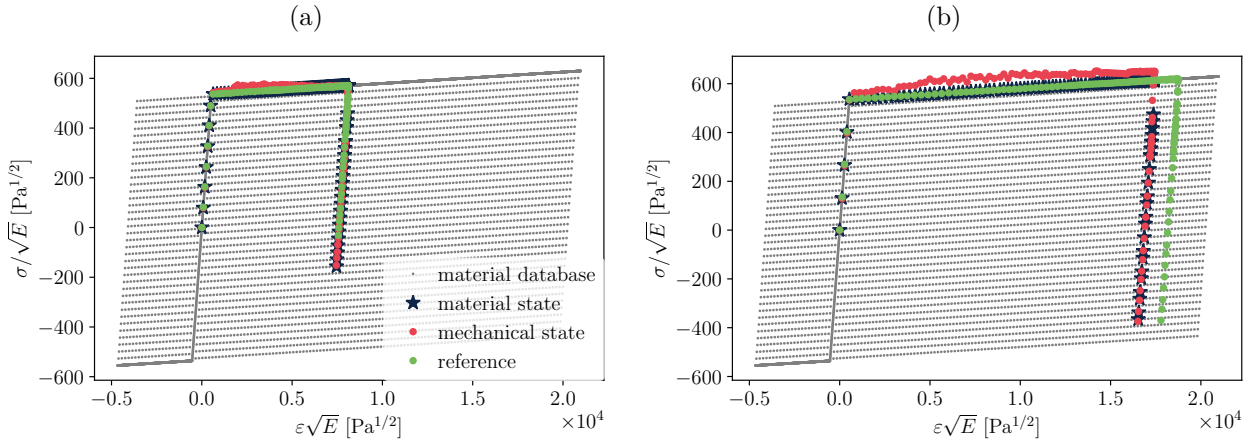

Figure 15: Evolution of DDCM mechanical, material states and projected reference mechanical states for the truss problem with predictor-corrector algorithm. (a) Yellow element. (b) Red element.

6 Conclusion

This work extends the concept of material database defined in the original DDCM approach to that of material digraph, suitable for representing history-dependent behaviours. The former is merely a sampling of the strain-stress response of the material, and therefore sufficient to represent elastic behaviours. The latter also encodes thermomechanically admissible transitions between states through the arcs of the digraph. In an incremental scheme, a local database representing the local material behaviour has to be selected at each loading step. We show that this selection operation can be efficiently performed with graph tools by computing a rooted tree from the last local material state. The obtained local database represents every possible future states that are compatible with the thermomechanical history. It therefore contains states that might be distant regarding different criteria. A reduction of the local database is proposed by filtering states according to three of those criteria. Additional conditions could also be introduced to filter states obtained from dissipation-wise non-monotonic paths. This restriction, similar to the hypothesis used by the classical return-mapping algorithm in constitutive modelling, would ensure that the local database only contains states belonging either to the same elastic domain as the root or to a linked dissipative part of the material database.

The presented numerical results were obtained for material databases computed with an elastoplastic constitutive model with linear kinematic hardening. However, neither the material digraph nor the local databases used by the DDCM solver rely on any internal variable. We thus believe that the method can handle any hardening type and possibly other inelastic behaviours, although the latter may require adapting the constitutive space. Cyclic loading can be predicted as long as the constitutive space discretisation (i.e. the database) covers the range of strain and dissipation that needs to be studied.

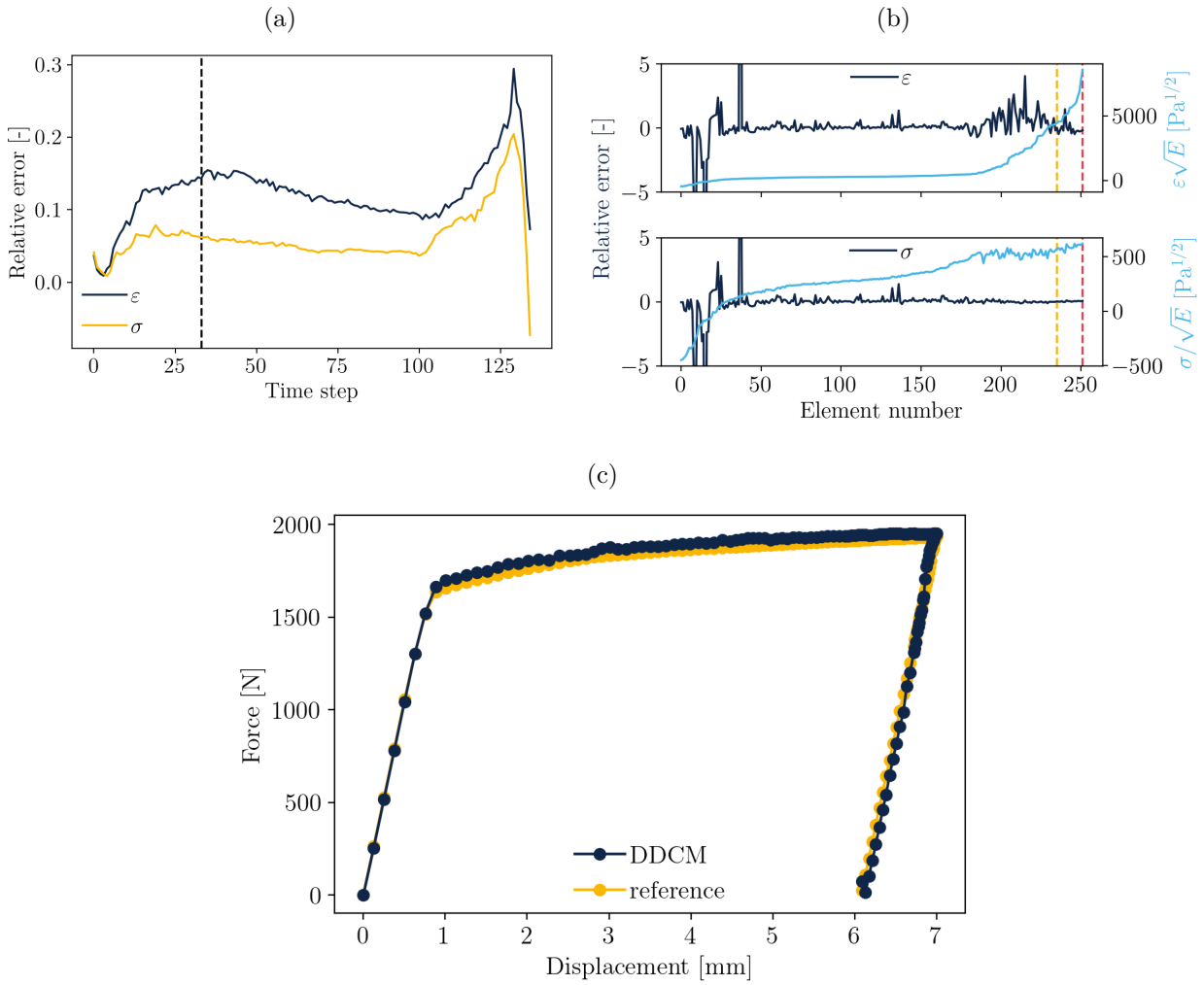

Figure 16: (a) Median relative errors between DDCM and projected reference mechanical states. (b) Relative errors at time step 33 (black dotted line on plot (a)) between DDCM and projected reference mechanical states, the yellow and red dotted lines refer to the elements highlighted on Fig. 12 (a). (c) Evolution of resulting force against displacement for the truss problem with predictor-corrector algorithm.

For trusses, the standard DDCM solver is strongly affected by the problem combinatorial complexity and can reach a spurious solution. Model-based computational mechanics usually circumvents such solutions by limiting the material behaviour to a tangent space and controlling loading increments. Within the DDCM framework, we take advantage of the material digraph to compute two local databases: the thermomechanically admissible database, as defined above, and a smaller iso-dissipation database. The first one is called dissipative database, while the latter is non-dissipative. An enhanced two-stage solver, composed of a prediction step with the non-dissipative local database and a correction step with the dissipative database, has been developed that exhibits promising results.

Fundamentally, there is *a priori* no theoretical obstacle to the extension of the method to full 3D elastoplasticity. However, the extension implies an increase of the constitutive space

dimensionality and a larger number of points is required to accurately represent the material behaviour. This problem is not specific to inelastic behaviours since it even arises in elasticity, and has been addressed, e.g. with adaptive approaches [Gorgogianni et al., 2023]. Additionally, the amount of possible dissipative paths will increase even faster and the material digraph will necessarily have to sample the possible transitions. An efficient description of the material digraph then becomes necessary, and graph theory provides a wide range of tools and algorithms that have not yet been explored in detail. In particular, a hierarchical representation of the digraph based on cliques, that play an important role in the approach, could be of use.

Nevertheless, once this operation is achieved, graph theory provides optimised algorithms for efficient graph search and database construction. For instance regarding local databases selection, an operation that has to be repeated at every integration point, a worst case scenario gives Dijkstra's algorithm (with Fibonacci heap) running time complexity in $O(|A| + |V| \times \log(|V|))$, with $|\bullet|$ the cardinal number of the sets of arcs A and vertices V. The reduction criteria yet restrict the graph search to the root's neighbourhood, decreasing time cost. Alternatively, a Breadth-First Search (BFS) algorithm with a complexity in $O(|A| + |V|)$ could be used to explore the vertices from one depth to the next, starting from the root (as opposed to Depth-First Search algorithm, which explores the complete branch before moving to the next). These numerical optimisations will be the subject of future work.

Acknowledgments

All authors gratefully acknowledge the financial support of the Deutsche Forschungsgemeinschaft (DFG) and French Agence Nationale de la Recherche (ANR) through the project "Direct Data-Driven Computational Mechanics for Anelastic Material Behaviours" (ANR-19-CE46-0012, RE 1057/47-1, project number 431386925) within the French-German Collaboration for Joint Projects in Natural, Life and Engineering (NLE) Sciences.

Bibliography

T. Bartel, M. Harnisch, B. Schweizer, and A. Menzel. A data-driven approach for plasticity using history surrogates: Theory and application in the context of truss structures. *Computer Methods in Applied Mechanics and Engineering*, 414:116138, Sept. 2023. ISSN 0045-7825. doi: 10.1016/j.cma.2023.116138.

A. Bondy and U. Murty. *Graph Theory*, volume 244 of *Graduate Texts in Mathematics*. Springer-Verlag, London, 2008. ISBN 978-1-84628-969-9.

A. Chrysochoos, V. Huon, F. Jourdan, J.-M. Muracciole, R. Peyroux, and B. Wattrisse. Use of Full-Field Digital Image Correlation and Infrared Thermography Measurements for the

Thermomechanical Analysis of Material Behaviour. *Strain*, 46(1):117–130, 2010. ISSN 1475-1305. doi: 10.1111/j.1475-1305.2009.00635.x.

K. Ciftci and K. Hackl. Model-free data-driven simulation of inelastic materials using structured data sets, tangent space information and transition rules. *Computational Mechanics*, 70(2): 425–435, Aug. 2022. ISSN 1432-0924. doi: 10.1007/s00466-022-02174-x.

M. Dalémat, M. Coret, A. Leygue, and E. Verron. Measuring stress field without constitutive equation. *Mechanics of Materials*, 136:103087, Sept. 2019. ISSN 0167-6636. doi: 10.1016/j.mechmat.2019.103087.

E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959. doi: 10.1007/BF01386390.

R. Eggersmann, T. Kirchdoerfer, S. Reese, L. Stainier, and M. Ortiz. Model-Free Data-Driven inelasticity. *Computer Methods in Applied Mechanics and Engineering*, 350:81–99, June 2019. ISSN 0045-7825. doi: 10.1016/j.cma.2019.02.016.

M. Flaschel, S. Kumar, and L. De Lorenzis. Discovering plasticity models without stress data. *npj Computational Materials*, 8(1):1–10, Apr. 2022. ISSN 2057-3960. doi: 10.1038/s41524-022-00752-4.

F. Ghavamian and A. Simone. Accelerating multiscale finite element simulations of history-dependent materials using a recurrent neural network. *Computer Methods in Applied Mechanics and Engineering*, 357:112594, Dec. 2019. ISSN 0045-7825. doi: 10.1016/j.cma.2019.112594.

J. D. Goddard. From Granular Matter to Generalized Continuum. In G. Capriz, P. M. Mariano, and P. Giovine, editors, *Mathematical Models of Granular Matter*, Lecture Notes in Mathematics, pages 1–22. Springer, Berlin, Heidelberg, 2008. ISBN 978-3-540-78277-3. doi: 10.1007/978-3-540-78277-3_1.

A. Gorgogianni, K. Karapiperis, L. Stainier, M. Ortiz, and J. E. Andrade. Adaptive goal-oriented data sampling in Data-Driven Computational Mechanics. *Computer Methods in Applied Mechanics and Engineering*, 409:115949, May 2023. ISSN 0045-7825. doi: 10.1016/j.cma.2023.115949.

F. Harary, R. Z. Norman, and D. Cartwright. *Structural models: an introduction to the theory of directed graphs*, volume IX. New York, Wiley, 1965.

Y. Kanno. Mixed-integer programming formulation of a data-driven solver in computational elasticity. *Optimization Letters*, 13(7):1505–1514, Oct. 2019. ISSN 1862-4472, 1862-4480. doi: 10.1007/s11590-019-01409-w.

K. Karapiperis, L. Stainier, M. Ortiz, and J. E. Andrade. Data-Driven multiscale modeling in mechanics. *Journal of the Mechanics and Physics of Solids*, 147:104239, Feb. 2021. ISSN 0022-5096. doi: 10.1016/j.jmps.2020.104239.

T. Kirchdoerfer and M. Ortiz. Data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 304:81–101, June 2016. ISSN 0045-7825. doi: 10.1016/j.cma.2016.02.001.

T. Kirchdoerfer and M. Ortiz. Data Driven Computing with noisy material data sets. *Computer Methods in Applied Mechanics and Engineering*, 326:622–641, Nov. 2017. ISSN 0045-7825. doi: 10.1016/j.cma.2017.07.039.

T. F. Korzeniowski and K. Weinberg. A multi-level method for data-driven finite element computations. *Computer Methods in Applied Mechanics and Engineering*, 379:113740, June 2021. ISSN 0045-7825. doi: 10.1016/j.cma.2021.113740.

R. Langlois, M. Coret, and J. Réthoré. Non-parametric stress field estimation for history-dependent materials: Application to ductile material exhibiting Piobert–Lüders localization bands. *Strain*, page e12410, Feb. 2022. ISSN 1475-1305. doi: 10.1111/str.12410.

A. Leygue, R. Seghir, J. Réthoré, M. Coret, E. Verron, and L. Stainier. Non-parametric material state field extraction from full field measurements. *Computational Mechanics*, 64(2):501–509, Aug. 2019. ISSN 0178-7675, 1432-0924. doi: 10.1007/s00466-019-01725-z.

J. Lubliner. *Plasticity Theory*. Dover edition, Jan. 2008. ISBN 978-0-486-46290-5.

D. K. N. Pham, N. Blal, and A. Gravouil. Tangent space Data Driven framework for elasto-plastic material behaviors. *Finite Elements in Analysis and Design*, 216:103895, Apr. 2023. ISSN 0168-874X. doi: 10.1016/j.finel.2022.103895.

A. Platzer. *Finite strain data-driven computational mechanics. : From tailored data to adaptive solvers for multiscale simulations*. PhD thesis, École Centrale de Nantes, Dec. 2020.

K. Poelstra, T. Bartel, and B. Schweizer. A data-driven framework for evolutionary problems in solid mechanics. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, n/a(n/a):e202100538, Nov. 2022. ISSN 1521-4001. doi: 10.1002/zamm.202100538.

M. Rigo. *Advanced Graph Theory and Combinatorics*. John Wiley & Sons, Nov. 2016. ISBN 978-1-119-05861-8.

R. Seghir, J.-F. Witz, L. Bodelot, E. Charkaluk, and P. Dufrénoy. An improved lagrangian thermography procedure for the quantification of the temperature fields within polycrystals. *Quantitative InfraRed Thermography Journal*, 10(1):74–95, June 2013. ISSN 1768-6733. doi: 10.1080/17686733.2013.785207.

K. Starkey, T. Hochrainer, and A. El-Azab. Development of mean-field continuum dislocation kinematics with junction reactions using de Rham currents and graph theory. *Journal of the Mechanics and Physics of Solids*, 158:104685, Jan. 2022. ISSN 0022-5096. doi: 10.1016/j.jmps.2021.104685.

G. Valdés-Alonzo. *Identification of material properties and phase distribution of heterogeneous materials through data-driven computational methods : Towards an enhaced constitutive space*. PhD thesis, École Centrale de Nantes ; Universitat Politècnica de Catalunya - BarcelonaTech, June 2022.

A. Vinel. *Characterization of the thermomechanical behaviour of metals for high strain-rates, using ultra-high speed imaging cameras*. PhD thesis, École Centrale de Nantes, Jan. 2022.

K. Weinberg, L. Stainier, S. Conti, and M. Ortiz. Data-Driven Games in Computational Mechanics. *Computer Methods in Applied Mechanics and Engineering*, 417:116399, 2023. doi: 10.1016/j.cma.2023.116399.