Unsupervised learning of history-dependent constitutive material laws with thermodynamically-consistent neural networks in the modified Constitutive Relation Error framework.

Antoine Benady, Emmanuel Baranger, Ludovic Chamoin

UNSUPERVISED LEARNING OF HISTORY-DEPENDENT CONSTITUTIVE MATERIAL LAWS WITH THERMODYNAMICALLY-CONSISTENT NEURAL NETWORKS IN THE MODIFIED CONSTITUTIVE RELATION ERROR FRAMEWORK

Antoine Benady
Université Paris-Saclay, CentraleSupélec
ENS Paris-Saclay, CNRS,
LMPS - Laboratoire de Mécanique Paris-Saclay,
4, Avenue des Sciences, 91190, Gif-sur-Yvette, France
antoine.benady@ens-paris-saclay.fr

Emmanuel Baranger
Université Paris-Saclay, CentraleSupélec
ENS Paris-Saclay, CNRS,
LMPS - Laboratoire de Mécanique Paris-Saclay,
4, Avenue des Sciences, 91190, Gif-sur-Yvette, France
emmanuel.baranger@ens-paris-saclay.fr

Ludovic Chamoin
Université Paris-Saclay, CentraleSupélec
ENS Paris-Saclay, CNRS,
LMPS - Laboratoire de Mécanique Paris-Saclay,
4, Avenue des Sciences, 91190, Gif-sur-Yvette, France

IUF - Institut Universitaire de France
1 rue Descartes, 75231, Paris Cedex 5 France
ludovic.chamoin@ens-paris-saclay.fr

ABSTRACT

This article proposes a consistent and general approach to train physics-augmented neural networks with observable data to enrich and represent nonlinear history-dependent material behaviors in terms of both state equations and evolution laws. In this learning strategy consistent with thermodynamics, the constitutive model is expressed using two potentials (free energy and dissipation potential) which are represented by input-convex neural networks, thus automatically satisfying the principles of thermodynamics. The neural network is trained thanks to an unsupervised procedure that does not rely on strain-stress pairs but needs only partial strain or displacement measurements inside the structure, moreover with uncertain boundary conditions. This method is based on the minimization of the modified Constitutive Relation Error functional, and it extends previous works on this error measure for neural networks to the case of history-dependent behaviors, which requires to design a specific minimization procedure. Given that neural networks for typical structural health monitoring applications often need to be trained online, there is here a significant emphasis placed on automatically and adaptively tuning crucial hyperparameters such as learning rate or weighting between losses.

The method is evaluated on elastoplastic and elastoviscoplastic test cases with synthetic data collected from optic fiber measurements or digital image correlation. It is shown that the method can properly learn hidden behaviors, achieves high robustness to noise level, and low sensitivity to user-defined hyperparameters

Keywords Constitutive modeling, Physics-augmented Neural Network, Unsupervised learning, Constitutive relation error, Evolution laws, Data assimilation

1 Introduction

Structural health monitoring is a significant concern in the field of engineering and has a wide-ranging impact on various structures such as wind turbines, aircrafts, and space structures. In this context, it has become crucial to implement techniques that can detect defects early on and track their growth, ensuring the integrity of structures equipped with sensors throughout their lifespan. To effectively combine the advantages of numerical simulations and physical measurements, there is a growing trend to develop novel processes that enable real-time, dynamic information exchange between the physical system and its corresponding numerical simulator (referred to as a virtual twin). This approach, known as Dynamic Data Driven Application Systems (DDDAS) [1, 2], aims to achieve the following objectives:

- continuously predict the evolution of the relevant physical phenomena and adjust the system accordingly;
- dynamically update the computational model by assimilating in situ measurements in real-time, for effective diagnosis and prognosis.

This trend benefits from the development over the past decades of new measurement tools such as optic fibers which enable high spatial resolution [3], as well as the significant improvement of physics-based constitutive modeling which represents a rich history of knowledge that is nowadays used in virtual twins [4]. The coupling between measured data and physics-based constitutive modeling is generally performed by using the data to identify the parameters of a constitutive model with given structure. Many methods have been developed for inverse problems in continuum mechanics (see for example [5, 6, 7]) to find the parameters of a constitutive model that match best to the observations. The modified Constitutive Relation Error (mCRE) [8] - also referred to as constitutive equation gap method (CEGM) [9] - is an interesting framework for inverse problems as it gives a strong physical sense through the modeling error term of the functional to minimize. The main idea of this framework is to deal with the reliability of information. Reliable information (*e.g.* mechanical equilibrium) is enforced in the process whereas unreliable information (*e.g.* measurements values or constitutive relation) is released. This hybrid approach, in the sense that it relies on both physics-based and data-based information, is a good compromise between purely data-based approaches (with high variance and low bias) which require high data quantity, and model-based approaches (with low variance and high bias) which require low data quantity. It also provides for an estimate of the mechanical state, by means of the computation of admissible fields, still in a hybrid manner. The mCRE is now a widely studied method that has been tested in many cases involving forced vibrations dynamics [8, 10, 11, 12], transient dynamics [13, 14], sequential data-assimilation [15, 16], and nonlinear material behavior [17, 18, 19]. The bias-aware mCRE framework is also known to be robust to corrupted [20] or noisy [21] measurements. The classical mCRE approach, like other parameter identification methods, is nevertheless based on a postulated model form. This kind of approach has the advantage of being physically interpretable but has the following drawbacks:

- dealing with constitutive modeling, the modeling process can be difficult in the case of complex behaviors;
- postulating a model form may imply having a large model bias, *i.e.* when the model form does not allow to describe the observed complexity correctly (although it is taken into account in the mCRE framework thanks to a modeling error term);
- once the model is fixed in an SHM context, enriching the constitutive model to take into account new mechanisms that could occur is not an easy task.

An idea to merge the modeling task with the parameters identification one is to replace the constitutive model with neural networks (which are known to be universal approximators [22]), thus getting rid of the above mentioned drawbacks. Indeed, the use of neural networks allows to not postulate any model form. Its use in constitutive modeling is now a growing trend, with first developments introduced in [23]. At this time, learning strain-stress relationships relied solely on data, neglecting any physical insight into the network architecture or loss function. Coupling deep learning and physical models has recently gained attention across scientific domains, as seen in [24, 25], helping to mitigate typical neural networks concerns such as physical consistency, generalization, and training difficulties [25]. These coupling techniques can be grouped into 3 families. First, one possibility introduced as "physics-informed" in [26] incorporates physical knowledge into the loss function, combining measurement deviations and penalties for non-physical outcomes. Second, the "physics-augmented" approach integrates physical constraints into the network architecture [27, 28, 29, 30, 31, 32] through frameworks anchored in the convexity of the free energy using Input-Convex Neural Networks (ICNN) from [33]. The third strategy is transfer learning which consists of injecting prior knowledge during the network initialization, thus increasing the data quantity and reducing the impact of random initialization. The NN-mCRE strategy proposed in [34] and extended here naturally integrates all these recent trends: the constitutive model is represented by an ICNN (initialized with a priori knowledge) that is trained in an unsupervised way by minimizing the mCRE functional. Note that neural networks and mCRE have also been coupled in [35] in a different context where the constitutive relation error concept is used to identify parameters of a linear elastic constitutive law.

One difference (among many others) between [34] and [35] is that in [34] the constitutive relation form is not given (it is replaced by a neural network) whereas in [35] a neural network outputs Young's modulus and Poisson's ratio of the linear elastic law. In the NN-mCRE version of the work proposed in [34], the constitutive models considered were nonlinear elastic, and the extension here aims at learning history-dependent dissipative models (elastoplastic and elastoviscoplastic).

A wide variety of works addressed the questions of representing history-dependent constitutive models with neural networks [36, 37, 38, 39, 40]. As is classically the case in Deep Learning, different types of neural networks are used such as:

- feed-forward neural networks. With this type of network, the inputs of the neural network are generally the strain increment and the postulated internal variables (see for examples [41, 42]);
- recurrent neural networks which are suited to deal with time series. One strong interest of such recurrent networks is the use of internal memories which can play the role of internal variables (see for examples [43, 44, 36, 45, 39]);
- time convolutional neural networks that are also suited for time series: a comparison between time convolutional neural networks and recurrent neural networks can be found in [46].

Moreover, as detailed in the following, the constitutive model should satisfy the thermodynamic principles. Several works have addressed the issue of enforcing thermodynamic constraints in the architecture or penalizing its violation in the loss function during the training phase. The advantage of enforcing the constraints in the architecture is that they are automatically satisfied in the inference phase, whereas it is not the case when constraints are put in the loss function during training. In [47] a multilayer perceptron predicting the stress increment derived from a thermodynamic potential is proposed. The mechanical dissipation is then computed and its positivity constraint is put in the loss function used for training. The dissipation can also be computed in the inference phase to check whether the Clausius-Duhem inequality is satisfied or not. In [48] the stress is also derived from a thermodynamic potential but the increment of internal variable is given by a recurrent neural network. The main advantage is that this approach does not require choosing the internal variables as they are obtained from the internal memories of the recurrent cells. The second principle is also taken into account in the loss function. In [30] thermodynamic potentials are predicted with the input-convex architecture proposed in [33] (with internal variables as input), thus automatically satisfying Clausius-Duhem's inequality. In this method, the resolution of the evolution laws is not performed by the neural network but the nonlinear evolution laws are solved with the Newton-Raphson algorithm. Another possibility to define a thermodynamically-consistent architecture can be found in [49], in which the metriplectic structure of dissipative Hamiltonian systems is implemented in the form of the so-called General Equation for the Non-Equilibrium Reversible-Irreversible Coupling (GENERIC)[50].

The previously mentioned approaches focus on the architecture of the neural networks, whereas the focus of this contribution is on the choice of the loss function and its specific minimization in an unsupervised way. This work is motivated by the fact that in practice, acquiring dataset of strain-stress (or strain-potential) is not feasible, so these methods are not suited for cases where one wants to learn a new constitutive model, or are limited to cases with simple loading paths such as tension/compression. Furthermore, the amount of available data is rather scarce in practical applications (full-field measurement hardly achievable).

The extension of NN-mCRE [34] to take into account history-dependent behaviors requires a specific minimization algorithm presented in the following. The present paper aims to address a case where data are measurable, thus dealing with a limited database. In this context, enough a priori knowledge is needed so the internal variables are chosen a priori. The constitutive model is searched within the framework of the Generalized Standard Material [51]. The measurement noise level and the boundary conditions are assumed to be known, even though the mCRE framework is suited to deal with uncertain boundary conditions. The method is evaluated on elastoplastic and elastoviscoplastic 2D test cases with synthetic measurements (either strain measurements obtained from optic fiber sensors or full-field displacement obtained from digital image correlation).

The reminder of the paper is organized as follows; Section 2 defines the thermodynamic framework as well as the problem tackled by the minimization of the mCRE. Section 3 addresses the questions raised by the use of neural networks to describe the material behavior. The method is evaluated on different test cases in Section 4 by means of several criteria such as the accuracy of the learned model, evolution of the loss function during the training, the noise robustness, the localization of modeling error and the relevance of automatic hyperparameters tuning rules. Eventually, conclusions and prospects are drawn in Section 5.

2 The modified Constitutive Relation Error concept for history-dependent dissipative models

2.1 Thermodynamic framework

This section aims to briefly describe the thermodynamically consistent framework of the Generalized Standard Material [51] used in this article. In such a formulation, the material behavior is described in terms of state equations and evolution laws involving convex potentials and internal variables.

The state equations are derived from the Helmholtz free energy used as a thermodynamic potential $\psi(\mathbb{\varepsilon}, \mathbb{\varepsilon}_p, \mathbf{X})$, where the total linearized strain tensor $\mathbb{\varepsilon}$ is the sum of the elastic strain $\mathbb{\varepsilon}_e$ and the plastic strain $\mathbb{\varepsilon}_p$ ($\mathbb{\varepsilon} = \mathbb{\varepsilon}_e + \mathbb{\varepsilon}_p$), and $\mathbf{X}$ is a vector collecting internal variables X_k $(k = 1, \dots, K)$ introduced to represent other phenomena (such as hardening for example). The use of a convex function ψ automatically satisfies the Clausius-Duhem inequality. A convex, non-negative and zero at origin dissipation potential $\varphi(\dot{\mathbb{\varepsilon}}_p, \dot{\mathbf{X}})$ may be postulated for the definition of evolution laws, to guarantee the positivity of the mechanical dissipation $\mathcal{D} = (\mathbb{\sigma} : \dot{\mathbb{\varepsilon}}_p - \mathbf{Y} : \dot{\mathbf{X}})$, with $\mathbb{\sigma}$ the Cauchy stress tensor and $\mathbf{Y}$ the vector collecting thermodynamics forces $Y_k = \partial\psi/\partial X_k$.

The dual potentials that might be more convenient to express the constitutive model are defined with the Legendre-Fenchel transform :

$$\psi^*(\mathbf{s}) = \sup_{\mathbf{e}_e}[\mathbf{s}.\mathbf{e}_e - \psi(\mathbf{e}_e)] \quad \text{and} \quad \varphi^*(\mathbf{s}) = \sup_{\dot{\mathbf{e}}_p}[\mathbf{s}.\dot{\mathbf{e}}_p - \varphi(\dot{\mathbf{e}}_p)] \tag{1}$$

with $\mathbf{e}_e = [\mathbb{\varepsilon}_e, \mathbf{X}]$, $\mathbf{e}_p = [\mathbb{\varepsilon}_p, -\mathbf{X}]$ the global vectors of flux variables, and $\mathbf{s} = [\mathbb{\sigma}, \mathbf{Y}]$ the global vector of thermodynamic forces.

For a rate-independent behavior, φ^* is not differentiable and the yield criterion function $f(\mathbb{\sigma})$ is introduced, as well as the Karush–Kuhn–Tucker (KKT) conditions [52]. In this case :

$$\varphi^*(\mathbb{\sigma}) = \mathbf{I}(f) = \begin{cases} 0 \text{ if } f < 0 \\ +\infty \text{ if } f = 0 \end{cases} \tag{2}$$

The potential ψ, φ, ψ^* and φ^* are involved in the definition of the Constitutive Relation Error (CRE) recalled in Section 2.3.

2.2 Problem definition

To define the problem notations, let us consider a body in an initial configuration $\Omega \subset \mathbb{R}^d (d = 1, 2, 3)$ with boundary $\partial\Omega$ and isothermal environment, observed for a period $[0, T]$ under the small strain assumptions. Dirichlet boundary conditions are imposed on $\partial\Omega_1 \subset \partial\Omega$ by means of a time-dependent displacement field $\mathbf{u}_d$. Neumann boundary conditions are prescribed on $\partial\Omega_2 \subset \partial\Omega$ by means of a time-dependent traction field $\mathbf{f}_s^d$. A time-dependent body force field $\mathbf{f}_v^d$ may also be prescribed in Ω. Additionally, noisy strain measurements $\mathbb{\varepsilon}_{obs}$ (in the case of observations from optic fibers) or displacement measurements $\mathbf{u}_{obs}$ (in the case of digital image correlation) are available.

The solution to the direct mechanical problem is the set of the variables $(\mathbf{e}_e, \mathbf{e}_p, \mathbf{s})$ satisfying the three following groups of equations for each time $t \in [0, T]$:

- **kinematic admissibility** defines the space $\mathcal{U}_{ad}$ of displacement fields $\mathbf{u}$ satisfying the Dirichlet boundary conditions:
$$\mathbf{u}_{|\partial\Omega_1} = \mathbf{u}_d \tag{3}$$
- **static admissibility** defines the space $\mathcal{S}_{ad}$ of stress fields satisfying the equilibrium:
$$\int_\Omega \mathbf{s} : \mathbf{e}(\mathbf{v}) = \int_\Omega \mathbf{f}_d^v.\mathbf{v} + \int_{\partial\Omega_2} \mathbf{f}_d^s.\mathbf{v} \quad \forall \mathbf{v} \in \mathcal{U}_0 \tag{4}$$
with $\mathbf{e} = \mathbf{e}_e + \mathbf{e}_p$, and $\mathcal{U}_0$ the space of kinematic admissibility with homogeneous Dirichlet conditions.
- **constitutive behavior**:

  - a set of state equations:

$$\mathbf{s} = \frac{\partial\psi}{\partial\mathbf{e}_e} \tag{5}$$

- a set of evolution laws:

$$\mathbf{s} = \frac{\partial \varphi}{\partial \dot{\mathbf{e}}_p} \tag{6}$$

The potentials ψ and φ are parametrized with some parameters $\mathbf{p}$.

The minimization of the mCRE aims to identify the parameters $\mathbf{p}$ of constitutive relations that fit the best to experimental data. Here Dirichlet and Neumann boundary conditions are assumed to be known (even though the framework extends to unreliable boundary conditions [53]), whereas the observations $\mathbb{e}_{obs}$ or $\mathbf{u}_{obs}$ are affected by measurement noise.

2.3 CRE concept

The CRE concept was introduced in the 70s in the context of finite element verification [54] (*i.e.* a posteriori error estimation and mesh adaptation, see [55] for details). The general idea of this error measure is to split the equations of Section 2.2 following the reliability of information. The constitutive relation is considered as the unreliable equation and given an admissible pair $(\hat{\mathbf{u}}, \hat{\mathbb{\sigma}}) \in (\mathcal{U}_{ad} \times \mathcal{S}_{ad})$, the CRE is defined as follows for elasticity problems:

$$\mathcal{E}^2_{CRE}(\hat{\mathbf{u}}, \hat{\mathbb{\sigma}}) = \int_\Omega \Big(\psi(\hat{\mathbf{u}}) + \psi^*(\hat{\mathbb{\sigma}}) - \hat{\mathbb{\sigma}} : \mathbb{e}(\hat{\mathbf{u}}) \Big) d\Omega \tag{7}$$

With quadratic potential ψ describing the Hooke law ($\psi(\mathbb{e}) = \frac{1}{2}\mathbb{e} : \mathbf{K} : \mathbb{e}$ and $\psi^*(\mathbb{\sigma}) = \frac{1}{2}\mathbb{\sigma} : \mathbf{K^{-1}} : \mathbb{\sigma}$), the linear elasticity constitutive relation is $\mathbb{\sigma} = \mathbf{K} : \mathbb{e}(\mathbf{u})$ and the CRE is written

$$\mathcal{E}^2_{CRE}(\hat{\mathbf{u}}, \hat{\mathbb{\sigma}}) = \int_\Omega (\hat{\mathbb{\sigma}} - \mathbf{K}\mathbb{e}(\hat{\mathbf{u}})) : \mathbf{K}^{-1} : (\hat{\mathbb{\sigma}} - \mathbf{K}\mathbb{e}(\hat{\mathbf{u}})) d\Omega = \int_\Omega ||\hat{\mathbb{\sigma}} - \mathbf{K}\mathbb{e}(\hat{\mathbf{u}})||^2_{\mathbf{K}^{-1}} d\Omega \tag{8}$$

where $||\bullet||_{\mathbf{K}^{-1}}$ is the energy norm on stress fields. From (8), one can observe that the CRE is an indicator of how much an admissible couple does not satisfy the constitutive relation.

Later, still in the context of a posteriori error estimation, the CRE concept was adapted for nonlinear history-dependent problems [56, 57]. In this context, the CRE reads for and admissible solution $\hat{s} = (\hat{\mathbb{e}}_e, \hat{\mathbb{e}}_p, \hat{\mathbb{\sigma}}, \hat{\mathbf{X}}, \hat{\mathbf{Y}})$:

$$\mathcal{E}^2_{CRE}(\hat{s}) = \int_0^T \int_\Omega \eta_\psi(\hat{\mathbb{e}}_e, \hat{\mathbb{\sigma}}, \hat{\mathbf{X}}, \hat{\mathbf{Y}}) d\Omega dt + \int_0^T \int_0^t \int_\Omega \eta_\varphi(\dot{\hat{\mathbb{e}}}_p, \hat{\mathbb{\sigma}}, \dot{\hat{\mathbf{X}}}, \hat{\mathbf{Y}}) d\Omega ds dt \tag{9}$$

with

$$\eta_\psi(\hat{\mathbb{e}}_e, \hat{\mathbb{\sigma}}, \hat{\mathbf{X}}, \hat{\mathbf{Y}}) = \psi(\hat{\mathbb{e}}_e, \hat{\mathbf{X}}) + \psi^*(\hat{\mathbb{\sigma}}, \hat{\mathbf{Y}}) - \hat{\mathbb{\sigma}} : \hat{\mathbb{e}}_e - \hat{\mathbf{X}}.\hat{\mathbf{Y}} \tag{10}$$

$$\eta_\varphi(\dot{\hat{\mathbb{e}}}_p, \hat{\mathbb{\sigma}}, \dot{\hat{\mathbf{X}}}, \hat{\mathbf{Y}}) = \varphi(\dot{\hat{\mathbb{e}}}_p, -\dot{\hat{\mathbf{X}}}) + \varphi^*(\hat{\mathbb{\sigma}}, \hat{\mathbf{Y}}) - \hat{\mathbb{\sigma}} : \dot{\hat{\mathbb{e}}}_p + \dot{\hat{\mathbf{X}}}.\hat{\mathbf{Y}} \tag{11}$$

2.4 Extension of the CRE concept for parameter identification: the modified Constitutive Relation Error (mCRE)

The CRE concept was adapted in the 90s to tackle inverse problems [8]; parameters of the constitutive model were found so that they minimize the CRE. In the first idea of the method, observations were directly included in the definition of the admissible space, leading to weak robustness to measurement noise. To improve this robustness, the mCRE [58, 17, 10] procedure does not impose the observations in the admissible space. Only reliable information is enforced by construction whereas unreliable information is released. Recently the mCRE concept has been extended to the case of history-dependent problems in [18, 19]. The present paper is based on the mCRE framework developed in [18, 19] and is adapted to the case where the constitutive model is described by neural networks. This section recalls the basics of the mCRE for history-dependent problems.

The inverse problem consists of finding the optimal parameters $\mathbf{p}_{opt}$ (involved in the thermodynamic potentials) such that:

$$\mathbf{p}_{opt} = \underset{\mathbf{p}}{\operatorname{argmin}} \Big[\min_{\hat{s} \in \mathcal{A}_d} \mathcal{E}^2_{mCRE}(\hat{s} \ ; \ \mathbf{p}) \Big] \tag{12}$$

with

$$\mathcal{E}^2_{mCRE}(\hat{s}\ ;\ \mathbf{p}) = \mathcal{E}^2_{CRE}(\hat{s}\ ;\ \mathbf{p}) + \frac{\alpha}{2}\int_0^T ||\Pi\mathbf{u} - \mathbf{u}_{obs}||^2 dt \tag{13}$$

where $\mathcal{A}_d = (\mathcal{U}_{ad} \times \mathcal{S}_{ad})$ (even though it is possible to consider loading as uncertain information, see [53]), α is a scaling factor, and Π is a projector of $\mathbf{u}$ on the sensing quantities.

The minimization of the mCRE is performed with an iterative process in which, at each iteration :

- in Step 1, an optimal admissible solution $\hat{s}^{(n+1)}$ is computed for the current parameters $\mathbf{p}^{(n)}$ such that :

$$\hat{s}^{(n+1)} = \underset{s\in\mathcal{A}_d}{\operatorname{argmin}} \left[\mathcal{E}^2_{mCRE}(s\ ;\ \mathbf{p}^{(n)})\right] \tag{14}$$

- in Step 2, the parameters of the constitutive model are updated following a gradient descent step to get $\mathbf{p}^{(n+1)}$.

Sections 2.4.1 and 2.4.2 respectively detail the first and second steps of this iterative minimization.

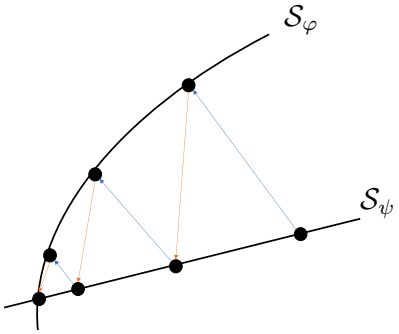
2.4.1 First step: computing the optimal admissible solution for given parameters

The first step, which is the most expensive one regarding computation time, is performed with a strategy similar to the one used in [18, 19]. This strategy is inspired by the LATIN method [59], which is non-incremental (*i.e.* global in time) and is well-suited to the mathematical structure of the mCRE. The choice made here is to split the mCRE into two positive parts $\mathcal{E}^2_\psi(\mathbb{e}_e, \mathbb{\sigma}, \mathbf{X}, \mathbf{Y})$ and $\mathcal{E}^2_\varphi(\dot{\mathbb{e}}_p, \mathbb{\sigma}, \dot{\mathbf{X}}, \mathbf{Y})$ defined by :

$$\mathcal{E}^2_\psi(\hat{\mathbb{e}}_e, \hat{\mathbb{\sigma}}, \hat{\mathbf{X}}, \hat{\mathbf{Y}}) = \int_0^T \int_\Omega \eta_\psi(\hat{\mathbb{e}}_e, \hat{\mathbb{\sigma}}, \hat{\mathbf{X}}, \hat{\mathbf{Y}}) d\Omega dt + \frac{\alpha}{2}\int_0^T ||\Pi\hat{\mathbf{u}} - \mathbf{u}_{obs}||^2 dt \tag{15}$$

$$\mathcal{E}^2_\varphi(\dot{\hat{\mathbb{e}}}_p, \hat{\mathbb{\sigma}}, \dot{\hat{\mathbf{X}}}, \hat{\mathbf{Y}}) = \int_0^T \int_0^t \int_\Omega \eta_\varphi(\dot{\hat{\mathbb{e}}}_p, \hat{\mathbb{\sigma}}, \dot{\hat{\mathbf{X}}}, \hat{\mathbf{Y}}) d\Omega ds dt \tag{16}$$

This separation into two positive parts enables minimizing each term alternatively. Equation (15) is a compromise between the residual on the state equations and the discrepancy with measurements. Its minimization is a linear (because elasticity is assumed to be linear here) and global in space problem. The minimization of (16) is local in space and corresponds to the integration of evolution laws, performed here with an Euler scheme. The general philosophy of Step 1 minimization is illustrated in Figure 1. $\mathcal{S}_\psi$ (resp. $\mathcal{S}_\varphi$) is the space of variables searched in the global (resp. local) step that minimizes the quantity $\mathcal{E}_\psi$ (resp $\mathcal{E}_\varphi$).



Local step: minimization of $\mathcal{E}_\varphi$

Searching for $\mathbf{X_u}, \mathbf{Y_u}, \mathbb{e}_{e,\mathbf{u}}, \mathbb{e}_{p,\mathbf{u}}, \mathbb{\sigma}_\mathbf{u}, \mathbf{X_v}, \mathbf{Y_v}, \mathbb{e}_{e,\mathbf{v}}, \mathbb{e}_{e,\mathbf{v}}, \mathbb{\sigma}_\mathbf{v}$
$\mathbb{e}(\mathbf{u}), \mathbb{e}(\mathbf{v})$ frozen.

Global step: minimization of $\mathcal{E}_\psi$

Searching for $\mathbf{u}, \mathbb{e}_{e,\mathbf{u}}, \mathbb{\sigma}_\mathbf{u}, \mathbf{v}, \mathbb{e}_{e,\mathbf{v}}, \mathbb{\sigma}_\mathbf{v}$
Internal variables $\mathbf{X_u}, \mathbf{Y_u}, \mathbb{e}_{p,\mathbf{u}}, \mathbf{X_v}, \mathbf{Y_v}, \mathbb{e}_{p,\mathbf{v}}$ frozen.

Figure 1: Illustration of the LATIN-inspired scheme for Step 1 minimization of the mCRE.

Global step (minimization of $\mathcal{E}^2_\psi$)

In this minimization step, $\mathcal{E}^2_\psi$ is minimized under the admissiblity constraint $(\mathbf{u}, \mathbb{\sigma}) \in (\mathcal{U}_{ad} \times \mathcal{S}_{ad})$. The kinematic admissibility is enforced in the search space, in which the discretization of $\mathbf{u}$ is split into the imposed and free degrees of freedom. The static admissibility of $\mathbb{\sigma}$ is imposed through a Lagrangian :

$$\mathcal{L}(\mathbb{e}_e, \mathbb{\sigma}, \mathbf{X}, \mathbf{Y}, \boldsymbol{\lambda}) = \mathcal{E}^2_\psi(\mathbb{e}_e, \mathbb{\sigma}, \mathbf{X}, \mathbf{Y}) - \int_0^T \left[\int_\Omega \mathbb{\sigma} : \mathbb{e}(\boldsymbol{\lambda}) - \int_\Omega \mathbf{f}^d_v.\boldsymbol{\lambda} - \int_{\partial\Omega_2} \mathbf{f}^d_s.\boldsymbol{\lambda}\right] \tag{17}$$

In the expression of $\mathcal{E}^2_\psi$, the quantity η_ψ defined in (10) involves a dual Legendre-Fenchel transform, defined in (1), which is not convenient in practice. Therefore, a displacement field $\mathbf{v}$ (defined up to a rigid body motion) is introduced by duality such that:

$$\boldsymbol{\sigma} \in \mathcal{S}_{ad}, \quad \boldsymbol{\sigma} = \frac{\partial \psi}{\partial \boldsymbol{\varepsilon}_e}\Big|_{\boldsymbol{\varepsilon}_{e,\mathbf{v}}} \tag{18}$$

For a given $\boldsymbol{\sigma}$, $\boldsymbol{\varepsilon}_{e,\mathbf{v}}$ is the strain providing for the supremum in the definition of the Legendre-Fenchel transform. The variables associated with the displacement $\mathbf{v}$ (resp. $\mathbf{u}$) are denoted $s_\mathbf{v} = (\boldsymbol{\varepsilon}_{e,\mathbf{v}}, \boldsymbol{\varepsilon}_{p,\mathbf{v}}, \boldsymbol{\sigma}_\mathbf{v}, \mathbf{X}_\mathbf{v}, \mathbf{Y}_\mathbf{v})$ (resp. $s_\mathbf{u} = (\boldsymbol{\varepsilon}_{e,\mathbf{u}}, \boldsymbol{\varepsilon}_{p,\mathbf{u}}, \boldsymbol{\sigma}_\mathbf{u}, \mathbf{X}_\mathbf{u}, \mathbf{Y}_\mathbf{u})$).

Replacing the Legendre-Fenchel expression involved in (10), (17) leads to:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\varepsilon}_e, \boldsymbol{\sigma}, \mathbf{X}, \mathbf{Y}, \boldsymbol{\lambda}) = & \int_0^T \int_\Omega \Big[\psi(\boldsymbol{\varepsilon}_{e,\mathbf{u}}, \mathbf{X}_\mathbf{u}) - \psi(\boldsymbol{\varepsilon}_{e,\mathbf{v}}, \mathbf{X}_\mathbf{v}) - \boldsymbol{\sigma}_\mathbf{v} : (\boldsymbol{\varepsilon}_{e,\mathbf{v}} - \boldsymbol{\varepsilon}_{e,\mathbf{u}}) - \mathbf{Y}_\mathbf{v}(\mathbf{X}_\mathbf{v} - \mathbf{X}_\mathbf{u})\Big] \\ & + \frac{\alpha}{2} \int_0^T ||\Pi \mathbf{u} - \mathbf{u}_{obs}||^2 - \int_0^T \Big[\int_\Omega \boldsymbol{\sigma} : \boldsymbol{\varepsilon}(\boldsymbol{\lambda}) - \int_\Omega \mathbf{f}^d_v . \boldsymbol{\lambda} - \int_{\partial\Omega_2} \mathbf{f}^d_s . \boldsymbol{\lambda}\Big]\end{aligned} \tag{19}$$

During the global step, internal variables $\mathbf{X}_\mathbf{u}, \mathbf{Y}_\mathbf{u}, \boldsymbol{\varepsilon}_{p,\mathbf{u}}, \mathbf{X}_\mathbf{v}, \mathbf{Y}_\mathbf{v}, \boldsymbol{\varepsilon}_{p,\mathbf{v}}$ are frozen to the value obtained at the last local step. The stationarity of the Lagrangian is found with a Newton scheme (see [34]). In the case of linear elasticity (which will be assumed in the following for the sake of simplicity), this Newton scheme converges in one iteration.

Local step (minimization of $\mathcal{E}^2_\varphi$)

In this step, the minimization of $\mathcal{E}^2_\varphi$ corresponds to the integration of evolution laws at each Gauss point (local step). This step can be parallelized. The integration of the evolution laws is performed with an Euler scheme with the total strains $\boldsymbol{\varepsilon}(\mathbf{u})$ and $\boldsymbol{\varepsilon}(\mathbf{v})$ frozen. The initial conditions on internal variables are enforced for the first time step. As mentioned in Section 2.1 dealing with the thermodynamic framework, in the rate-independent case the dissipation potential is the indicator of a convex domain and is not differentiable, whereas in the rate-dependent case, this potential is differentiable. This leads to different formulations in the integration of evolution laws. The local steps are detailed for each case in Section 4.

Stopping criterion for Step 1

The stopping criterion used for Step 1 (alternation of local and global steps) needs to be defined. This minimization is a fixed point algorithm, so the stopping criterion is defined regarding the stagnation of the CRE between two successive local and global steps. The tolerance is defined by the user and its influence will be discussed in the section dedicated to the automatic tuning of the learning rate. The last step performed needs to be the global step so that the solution of the minimization is statically admissible.

Restart strategy

After an update of the model parameters, the new Step 1 is initialized with the solutions $s_\mathbf{u}$, $s_\mathbf{v}$ obtained at the previous Step 1. This restart strategy significantly reduces the number of iterations required for convergence, thus reducing computation time.

2.4.2 Second step: update of parameters

This step consists of the updating of parameters $\mathbf{p}$ with a gradient descent step :

$$\mathbf{p}^{(n+1)} = \mathbf{p}^{(n)} - l_r \frac{\partial \mathcal{E}^2_{mCRE}(\hat{s}^{(n+1)} \,;\, \mathbf{p}^{(n)})}{\partial \mathbf{p}} \tag{20}$$

with $\hat{s}^{(n+1)}$ the solution obtained at the end of Step 1 of the $n+1$ iteration of the mCRE minimization.

To summarize this section and introduce the next one, Figure 2 illustrates the general methodology for further training the physics-augmented neural network with the mCRE.

3 Training of thermodynamically-consistent neural networks with the mCRE framework

The previous section has recalled basics on the minimization of the modified Constitutive Relation Error in the case of a nonlinear constitutive model involving evolution laws. This procedure is suited for parameter identification of a given

Minimization of mCRE:

$$\mathbf{p}_{opt} = \underset{\mathbf{p}}{\operatorname{argmin}} \left[\min_{\hat{s} \in \mathcal{A}_d} \mathcal{E}^2_{mCRE}(\hat{s}\ ;\ \mathbf{p}) \right]$$

While $\mathcal{E}^2_{CRE} < \epsilon_{stop}$:

Step 1: $\hat{s}^{(n+1)} = \underset{s \in \mathcal{A}_d}{\operatorname{argmin}} \left[\mathcal{E}^2_{mCRE}(s\ ;\ \mathbf{p}^{(n)}) \right]$

Elastic initialization for the first iteration or restart with the solution from last iteration

While stopping criterion is not reached :

Local step: Integration of evolution laws. Minimization of

$$\mathcal{E}^2_{\varphi}(\dot{\hat{\varepsilon}}_p, \hat{\sigma}, \dot{\hat{\mathbf{X}}}, \hat{\mathbf{Y}}) = \int_0^T \int_0^t \int_\Omega (\varphi(\dot{\hat{\varepsilon}}_p, -\dot{\hat{\mathbf{X}}}) + \varphi^*(\hat{\sigma}, \hat{\mathbf{Y}}) - \hat{\sigma} : \dot{\hat{\varepsilon}}_p + \dot{\hat{\mathbf{X}}}.\hat{\mathbf{Y}}) d\Omega ds dt$$

Global step: Global equilibrium

$$\mathcal{E}^2_{\psi}(\hat{\varepsilon}_e, \hat{\sigma}, \hat{\mathbf{X}}, \hat{\mathbf{Y}}) = \int_0^T \int_\Omega (\psi(\hat{\varepsilon}_e, \hat{\mathbf{X}}) + \psi^*(\hat{\sigma}, \hat{\mathbf{Y}}) - \hat{\sigma} : \hat{\varepsilon}_e - \hat{\mathbf{X}}.\hat{\mathbf{Y}}) d\Omega dt + \frac{\alpha}{2} \int_0^T ||\Pi \hat{\mathbf{u}} - \mathbf{u}_{obs}||^2 dt$$

- Kinematic admissibility imposed by the searched space
- Static admissibility imposed by Lagrangian

Step 2:

$$\mathbf{p}^{(n+1)} = \mathbf{p}^{(n)} - l_r \frac{\partial \mathcal{E}^2_{mCRE}(\hat{s}^{(n+1)}\ ;\ \mathbf{p}^{(n)})}{\partial \mathbf{p}}$$

- Gradient computed with adjoint state method

Figure 2: Description of the method developed.

constitutive model. In addition, it may be used for neural network training (*i.e.* finding parameters $\mathbf{p}$ containing weights and biases $(\mathbf{W}, \mathbf{b})$) when the constitutive model is described by a neural network.

Representing a constitutive model by a neural network enables to release the form of the constitutive relation: potentials ψ and φ are searched in the space of functions that satisfy physical requirements defined in Section 3.1. Yet several questions and difficulties emerge when the constitutive model is described by a neural network. On the one hand, the question of consistency in the inference phase with respect to physical requirements is addressed in Section 3.1. On the other hand, the important number of parameters to find compared to the case of a given constitutive model form makes the optimization task [60] more difficult. Specifically, Deep Learning is known to be sensitive to user-defined hyperparameters such as learning rate, number of epochs, batch size, etc. In the present context of DDDAS, in which the hybrid twin interacts with the physical structure without human intervention, it is not possible to imagine a strategy based on human hyperparameter tuning. Thus, a strategy to automatically and adaptively select suitable hyperparameters is proposed in Section 3.2.

3.1 Constitutive model described by a thermodynamically-consistent neural network

In Section 2.1, a thermodynamic framework has been defined for constitutive modeling. In the following, neural networks which satisfy this framework are presented. As a reminder, the constitutive behavior is described by means

of two potentials: the Helmholtz free energy ψ and the dissipation pseudo-potential φ. To automatically satisfy thermodynamic principles, it is sufficient to assume ψ to be convex and φ to be convex, non-negative and zero at the origin.

According to the reliability of the initial guess on the constitutive model, either the whole behavior can be described by neural networks, or only some parts, as it is done in [30] in a context of limited data. In the case where some parts are well-known - such as elasticity - it is possible to enforce the form of the known part of the model. For example, function ψ can be split into $\psi = \psi_e(\mathbb{e}_e) + \psi_p(\mathbb{e}_p)$, in which $\psi_e = \frac{1}{2}\mathbb{e}_e : \mathbf{K} : \mathbb{e}_\mathbf{e}$ and ψ_p is represented by a neural network. In this example, the proposed mCRE framework can simultaneously identify parameters of ψ_e and the weights and biases of the neural network describing ψ_p.

In both cases, it is possible to enforce positivity, convexity, and zero at origin constraints in the neural network. The general idea is described in the following and details can be found in [30, 31]. Positivity is enforced through the use of positive activation functions. Concerning convexity, the input convex neural network (ICNN) architecture proposed in [33] is employed. This architecture uses convex non-decreasing activation functions and positivity constraints on intermediate weights. As the composition of a convex and convex non-decreasing function is convex and the sum of convex functions is also convex, this architecture guarantees convexity. Finally, the output of the network is corrected by subtracting its value in zero, so that the output is zero when the input is zero.

3.2 Hyperparameters automatic tuning strategy

In this section, the strategy for automatic tuning of the hyperparameters is presented. The relevance of these rules is evaluated in Section 4. All the rules have been designed on a test case (with different target model and geometry which are not presented here) and evaluated on the test cases presented in the paper. The following details a physics-guided initialization, a Morozov-based automatic tuning of the weighting between losses, an empirical adaptive learning rate rule, and a CRE-based stopping criterion used to avoid the predefined choice of the number of epochs. Figure 4 provides an idea of the general strategy for updating hyperparameters during training.

Physics-guided initialization with a priori knowledge

Here, the initialization is a critical issue. Previous work with the mCRE for parameter identification has shown the importance of initializing the parameter to identify relatively close to the true value [53]. Additionally, optimization methods for Deep Learning are also known to be sensitive to initialization [60]. A common approach is to initialize the neural networks randomly following different distributions such as the one used in [60] (known as "Xavier uniform" and "Xavier normal"), in [61] (known as Kaiming initialization) or in [62]. These random initiations are sometimes associated with a normalization of input data in order to avoid exploding and vanishing gradients. Some approaches even train several neural networks with different initializations and choose the best after training [29].

Yet, when dealing with constitutive modeling, and especially when the internal variables are chosen a priori, an initial idea can be available such as is the case in this work. Therefore, before training within the mCRE framework, the network is initialized following an initial guess. This initialization can be for example performed by means of a classical supervised training.

Morozov-based criterion for automatic tuning of weighting between losses

The mCRE framework lies within the multiple loss optimization problem scope, in which a frequent concern is the choice of the weighting between losses [63, 64]. More specifically dealing with the mCRE, the tuning of the parameter α is known to be an important question to solve when using the method [65, 66]. The mCRE functional provides an interesting physical sense that can be used to tune α. The idea of the tuning is that the model should be updated until the gap between prediction and observations reaches the value of the noise level. This requires the hypothesis of an a priori known noise level. The CRE term thus informs on the quality of the model. Hence, the field $\mathbf{u}$ obtained at Step 1 should get as close as possible to the measure, but should not fit the noise. To quantify this criterion, α is re-written:

$$\alpha = \alpha' \frac{1}{n_{obs}\sigma^2}$$

where n_{obs} is the number of observations and σ the standard deviation of the measurement noise. In order for $\mathbf{u}$ not to fit the observations below the noise level, the order of magnitude of $\frac{1}{n_{obs}\sigma^2}||\Pi\mathbf{u} - \mathbf{u}_{obs}||^2$ should be 1, as stated in the Morozov criterion [67]. In the version of this work dedicated to state laws [34], α was tuned at each epoch using a dichotomy. The computation time needed for the tuning of α was high (because it needs to re-perform Step 1 for each value of α tested) but still reasonable when dealing with state laws. On the contrary, when dealing with evolution laws, the computation time prohibits the use of this strategy. The idea here is to select an initial value of α, perform the

training until convergence of the normalized CRE, and then progressively update the value of α. As exploring the full range of possible values of α is prohibited by the computation time, the value of α is thus progressively increased (resp. decreased) if the model is updated above (resp. below) the noise level until one of the following conditions is met:

- the Morozov criterion is satisfied;
- or the normalized CRE is far from the user-defined target (the notion of far is defined in the paragraph dedicated to the CRE-based interpretable stopping criterion).

After updating the value of α, the Step 1 is inialized with the solution $\hat{s}$ obtained before the update.

Adaptive tuning of the learning rate

Another critical parameter in the training of neural networks is the selection of the learning rate, denoted as l_r [68]. An excessively small value for l_r can result in a slow training process, while an overly large value can introduce instability during training, potentially preventing the network from reaching convergence. Evaluating the appropriate learning rate should involve an analysis of the progress achieved during a single training step. In this context, a valuable indicator of the training update speed can be obtained from the two-step minimization procedure. The idea behind the evaluation of the progress made during training can be found in Figure 3. The number of iterations made in Step 1 for a given epoch is directly linked to the value of the update made in Step 2 of previous epochs. Hence, starting from the second epoch, the learning is updated so that the number of iterations made is close to an empirically defined target on the number of iterations. This rule has been designed on a different test case from the one presented in this article and the influence of this empirical rule will be shown in Section 4.2.5. This rule is strongly inspired by [34] even though the minimization performed in Step 1 is different.

According to multiple experiments, a good compromise is located around 6 iterations in Step 1. It is thus possible to automatically adapt the learning rate with the following empirical-based rule:

$$lr \leftarrow lr \times \texttt{update\_coefficient(number\_iterations\_step\_1)} \tag{21}$$

where `update_coefficient(number_iterations_step_1)` is a function depending on the number of iterations performed in previous Step 1. A point of attention is that the function `update_coefficient(number_iterations_step_1)` should be changed if the tolerance of the stopping criterion of Step 1 is modified (with a smaller tolerance the number of iteration is naturally higher). The main advantage of this rule is that the convergence of the method is no more sensitive to the user learning rate choice, as shown in Section 4.2.5, which enables to train the network online.

A point of attention is that near convergence, the learning rate should not be increased, as otherwise there is a risk of getting far from the global minimum. To do so, near convergence the learning rate is only updated if the rule tends to decrease the learning rate (the notion of near convergence is defined in the next paragraph). Additionally, a replay strategy has been implemented if the learning rate is too large, thus implying a very large number of iterations in Step 1. In this case, the model before updating is reloaded and a new gradient descent step is performed with a lower learning rate.

CRE-based interpretable stopping criterion

Training neural networks requires the establishment of a termination condition. One commonly employed approach involves initially specifying a predetermined number of epochs for training and subsequently adjusting this parameter based on the performance of predictions [69]. However, this approach is unsuitable in the context of Dynamic Data-Driven Application Systems (DDDAS) since training must occur online.

Another frequently adopted rule involves defining a criterion to identify overfitting, with the objective of stopping training when the loss computed on the validation dataset plateaus while the loss on the training dataset continues to decrease. This criterion is not applicable in this scenario due to the unsupervised nature of the training.

The concept of mCRE offers a strong physical foundation that can be employed to establish a termination criterion. As it will be illustrated in Section 4.3, the CRE can be interpreted as a modeling error. The CRE is homogeneous to an energy and can thus be compared to the energy in the structure; the normalized CRE reads:

$$\mathcal{E}^2_{\text{normalized\_}CRE} = \frac{\int_\Omega \int_0^T \eta_\psi}{\int_\Omega \int_0^T \psi(s_{\mathbf{v}})} \tag{22}$$

A similar expression can be written for the part concerning the dissipation potential and involving φ and η_φ.

This criterion is also used to estimate how far from convergence the minimization process is, which is required in the α and l_r updating rules.

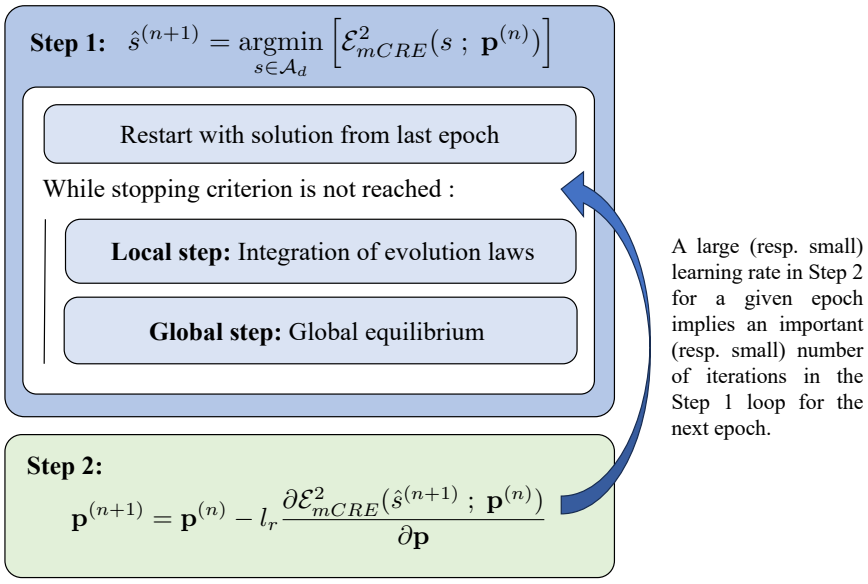


Figure 3: Influence of the learning rate on the optimization process.

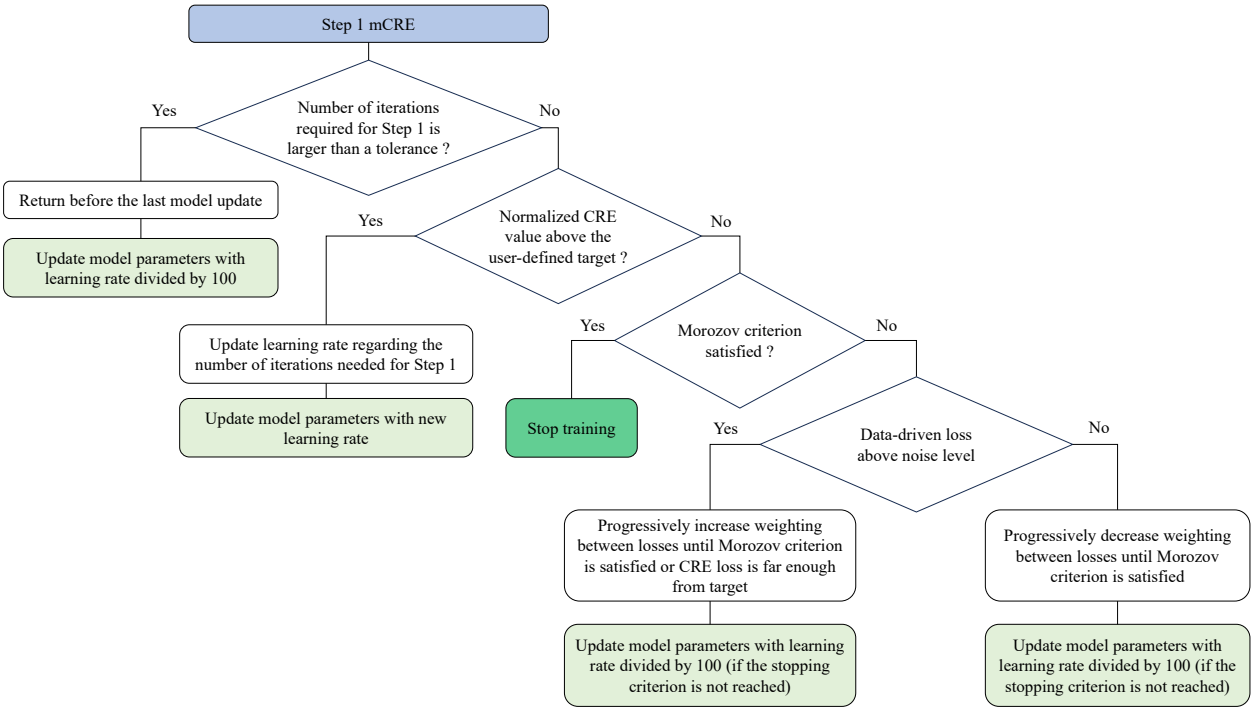


Figure 4: General idea for hyperparameters updating.

4 Results and discussion

This section is dedicated to the results and discussion on three different test cases. The first one in Section 4.1 is a toy example in 1D to illustrate the mCRE method in a case of parameter identification: only one parameter of a given hardening law is identified. The second one in Section 4.2 aims at learning a nonlinear hardening law. Finally the third one in Section 4.3 aims at learning a viscoplastic behavior. The evaluation of the performance is based on several criteria such as the accuracy of the learned model (Sections 4.2.3 and 4.3.2), the evolution of the loss function during the training (Sections 4.2.3 and 4.3.2), the noise robustness (Section 4.2.4), the localization of modeling error (Section

4.3.3), the relevance of automatic hyperparameters tuning rules (learning rate in Section 4.2.5, and the weighting between losses in Section 4.3.4).

4.1 A simple 1D example to identify an isotropic hardening modulus

Before presenting the results of neural network training with the mCRE framework, this section aims to show parameter identification of the hardening modulus in a simple 1D toy problem. This section is associated with an open-access code [1] and can be helpful for a better understanding of the mCRE concept, hyperparameters influence and tuning. Figure 5 illustrates the problem that is addressed. A one-dimensional beam, with Young modulus E, initial elasticity limit R_0, linear hardening modulus h and length $L = 1$, is loaded in tension. For the sake of simplicity, this problem is limited to one degree of freedom, in which the displacement is observed and affected by measurement noise. Here the constitutive laws are assumed to be known and the only parameter to identify is the hardening modulus h. The potentials are the following:

$$\psi(\epsilon_e, p) = \frac{1}{2} E \epsilon_e^2 + \frac{1}{2} h p^2 \tag{23}$$

$$\varphi^*(\sigma, R) = \begin{cases} 0 \text{ if } f < 0 \\ +\infty \text{ if } f = 0 \end{cases} \tag{24}$$

with $f = \sigma - (R + R_0)$.

Consequently,

$$\sigma = E \epsilon_e \tag{25}$$

$$R = hp \text{ with } p = \int_0^t ||\dot{\epsilon}_p|| \tag{26}$$

and, following the normality rule :

$$\dot{\epsilon}_p = \dot{\lambda} \frac{\partial f}{\partial \sigma} \tag{27}$$

with $\dot{\lambda} > 0$ if $f = 0$ and $\dot{f} = 0$.

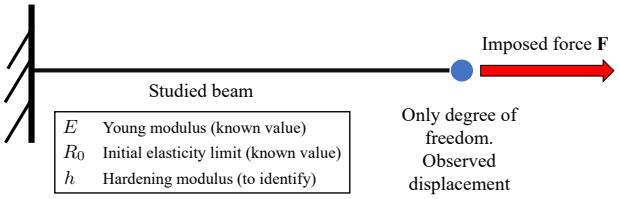


Figure 5: Simple 1D problem: beam with isotropic hardening.

The initial guess is $h_{ini} = 5 \times 10^4$MPa, whereas the target value is $h_{true} = 10^5$MPa. Figure 6 shows the difference between true and predicted responses for the initial parameter guess. For a given training epoch, after the first step (computing an admissible solution that minimizes $\mathcal{E}_{mCRE}$ for given model parameters), the expression of stationarity conditions of the Lagrangian defined in (19) gives that:

- $s_{\mathbf{u}}$ is a compromise between the solution of the forward problem and the observed data (hybrid state);
- $s_{\mathbf{v}}$ is the solution of the forward problem.

Figure 7 shows these solutions at the end of Step 1 for the first epoch, for different values of α. Indeed, the compromise between the resolution of the forward problem and the observed data depends on the value of α. The larger the α, the closer the predicted displacement field to the true displacement field. In the example of Figure 7, two values of α are presented. With $\alpha = 10^6$, the predicted strain associated with solution $s_{\mathbf{u}}$ corresponds exactly to the true strain.

As mentioned in Section 3.2, the tuning of this parameter is important for at least two reasons:

[1] `https://gitlab-research.centralesupelec.fr/antoine.benady/mcre_evolution_1d/-/tree/main?ref_type=heads`

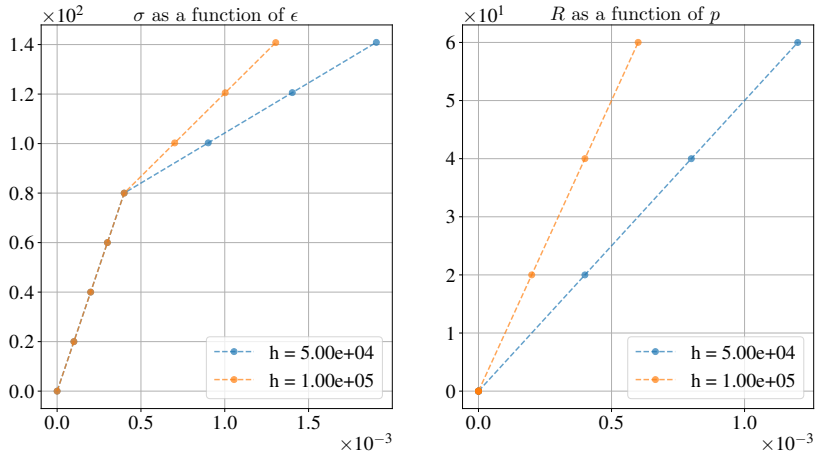


Figure 6: True and predicted responses with initial parameters.

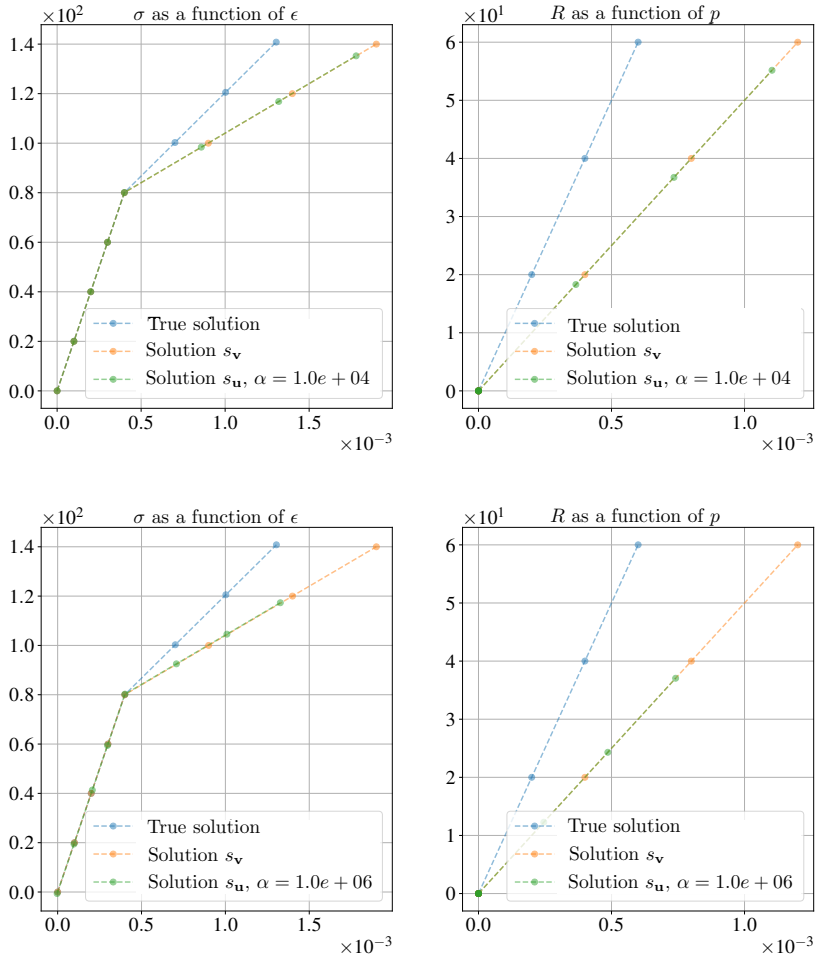


Figure 7: Solutions $s_{\mathbf{u}}$ and $s_{\mathbf{v}}$ for different values of α (top: $\alpha = 10^4$, bottom $\alpha = 10^6$)

- At the end of the training process, the compromise between the model and measurements should be such that the discrepancy between the predicted solution $s_{\mathbf{u}}$ and the observations is of the order of the magnitude of the noise level.
- As the gradient of the mCRE is directly linked to the difference between $s_{\mathbf{u}}$ and $s_{\mathbf{v}}$, the value of α has a strong influence on the optimization process. Yet, this influence is strongly linked to the role of the learning rate. As tuning α with respect to the Morozov criterion is more computationally expensive than tuning the learning rate following the empirical rule presented in Section 3.2, this value is only tuned at convergence.

Figure 8 (top) shows the results of the training for various epochs. At the end of the training, the parameter h is properly identified and the solution $s_{\mathbf{v}}$ is close to the measurements. The constitutive relation, even though it is not directly observed, is properly learned. The bottom of Figure 8 shows that the mCRE functional decreases smoothly. In this toy example, the hyperparameters are manually tuned because the computational time is very low (less than 30 seconds for the whole training).

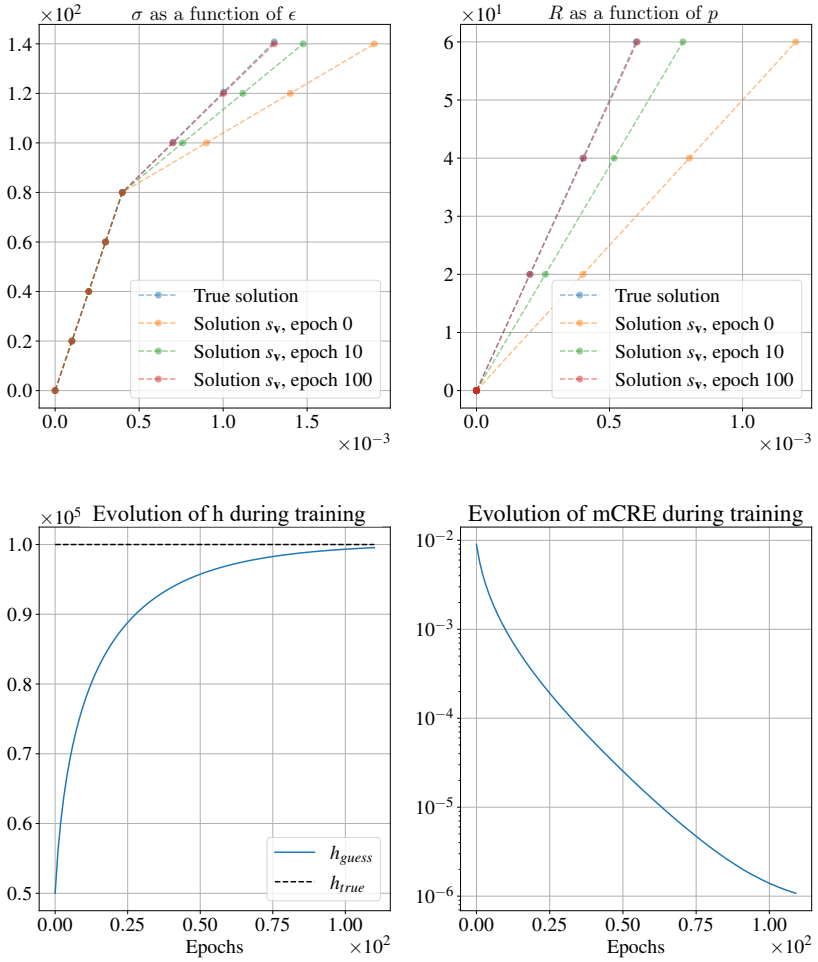


Figure 8: Results of training for the identification of h for the 1D test case.

4.2 Rate-independent test case: learning of a nonlinear isotropic hardening law

4.2.1 Reference problem

In the previous section, the model form was known and only one parameter was identified. In the use cases targeted in this paper, only an initial guess on the model form is known. The initial guess is represented by a neural network that is trained in the mCRE framework. To represent the initial guess, it is possible to perform a first supervised training with the Adam optimizer for example ([70]) before the mCRE training.

In this section, a nonlinear isotropic hardening law is recovered. For the sake of simplicity, a part of the model is assumed to be known as represented in Table 1, with the values of the parameters in Table 4. This is a modeling choice dictated by the reliability of material knowledge: other strategies can be imagined such as identifying some model parameters simultaneously or even representing every part with neural networks. The mCRE strategy easily adapts to these cases. It is worth noticing that in the addressed problem, the yield criterion is assumed to be known, which means that the dissipation potential is known (it is recalled that in the rate-independent case, the pseudo-potential is the indicator function of the convex elastic domain as defined in (2)). In the rate-independent case, representing the dissipation potential by a neural network is not convenient: it is more adapted to use the neural networks to represent the yield criterion.

Figure 9 shows the geometry, boundary conditions, and sensor positions of the problem under study. The studied case is a 2D beam loaded in tension with variable loading forces. The generated measurements come from two optic fibers oriented along the two main axes and positioned in the middle of the beam.

Model used to generate data	**Model to train**
Elastic part of the free energy:	
$\psi_e(\mathbb{\varepsilon}_e) = \frac{1}{2}(\lambda(\mathrm{tr}\,\mathbb{\varepsilon}_e)^2 + 2\mu\mathbb{\varepsilon}_e\colon \mathbb{\varepsilon}_e)$	Assumed to be known with the correct parameters
Plastic part of the free energy:	
$\psi_p(\mathbb{\varepsilon}_p) = Ap + \frac{A}{B}(e^{-Bp} - 1)$	Represented by a neural network initialized with: $\psi_p(\mathbb{\varepsilon}_p) = \frac{1}{2}hp^2$
Limit of the elasticity domain:	
$f = \mathbb{\sigma}_{eq} - (R + R_0)$	Assumed to be known with the correct parameters

Table 1: Summary of the nonlinear hardening law test case, with $\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}$, $\mu = \frac{E}{2(1+\nu)}$ and $\mathbb{\sigma}_{eq}$ the Von-Mises equivalent stress.

Parameters	**Value**
E	200 GPa
ν	0.25
A	45 MPa
B	3000
h	50 GPa
R_0	100 MPa

Table 2: Values of parameters used in the nonlinear hardening law test case.

4.2.2 mCRE minimization for the rate-independent test case.

This section aims to detail the mCRE minimization in the specific case treated. Step 1 minimization of the mCRE is composed of a local step (integration of evolution laws) and a global step (computation of admissible fields).

Local step

In the local step, the variables $\mathbb{\varepsilon}(\mathbf{u})$ and $\mathbb{\varepsilon}(\mathbf{v})$ and the variables $p_{\mathbf{u}}, p_{\mathbf{v}}, R_{\mathbf{u}}, R_{\mathbf{v}}, \mathbb{\varepsilon}_{e,\mathbf{u}}, \mathbb{\varepsilon}_{e,\mathbf{v}}, \mathbb{\varepsilon}_{p,\mathbf{u}}, \mathbb{\varepsilon}_{p,\mathbf{v}}$ are searched. The integration is performed with an Euler implicit scheme. In the following, the indices $\mathbf{u}$ and $\mathbf{v}$ are dropped and the integration is written in a generic form. For each element and for each time step, the integration consists in finding (both for $\mathbf{u}$ and $\mathbf{v}$ solution):

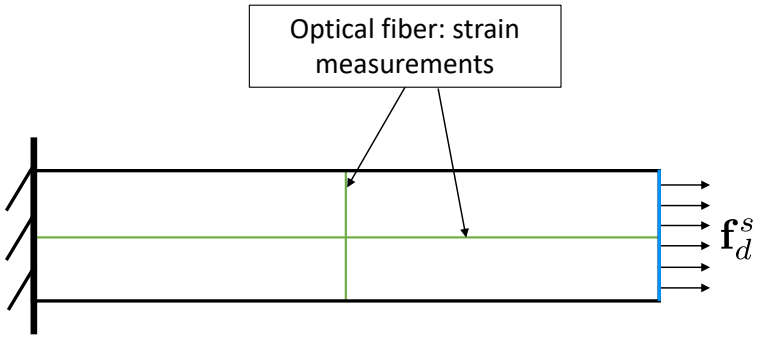


Figure 9: Geometry of the problem under study.

$$\begin{cases} \dot{\mathfrak{e}}_p = \dot{\lambda}\frac{\partial f}{\partial \sigma} \\ \dot{p} = -\dot{\lambda}\frac{\partial f}{\partial R} \\ f = 0 \\ R = \frac{\partial \psi_p(p)}{\partial p} \\ \dot{\mathfrak{e}} = \dot{\mathfrak{e}}_e + \dot{\mathfrak{e}}_p \\ \sigma = \frac{\partial \psi_e}{\partial \mathfrak{e}_e} \end{cases} \tag{28}$$

After time discretization, (28) becomes:

$$\begin{bmatrix} \Delta\mathfrak{e} - (\mathfrak{e}_e^{t+1} - \mathfrak{e}_e^t) - \Delta\lambda\frac{\partial f}{\partial \sigma} \\ p^{t+1} - p^t + \Delta\lambda\frac{\partial f}{\partial R} \\ R^{t+1} - R^t + \Delta\lambda\frac{\partial f}{\partial R}\frac{\partial^2 \psi_p(p)}{\partial p^2} \\ f(\mathfrak{e}_e^{t+1}, R) \\ \sigma - \frac{\partial \psi_e}{\partial \mathfrak{e}_e} \end{bmatrix} = \mathbf{0} \tag{29}$$

where $\bullet^t$ denotes the quantity $\bullet$ at the time step t. The system (33) is solved with a Newton-Raphson method in which the Jacobian matrix is computed through automatic differentiation. At the end of the local step, the variables $p_{\mathbf{u}}, p_{\mathbf{v}}, R_{\mathbf{u}}, R_{\mathbf{v}}, \mathfrak{e}_{e,\mathbf{u}}, \mathfrak{e}_{e,\mathbf{v}}, \mathfrak{e}_{p,\mathbf{u}}, \mathfrak{e}_{p,\mathbf{v}}$ are updated.

Global step

During the global step, the variables $\mathbf{X_u}, \mathbf{Y_u}, \mathfrak{e}_{p,\mathbf{u}}, \mathbf{X_v}, \mathbf{Y_v}, \mathfrak{e}_{p,\mathbf{v}}$ are frozen to the value obtained at the local step. An admissible solution is computed through the minimization of the Lagrangian defined in (19), with $\mathbf{X_u} = r_u, \mathbf{Y_u} = R_u, \mathbf{X_v} = r_{\mathbf{v}}, \mathbf{Y_v} = R_{\mathbf{v}}$

Step 2

As the dissipation potential is assumed to be known here, only the parameters involved in the plastic part of the free energy $\mathbf{p}_{\psi_p}$ are updated through gradient descent:

$$\mathbf{p}_{\psi_p}^{n+1} = \mathbf{p}_{\psi_p}^{n} - \frac{\partial \mathcal{E}_{mCRE}^2(s_{\mathbf{u}}, s_{\mathbf{v}}; \mathbf{p})}{\partial \mathbf{p}_{\psi_p}} \tag{30}$$

with

$$\frac{d\mathcal{E}^2_{mCRE}(s_{\mathbf{u}}, s_{\mathbf{v}}; \mathbf{p})}{d\mathbf{p}_{\psi_p}} = \int_0^T \int_\Omega \left(\frac{\partial \psi}{\partial \mathbf{p}_{\psi_p}}\Big|_{s_{\mathbf{u}}} - \frac{\partial \psi}{\partial \mathbf{p}_{\psi_p}}\Big|_{s_{\mathbf{v}}} \right) \tag{31}$$

4.2.3 General results

At the beginning of the training, the initial guess on the model form is wrong (see Table 1), which is referred to as model bias. If the form of the model were not released (*i.e.* not described by a neural network), it would not be possible to properly fit observations. In Figure 10, the material behavior is shown for the true model and initial guess. On the right, the shape of the hardening part of $\mathbb{\sigma}_{11}$ as a function of $\mathbb{\varepsilon}_{11}$ is not properly represented. The initial guess is linear whereas the true response has an exponential shape.

As previously mentioned, representing a constitutive model by a neural network enables relaxing the model form. Indeed, Figure 11 shows the response at the end of the training: the hardening law has evolved from a linear law to a nonlinear law and the response is properly represented. The relation R as a function of p is learned even though none of these quantities is observed in practice.

Figure 12 shows the evolution of the mCRE (with the CRE and distance to observation term) during the training. At the end of the training, the normalized data loss is approximately 1, which means that the Morozov criterion is satisfied. Concerning the normalized CRE, it is below the user-defined target value. As these two criteria are met and following the procedure defined in Figure 4, the training stops. The gap between the learnt relation and the true one (see Figure 11) is explained by the Morozov criterion: the updated model should not fit data below the noise level (1% here, where the noise is on the total strain observations coming for the optic fiber measurements).

After epoch 500, the normalized CRE has an oscillating behavior, which is explained by the updating of the weighting between losses α. The evolution of the parameter α is represented in Figure 13.

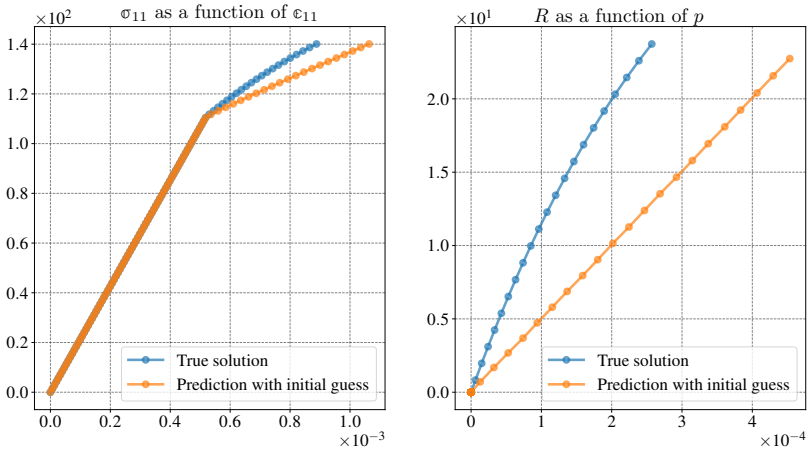


Figure 10: Model bias at the beginning of the training.

4.2.4 Robustness to noise level

Previous works about mCRE have shown the high robustness of the method to noise level (see [21, 20] in a case dealing with parameter identification and [34] with neural networks). This section aims to briefly analyze the noise robustness of this test case. In Figure 14, the evolution of the loss function, divided into the CRE and the distance to observations, is represented for different noise levels from 0.01% to 20%. The first observation to notice is the value of the normalized data loss at the end of training. This value is close to one, which means that the Morozov criterion is satisfied: the gap between the predicted strain and the measured strain is the same order of magnitude as the noise level. Secondly, the normalized CRE is decreasing during training, until the stopping criterion is met. These two observations show that the training is reaching the expected behavior: the modeling error is decreased to the expected value and the model is updated up to noise level.

When the noise is 20%, the right of Figure 14 shows that the mCRE is increasing while the normalized CRE and the normalized data loss are decreasing. This is explained by the increasing value of α that follows an evolution similar to

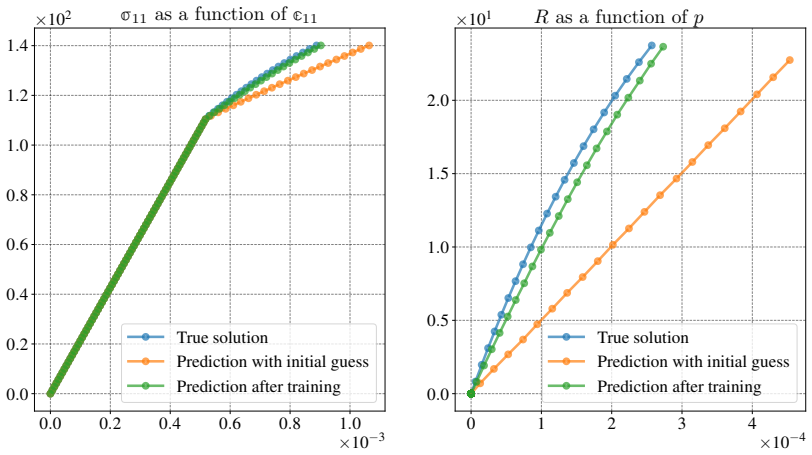


Figure 11: Learned hardening law at the end of the training.

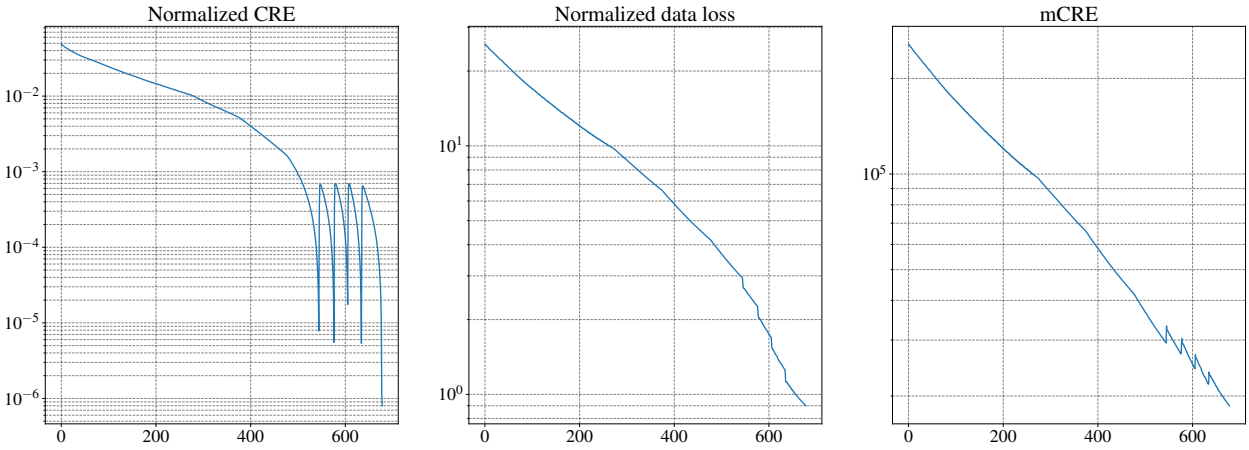


Figure 12: Left: evolution of the normalized CRE during the training. Middle: evolution of the normalized data loss during the training. Right: evolution of the mCRE during the training. For the 3 curves, the values are computed after step 1 and before updating the weights and biases of neural networks.

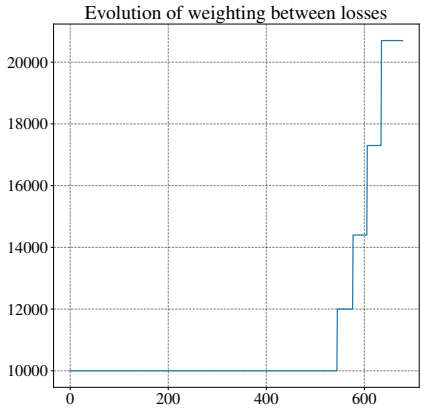


Figure 13: Evolution of α during the training.

the one in Figure 13. For the highest noise level, the training requires fewer epochs than the lower noise level. The first reason is that the value of α at the beginning of the training is adequate (normalized data loss is close to one) and as the model should not be updated below the noise level, the correction from the initial guess is lower than for lower noise levels as it is further shown.

Now that the behavior of the training has been checked, it is interesting to observe the curve of the learned hardening law and the response σ_{11} as a function of ε_{11}. Figure 15 shows this for two different noise levels ($0, 01\%$ and 20%). For the low noise level (even though it is a quite high noise level compared to other methods such as [29]) the hardening law is properly learned. On the other hand, for the noise level of 20%, only a slight correction has been made. Indeed, on the left of Figure 15, the gap between the predicted strain and the true strain (without noise on the curve) is around 20%: it would make no sense to update below the noise level.

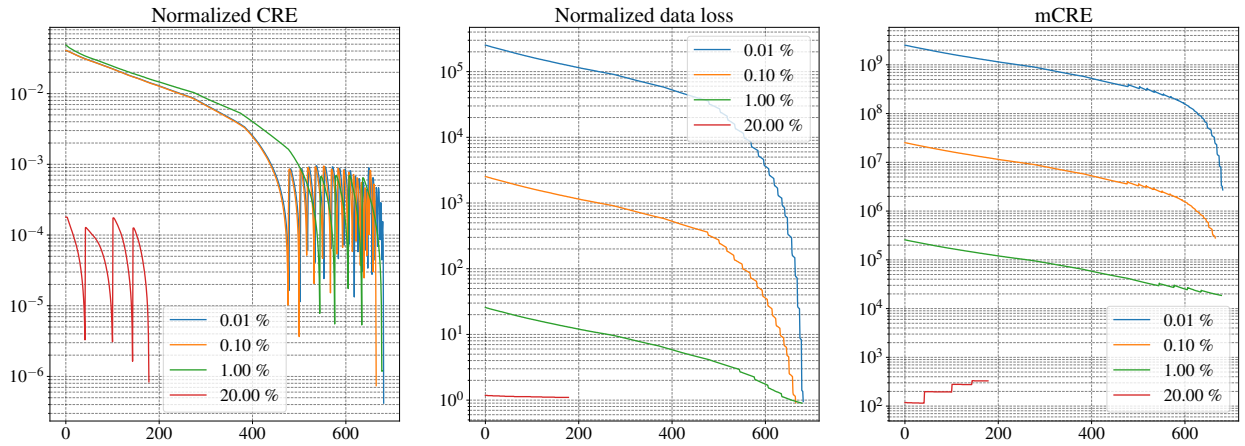


Figure 14: Impact of noise level on convergence. Left: evolution of the normalized CRE during the training. Middle: evolution of the normalized data loss during the training. Right: evolution of the mCRE during the training. For the 3 curves, the values are computed after Step 1 and before updating the weights and biases of neural networks.

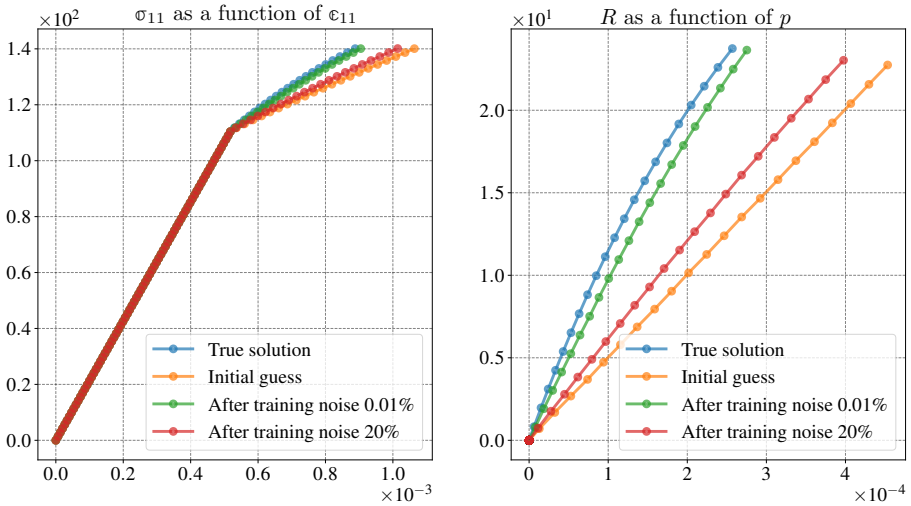


Figure 15: Learned hardening law at the end of training for different noise levels.

4.2.5 Relevance of the automatic learning rate tuning

This section aims to briefly illustrate the efficiency of the adaptive tuning of the learning rate presented in Section 3.2. In Figure 16, the evolution of the mCRE during training is represented for three different values of the initial

learning rate chosen by the user (with a difference of 5 orders of magnitude between the highest and the lowest). One strong advantage of this rule is that the level of mCRE reached at convergence is the same for the 3 initial values. This means that the training is not sensitive to the user learning choice, which is rarely the case in gradient-descent-based optimization in Deep Learning [69]. This is explained by the adaptive rule which automatically chooses close values of the learning rate after several epochs, independently of the initial choice. A sensitivity remains regarding the training time, which is not a major concern here.

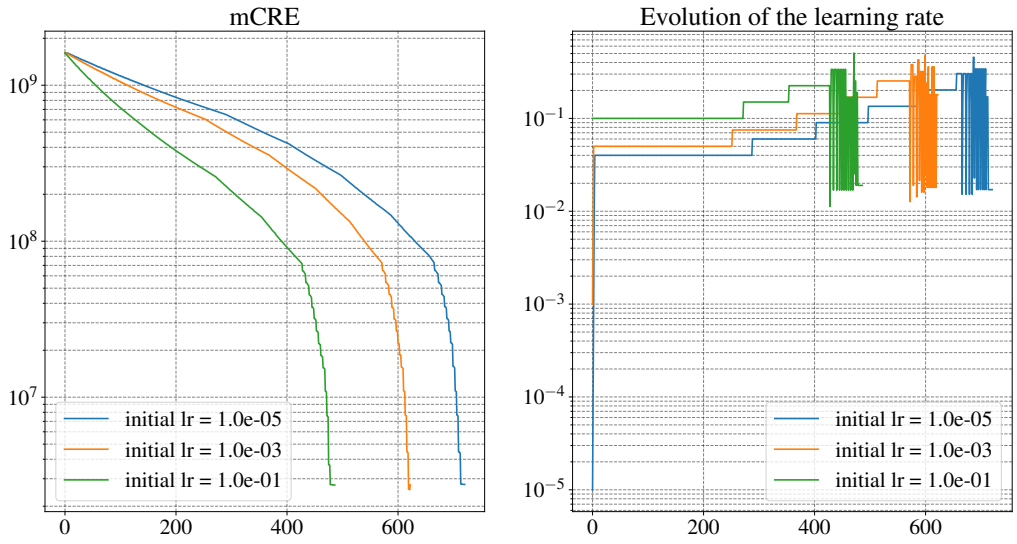


Figure 16: Influence of the initial learning rate on the optimization process.

4.3 Rate-dependent test case: learning of a dissipation pseudo-potential

In this section, the method is tested with a viscoplastic behavior. As in the previous section, a part of the constitutive model is assumed to be known and another part is released through the use of a neural network. Here, the free energy is assumed to be fully known (linear elasticity and linear hardening law), as well as the boundary of the elasticity domain. The parameters involved in the free energy and elasticity domain are also assumed to be known (even though they could be updated in the same way as the neural network parameters). In contrast, the form of the dissipation pseudo-potential is released with a neural network. This neural network is initialized to represent a classical power law [52], whereas the model used to generate the data is a hyperbolic cosine law [71], as summarized in Table 3. Figure 18 shows the discrepancy between the initial and the true models.

The case under study is the 2D notched beam shown in Figure 17. The beam is subjected to tension with varying loading forces, while the full-field displacement is measured.

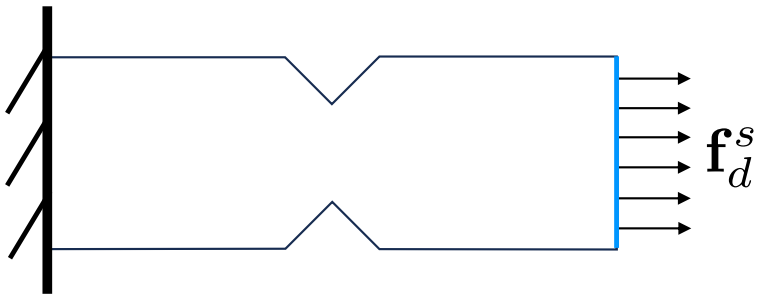


Figure 17: Geometry of the problem under study.

Model used to generate data	**Model to train**
Elastic part of the free energy:	
$\psi_e(\mathbb{\varepsilon}_e) = \frac{1}{2}(\lambda(\mathrm{tr}\,\mathbb{\varepsilon}_e)^2 + 2\mu\mathbb{\varepsilon}_e\colon\mathbb{\varepsilon}_e)$	Assumed to be known with the correct parameters
Plastic part of the free energy:	
$\psi_p(\mathbb{\varepsilon}_p) = \frac{1}{2}hp^2$	Assumed to be known with the correct parameters
Limit of the elasticity domain:	
$f = \mathbb{\sigma}_{eq} - (R + R_0)$	Assumed to be known with the correct parameters
Dissipation potential	
$\varphi^*(f) = AB\Big(\cos(\frac{< f >_+}{A}) - 1\Big)$	Represented by a neural network initialized with: $\varphi^*(f) = \frac{K_a}{N_a+1}\Big(\frac{< f >_+}{K_a}\Big)^{N_a+1}$

Table 3: Summary of the rate-dependent test case.

Parameters	**Value**
E	200 GPa
ν	0.3
A	18 MPa
B	$0.3s^{-1}$
h	1.5 GPa
R_0	160 MPa
K_a	50 MPa.s^{1/N_a}
N_a	10

Table 4: Values of parameters used in the rate-dependent test case.

4.3.1 mCRE minimization for the rate-dependent test case

This section aims to detail the mCRE minimization in the case of rate-dependent behavior. Step 1 minimization of the mCRE is composed of a local step (integration of evolution laws) and a global step (computation of admissible fields).

Local step

As in the rate-independent case, in the local step, the variables $\mathbb{\varepsilon}(\mathbf{u})$ and $\mathbb{\varepsilon}(\mathbf{v})$ and the variables $p_{\mathbf{u}}, p_{\mathbf{v}}, R_{\mathbf{u}}, R_{\mathbf{v}}, \mathbb{\varepsilon}_{e,\mathbf{u}}, \mathbb{\varepsilon}_{e,\mathbf{v}}, \mathbb{\varepsilon}_{p,\mathbf{u}}, \mathbb{\varepsilon}_{p,\mathbf{v}}$ are searched. The integration is performed with an Euler implicit scheme. In the following, the indices $\mathbf{u}$ and $\mathbf{v}$ are dropped and the integration is written in a generic form. For each element and for each time step, the integration consists in finding (both for $\mathbf{u}$ and $\mathbf{v}$ solution):

$$\begin{cases} \dot{\mathbb{\varepsilon}}_p = \dfrac{\partial \varphi^*}{\partial \mathbb{\sigma}} \\ \dot{p} = \dfrac{\partial \varphi^*}{\partial R} \\ R = \dfrac{\partial \psi_p(p)}{\partial p} \\ \dot{\mathbb{\varepsilon}} = \dot{\mathbb{\varepsilon}}_e + \dot{\mathbb{\varepsilon}}_p \\ \mathbb{\sigma} = \dfrac{\partial \psi_e}{\partial \mathbb{\varepsilon}_e} \end{cases} \tag{32}$$

After time discretization, (32) becomes:

$$\begin{bmatrix} \Delta\mathfrak{e} - (\mathfrak{e}_e^{t+1} - \mathfrak{e}_e^t) - \frac{\partial\varphi^*}{\partial\mathfrak{s}}dt \\ p^{t+1} - p^t + \frac{\partial\varphi^*}{\partial R}dt \\ R^{t+1} - R^t + \frac{\partial\varphi^*}{\partial R}\frac{\partial^2\psi_p(p)}{\partial p^2}dt \\ \mathfrak{s} - \frac{\partial\psi_e}{\partial\mathfrak{e}_e} \end{bmatrix} = \mathbf{0} \tag{33}$$

where $\bullet^t$ denotes the quantity $\bullet$ at the time step t, and dt the time increment. The system (33) is solved with a Newton-Raphson method in which the Jacobian matrix is computed through automatic differentiation. At the end of the local step, the variables $p_{\mathbf{u}}, p_{\mathbf{v}}, R_{\mathbf{u}}, R_{\mathbf{v}}, \mathfrak{e}_{e,\mathbf{u}}, \mathfrak{e}_{e,\mathbf{v}}, \mathfrak{e}_{p,\mathbf{u}}, \mathfrak{e}_{p,\mathbf{v}}$ are updated.

Global step

During the global step, the variables $\mathbf{X_u}, \mathbf{Y_u}, \mathfrak{e}_{p,\mathbf{u}}, \mathbf{X_v}, \mathbf{Y_v}, \mathfrak{e}_{p,\mathbf{v}}$ are frozen to the value obtained at the local step. An admissible solution is computed through the minimization of the Lagrangian defined in (19), with $\mathbf{X_u} = r_u, \mathbf{Y_u} = R_u, \mathbf{X_v} = r_{\mathbf{v}}, \mathbf{Y_v} = R_{\mathbf{v}}$

Step 2

As the free energy is assumed to be known here, only the parameters involved in the dissipation potential $\mathbf{p}_{\varphi^*}$ are updated through gradient descent:

$$\mathbf{p}_{\varphi^*}^{n+1} = \mathbf{p}_{\varphi^*}^{n} - \frac{\partial\mathcal{E}^2_{mCRE}(s_{\mathbf{u}}, s_{\mathbf{v}}; \mathbf{p})}{\partial\mathbf{p}_{\varphi^*}} \tag{34}$$

with

$$\frac{\partial\mathcal{E}^2_{mCRE}(s_{\mathbf{u}}, s_{\mathbf{v}}; \mathbf{p})}{\partial\mathbf{p}_{\varphi^*}} = \int_0^T \int_0^t \int_\Omega \Big(\frac{\partial\varphi^*}{\partial\mathbf{p}_{\varphi^*}}\Big|_{s_{\mathbf{v}}} - \frac{\partial\varphi^*}{\partial\mathbf{p}_{\varphi^*}}\Big|_{s_{\mathbf{u}}} \Big) \tag{35}$$

4.3.2 General results

This section presents the general results of the method on the viscoplastic test case. Figure 18 shows the results of the finite element simulation for the true model and neural network before and after training. The behavior is properly reconstructed after training and the model bias on the dissipation potential is properly corrected.

Figure 19 illustrates the evolution of the different terms of the mCRE during the training. At the end of the training, the stopping criterion is properly met as the normalized CRE is below the target $(1e^{-6})$ and the normalized data loss is close to 1, thus satisfying the Morozov criterion. The oscillating behavior of the normalized CRE is explained by the automatic update of the weighting between losses α. Indeed, every time the normalized CRE is below the target and the Morozov criterion is not met, the value of α is increased (see Figure 4). When α is increased, the normalized CRE increases and the normalized data loss decreases.

The previously obtained conclusions in the rate-independent test case regarding the robustness to noise level (Section 4.2.4) and the relevance of the learning rate tuning (Section 4.2.5) still stand in the rate-dependent test case (even though they are not presented again). The following sections evaluate the mCRE framework on the ability to properly localize the model bias on the structure in Section 4.3.3 and the relevance of the tuning of the weighting between losses α in Section 4.3.4.

4.3.3 Localization of the model bias in the structure

An interesting aspect of the mCRE framework is the localization of model bias. In the present example, Figure 18 shows that the plastic behavior is not properly modeled before the training of the neural network. This can also be seen in Figure 20 in which the cumulative plastic strain is shown in the structure for the last time step of the simulation, both for the true model and the neural network before training. On the right of Figure 20, the normalized CRE shows that the modeling error is localized in the part of the structure in which there is plasticity. This observation is explained by the

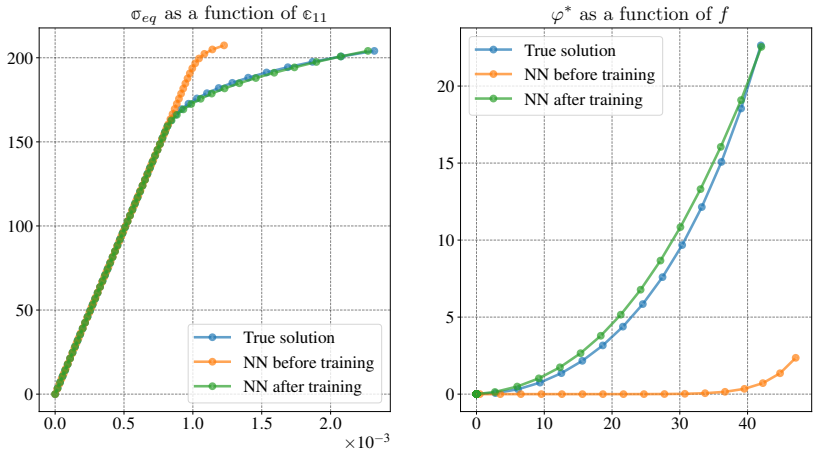


Figure 18: Finite element simulation for the true model and neural network before and after training (with 0.1% noise level).

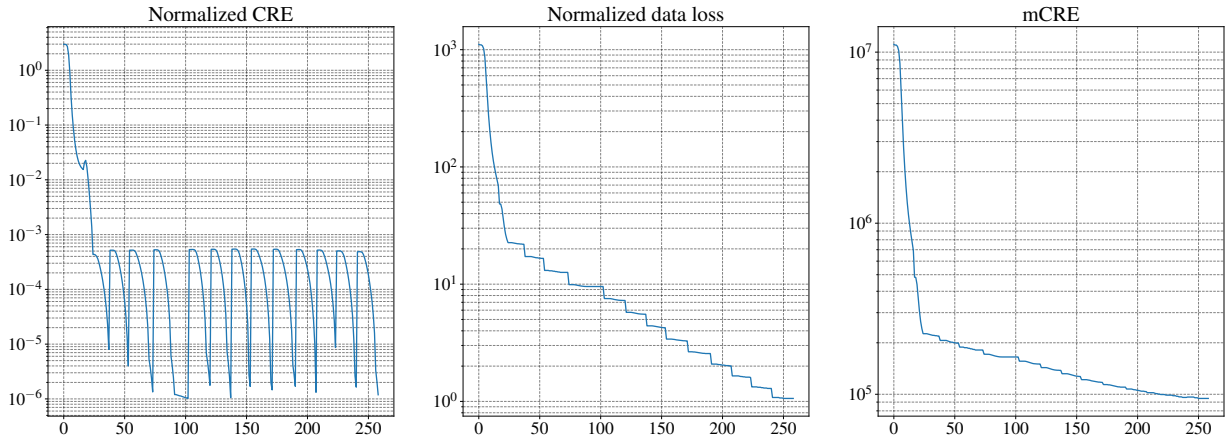


Figure 19: Left: evolution of the normalized CRE during the training. Middle: evolution of the normalized data loss during the training. Right: evolution of the mCRE during the training.

fact that this is the plastic part of the behavior that is not properly modeled. It is worth noticing that the computation of this error only requires information that is available in the inference phase (no use of the true potential nor true internal variable value).

4.3.4 Relevance of the automatic tuning of the weighting between losses

This section aims to briefly illustrate the relevance of the automatic tuning of the weighting between losses presented in Section 3.2. In Figure 21, the evolution of the mCRE during training is represented for four different values of the initial weighting between losses chosen by the user (with a difference of 4 orders of magnitude between the highest and the lowest). Figure 4 can be helpful to understand the evolution of α during training: the value of α is increased every time the normalized CRE is below the target and the Morozov criterion is not met. Figure 21 shows that the level of mCRE reached at convergence is the same for the 4 initial values, which means that the converged value is not sensitive to the user's choice of the initial value. Yet, a strong sensitivity remains regarding the computation time. Indeed a proper initial value can help achieve quicker convergence ($\alpha = 10000$) in Figure 21. This issue might be overcome by adding an initial automatic tuning at the beginning of training, such as is done in [34].

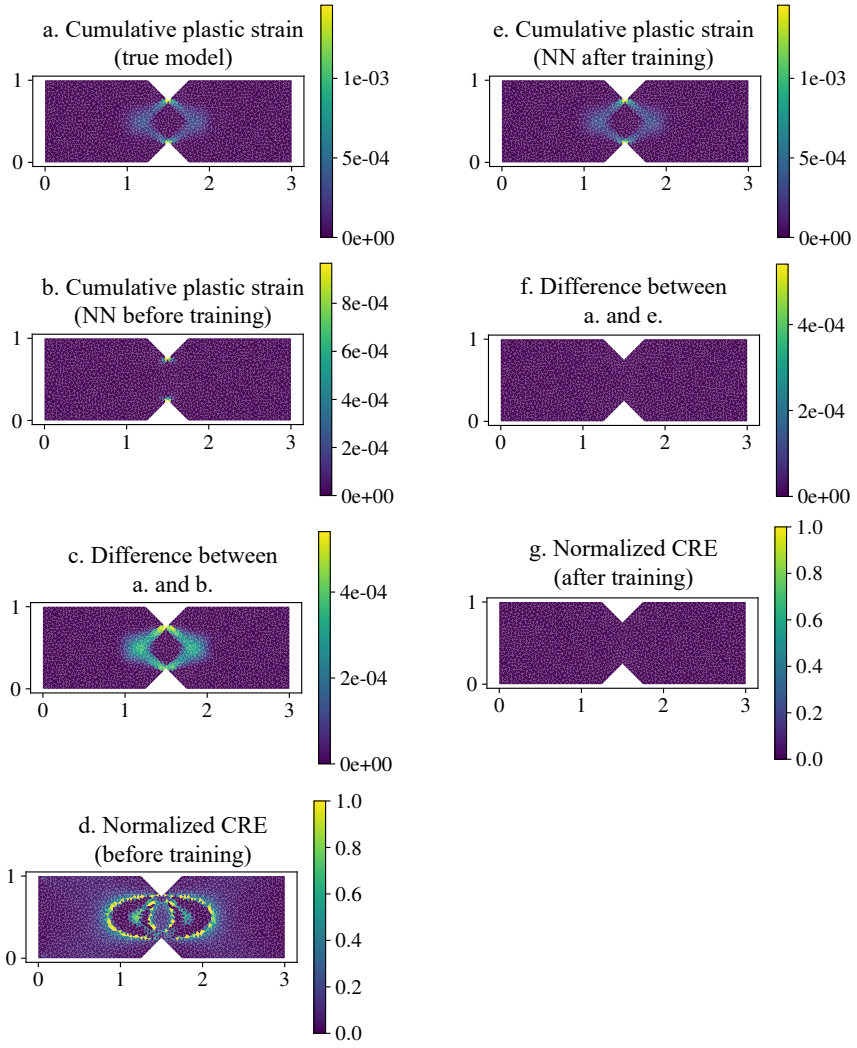


Figure 20: a: Cumulative plastic strain obtained with the true model. b: Cumulative plastic strain obtained with the neural network before training. c: Difference between the cumulative plastic strain obtained with the true model and the cumulative plastic strain obtained with the neural network before training. d: Normalized CRE at the beginning of the training. e: Cumulative plastic strain obtained with the neural network after training. f: Difference between the cumulative plastic strain obtained with the true model and the cumulative plastic strain obtained with the neural network after training. g: Normalized CRE at the beginning of the training.

5 Conclusion

This paper presented an innovative method for training physics-augmented neural networks to represent evolutionary laws. The approach involves the use of partial strain or displacement measurements for unsupervised training in which the modified Constitutive Relation Error is minimized. An important feature of the method is the classification of information with respect to its reliability. All reliable information is guaranteed by construction. The respect of thermodynamic principles is achieved by formulating the constitutive model in the generalized standard material framework with convex potentials represented by input-convex neural networks. The admissibility properties are enforced in the minimization process. In this bias-aware approach, the constitutive model is considered as unreliable information and is released by the use of a neural network. The noisy measurements are also released in the minimization. Moreover, the mCRE minimization provides an interpretable framework as the associated function provides a rich physical sense, the CRE being interpreted as a modeling error in the structure.

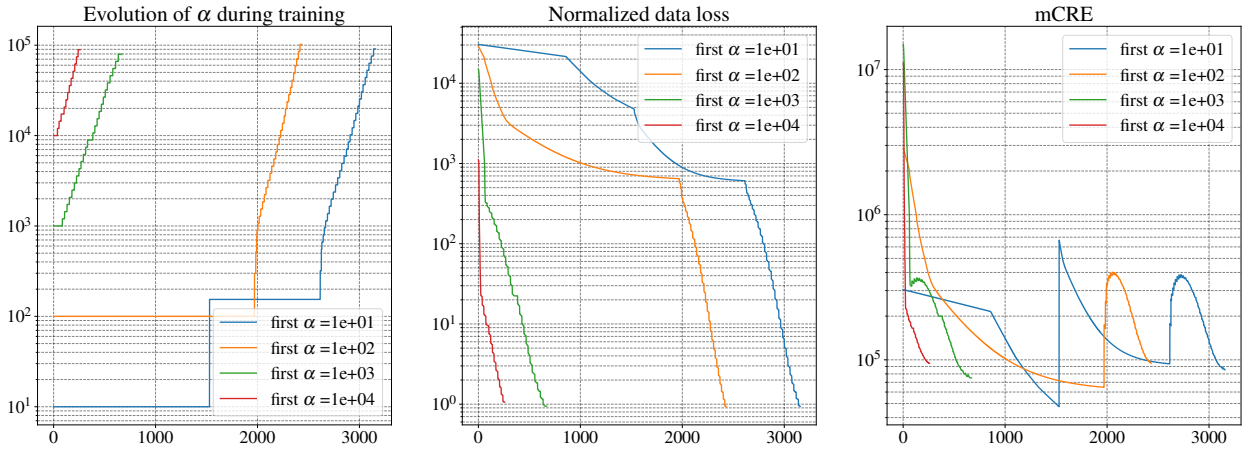


Figure 21: Left: evolution of the weighting between losses during the training. Middle: evolution of the normalized data loss during the training. Right: evolution of the mCRE during the training.

The performance of the methods was accessed on three test cases: a simple 1D problem where only one parameter of the constitutive model was updated, an elastoplastic behavior where a nonlinear hardening law was learned, and a viscoplastic behavior where a cosine hyperbolic dissipation potential was learned. The method showed interesting performance in terms of accuracy of the learned model, robustness to noise, localization of modeling error, and sensitivity of the law to user-defined hyperparameters. Since this research is part of a larger project [2] aiming at correcting the model bias of a hybrid twin involved in a DDDAS architecture, special attention was given to the automatic tuning of the hyperparameters (weighting between losses, learning rate, initialization). In summary, the proposed approach represents a promising tool for predicting the response of materials and structures to external loads. This could pave the way for new avenues of research in the field of computational mechanics.

However, this work requires additional studies before being applied to real structures. First, the computational cost seems prohibitive (a few hours for a 2D beam). This problem could be overcome by coupling this work with previous ones on the mCRE suitable for real-time control using reduced order modeling [16, 72] and Kalman filter [73]. Another reason is the required choice of postulated internal variables. Even if the model form of the relationships involving internal variables is properly corrected, a model bias remains in the choice of internal variables. A research direction to alleviate this problem could be the use of recurrent neural networks (RNN), in which the internal memories can play the role of internal variables [39], although the coupling with the mCRE error could be challenging.

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 101002857).

References

[1] Erik P. Blasch, Frederica Darema, and Dennis Bernstein. *Introduction to the Dynamic Data Driven Applications Systems (DDDAS) Paradigm*, pages 1–32. Springer International Publishing, Cham, 2022.

[2] Ludovic Chamoin. DREAM-ON: Merging advanced sensing techniques and simulation tools for future structural health monitoring technologies. In *The Project Repository Journal*, volume 10, pages 124–127. 2021.

[3] Ludovic Chamoin, S Farahbakhsh, and M Poncelet. An educational review on distributed optic fiber sensing based on Rayleigh backscattering for damage tracking and structural health monitoring. *Measurement Science and Technology*, pages 24–25, 2022.

[4] Francisco Chinesta, Elías G. Cueto, Emmanuelle Abisset-Chavanne, Jean Louis Duval, and Fouad El Khaldi. Virtual, Digital and Hybrid Twins: A New Paradigm in Data-Based Engineering and Engineered Data. *Archives of Computational Methods in Engineering*, 2019.

[5] Marc Bonnet and Andrei Constantinescu. Inverse problems in elasticity. *Inverse Problems*, 21(2):R1, 2005.

[6] Stéphane Avril, Marc Bonnet, Caro-Bretelle A.S, Michel Grédiac, François Hild, Patrick Ienny, Félix Latourte, D. Lemosse, Stéphane Pagano, Emmanuel Pagnacco, and Fabrice Pierron. Overview of identification methods of mechanical parameters based on full-field measurements. *Experimental Mechanics*, 48, 2008.

[7] Damien Claire, François Hild, and Stephane Roux. A finite element formulation to identify damage fields: The equilibrium gap method. *International Journal for Numerical Methods in Engineering*, 61, 2004.

[8] Nuno Maia, Marie Reynier, and Pierre Ladevèze. Error localization for updating finite element models using frequency-response-functions. 1994.

[9] Eric Florentin and Gilles Lubineau. Identification of the parameters of an elastic material model using the constitutive equation gap method. *Computational Mechanics*, 46:521–531, 2010.

[10] Pierre Ladevèze and Amar Chouaki. Application of a posteriori error estimation for structural model updating. *Inverse Problems*, 15:49, 1999.

[11] Biswanath Banerjee, Timothy F. Walsh, Wilkins Aquino, and Marc Bonnet. Large scale parameter estimation problems in frequency-domain elastodynamics using an error in constitutive equation functional. *Computer Methods in Applied Mechanics and Engineering*, 253:60–72, 2013.

[12] Arnaud Deraemaeker, Pierre Ladevèze, and Thierry Romeuf. Model validation in the presence of uncertain experimental data. *Engineering Computations*, 21:808–833, 2004.

[13] Olivier Allix, Pierre Feissel, and Pascal Thévenet. A delay damage mesomodel of laminates under dynamic loading: Basic aspects and identification issues. *Computers & Structures*, 81:1177–1191, 2003.

[14] Marc Bonnet and Wilkins Aquino. Three-dimensional transient elastodynamic inversion using an error in constitutive relation functional. *Inverse Problems*, 31(3):035010, 2015.

[15] Matthieu Diaz, Pierre-Étienne Charbonnel, and Ludovic Chamoin. A new Kalman filter approach for structural parameter tracking: application to the monitoring of damaging structures tested on shaking-tables. *Mechanical Systems and Signal Processing*, 182:109529, 2023.

[16] Basile Marchand, Ludovic Chamoin, and Christian Rey. Real-time updating of structural mechanics models using kalman filtering, modified constitutive relation error, and proper generalized decomposition. *International Journal for Numerical Methods in Engineering*, 107(9):786–810.

[17] Amar Chouaki, Pierre Ladevèze, and Laurent Proslier. An updating method for damped structural dynamic models. *Proceedings of the International Modal Analysis Conference - IMAC*, 1, 1997.

[18] Basile Marchand, Ludovic Chamoin, and Christian Rey. Parameter identification and model updating in the context of nonlinear mechanical behaviors using a unified formulation of the modified constitutive relation error concept. *Computer Methods in Applied Mechanics and Engineering*, 345:1094–1113, 2019.

[19] Hai Nguyen, Ludovic Chamoin, and Cuong Minh. Mcre-based parameter identification from full-field measurements: Consistent framework, integrated version, and extension to nonlinear material behaviors. *Computer Methods in Applied Mechanics and Engineering*, 400:115461, 2022.

[20] Olivier Allix, Pierre Feissel, and Hong-Minh Nguyen. Identification strategy in the presence of corrupted measurements. *Engineering Computations*, 22(5/6):487–504, 2005.

[21] Shaojuan Huang, Pierre Feissel, and Pierre Villon. Modified constitutive relation error: An identification framework dealing with the reliability of information. *Computer Methods in Applied Mechanics and Engineering*, 311:1–17, 2016.

[22] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

[23] Jamshid Ghaboussi, Xiping Wu, and Gintaris Kaklauskas PhD. Neural network material modelling. *Statyba*, 5(4):250–257, 1999.

[24] George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6), 2021.

[25] Jared Willard, Xiaowei Jia, Shaoming Xu, Michael Steinbach, and Vipin Kumar. Integrating scientific knowledge with machine learning for engineering and environmental systems. 55(4), 2022.

[26] Maziar Raissi, Paris Perdikaris, and Georges Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[27] Dominik Klein, Mauricio Fernández, Robert Martin, Patrizio Neff, and Oliver Weeger. Polyconvex anisotropic hyperelasticity with neural networks. 2021.

[28] Faisal As'ad, Philip Avery, and Charbel Farhat. A mechanics-informed artificial neural network approach in data-driven constitutive modeling. *International Journal for Numerical Methods in Engineering*, 123, 2022.

[29] Prakash Thakolkaran, Akshay Joshi, Yiwen Zheng, Moritz Flaschel, Laura De Lorenzis, and Siddhant Kumar. Nn-euclid: Deep-learning hyperelasticity without stress data. *Journal of the Mechanics and Physics of Solids*, 169:105076, 2022.

[30] Jan Niklas Fuhg, Craig M. Hamel, Kyle Johnson, Reese Jones, and Nikolaos Bouklas. Modular machine learning-based elastoplasticity: Generalization in the context of limited data. *Computer Methods in Applied Mechanics and Engineering*, 407:115930, 2023.

[31] Lennart Linden, Dominik Klein, Karl Alexander Kalina, Jörg Brummund, Oliver Weeger, and Markus Kästner. Neural networks meet hyperelasticity: A guide to enforcing physics. 2023.

[32] Shenglin Huang, Zequn He, Bryan Chem, and Celia Reina. Variational onsager neural networks (vonns): A thermodynamics-based variational learning strategy for non-equilibrium pdes. *Journal of the Mechanics and Physics of Solids*, 163:104856, 2022.

[33] Brandon Amos, Lei Xu, and J. Zico Kolter. Input convex neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 146–155. PMLR, 2017.

[34] Antoine Benady, Emmanuel Baranger, and Ludovic Chamoin. NN-mCRE: a modified Constitutive Relation Error framework for unsupervised learning of nonlinear state laws with physics-augmented Neural Networks. *https://hal.science/hal-04102108*, 2023. preprint.

[35] Yi Wei, Quentin Serra, Gilles Lubineau, and Eric Florentin. Coupling physics-informed neural networks and constitutive relation error concept to solve a parameter identification problem. *Computers & Structures*, 283:107054, 2023.

[36] Ling Wu, Van Dung Nguyen, Nanda Gopala Kilingar, and Ludovic Noels. A recurrent neural network-accelerated multi-scale model for elasto-plastic heterogeneous materials subjected to random cyclic and non-proportional loading paths. *Computer Methods in Applied Mechanics and Engineering*, 2020.

[37] Annan Zhang and Dirk Mohr. Using neural networks to represent von mises plasticity with isotropic hardening. *International Journal of Plasticity*, 2020.

[38] Colin Bonatti and Dirk Mohr. On the importance of self-consistency in recurrent neural network models representing elasto-plastic solids. *Journal of the Mechanics and Physics of Solids*, 2022.

[39] Maysam Gorji, Mojtaba Mozaffar, Julian N. Heidenreich, Jian Cao, and Dirk Mohr. On the potential of recurrent neural networks for modeling path dependent plasticity. 2020.

[40] Nikolaos N. Vlassis and WaiChing Sun. Sobolev training of thermodynamic-informed neural networks for interpretable elasto-plasticity models with level set hardening. *Computer Methods in Applied Mechanics and Engineering*, 377:113695, 2021.

[41] Olga Ibragimova, Abhijit Brahme, Waqas Muhammad, Julie Lévesque, and Kaan Inal. A new ann based crystal plasticity model for fcc materials and its application to non-monotonic strain paths. *International Journal of Plasticity*, 2021.

[42] P. Weber, W. Wagner, and S. Freitag. Physically enhanced training for modeling rate-independent plasticity with feedforward neural networks. *Computational Mechanics*, 2023.

[43] Colin Bonatti and Dirk Mohr. One for all: Universal material model based on minimal state-space neural networks. *Science Advances*, 7(26):eabf3658, 2021.

[44] Sunyoung Im, Jonggeon Lee, and Maenghyo Cho. Surrogate modeling of elasto-plastic problems via long short-term memory neural networks and proper orthogonal decomposition. *Computer Methods in Applied Mechanics and Engineering*, 2021.

[45] Mojtaba Mozaffar, Ramin Bostanabad, Ramin Bostanabad, Wei Chen, Kornel F. Ehmann, Jian Cao, and Miguel A. Bessa. Deep learning predicts path-dependent plasticity. *Proceedings of the National Academy of Sciences of the United States of America*, 116:26414 – 26420, 2019.

[46] Diab W. Abueidda, Seid Koric, Nahil Sobh, and Huseyin Sehitoglu. Deep learning for plasticity and thermo-viscoplasticity. *International Journal of Plasticity*, 2021.

[47] Filippo Masi and Ioannis Stefanou. Multiscale modeling of inelastic materials with thermodynamics-based artificial neural networks (tann). *Computer Methods in Applied Mechanics and Engineering*, 398:115190, 2022.

[48] Xiaolong He and Jiun-Shyan Chen. Thermodynamically consistent machine-learned internal state variable approach for data-driven modeling of path-dependent materials. *Computer Methods in Applied Mechanics and Engineering*, 2022.

[49] Quercus Hernández, Alberto Badías, David González, Francisco Chinesta, and Elías Cueto. Structure-preserving neural networks. *Journal of Computational Physics*, 426:109950, 2021.

[50] Hans Christian Öttinger and Miroslav Grmela. Dynamics and thermodynamics of complex fluids. ii. illustrations of a general formalism. *Phys. Rev. E*, 56, 1997.

[51] Bernard Halphen and Quoc Nguyen. On generalized standard materials.[sur les materiaux standards generalises.]. *J Mec*, 14:39–63, 1975.

[52] Jean Lemaitre and Jean-Louis Chaboche. *Mechanics of Solid Materials*. Cambridge University Press, 1990.

[53] Hai Nam Nguyen. *New numerical strategies for robust, consistent, and computationally efficient model identification from full-field measurements*. Theses, Université Paris-Saclay, 2021.

[54] Pierre Ladevèze and Dominique Leguillon. Error estimate procedure in the finite element method and applications. *Siam Journal on Numerical Analysis*, 20:485–509, 1983.

[55] Pierre Ladevèze and Jean-Pierre Pelle. Mastering calculation in linear and nonlinear mechanics. 2004.

[56] Pierre Ladevèze and Nicolas Moës. A new a posteriori error estimation for nonlinear time-dependent finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, 157(1):45–68, 1998.

[57] Pierre Ladevèze. Constitutive relation errors for f.e. analysis considering (visco-) plasticity and damage. *International Journal for Numerical Methods in Engineering*, 52(5-6):527–542, 2001.

[58] Pierre Ladevèze, Djamel Nedjar, and M. Reynier. Updating of finite element models using vibration tests. *AIAA Journal*, 32:1485–1491, 1994.

[59] Pierre Ladevèze. *Principles of the Method of Large Time Increments*, pages 55–100. Springer New York, New York, NY, 1999.

[60] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. 9:249–256, 2010.

[61] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, 2015.

[62] Andrew Saxe, James Mcclelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. pages 1–22, 2014.

[63] Rafael Bischof and Michael Kraus. Multi-objective loss balancing for physics-informed deep learning. 2021.

[64] Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient pathologies in physics-informed neural networks. *SIAM J. Sci. Comput.*, 43:A3055–A3081, 2020.

[65] Wilkins Aquino, James Warner, and Manuel Diaz. Inverse material identification in coupled acoustic-structure interaction using a modified error in constitutive equation functional. *The Journal of the Acoustical Society of America*, 134:4065, 2013.

[66] Matthieu Diaz, Pierre-E Charbonnel, and Ludovic Chamoin. Robust energy-based model updating framework for random processes in dynamics: Application to shaking-table experiments. *Computers and Structures*, 264, 2022.

[67] V.A. Morozov. The error principle in the solution of operational equations by the regularization method. *USSR Computational Mathematics and Mathematical Physics*, 8(2):63–87, 1968.

[68] Robert A. Jacobs. Increased rates of convergence through learning rate adaptation. *Neural Networks*, 1(4):295–307, 1988.

[69] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016.

[70] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. 2015.

[71] Jean-Marc Pipard, Tudor Balan, Farid Abed-Meraim, and Xavier Lemoine. Physically-motivated elasto-visco-plastic model for the large strain-rate behavior of steels. *Key Engineering Materials*, 554-557:1164–1173, 2013.

[72] Ludovic Chamoin, Pierre-Eric Allier, and Basile Marchand. Synergies between the constitutive relation error concept and pgd model reduction for simplified v&v procedures. *Advanced Modeling and Simulation in Engineering Sciences*, 3, 2016.

[73] Matthieu Diaz, Pierre-Étienne Charbonnel, and Ludovic Chamoin. A new Kalman filter approach for structural parameter tracking: application to the monitoring of damaging structures tested on shaking-tables. *Mechanical Systems and Signal Processing*, 182:109529, 2023.