

Parallel projection—An improved return mapping algorithm for finite element modeling of shape memory alloys

Ziliang Kang^{a,b,c,*}, Daniel A. Tortorelli^{d,e}, Kai A. James^a

^a Aerospace Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

^b Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

^c Division of Gastroenterology, Brigham and Women's Hospital, Harvard Medical School, Boston, MA 02115, USA

^d Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

^e Lawrence Livermore National Laboratory, Livermore, CA 94550, USA

Received 29 April 2021; received in revised form 21 October 2021; accepted 14 November 2021

Available online xxxx

Abstract

We present a novel finite element analysis of inelastic structures containing Shape Memory Alloys (SMAs). Phenomenological constitutive models for SMAs lead to material nonlinearities, that require substantial computational effort to resolve. Finite element analysis methods, which rely on Gauss quadrature integration schemes, must solve two sets of coupled differential equations: one at the global level and the other at the local, i.e. Gauss point level. In contrast to the conventional return mapping algorithm, which solves these two sets of coupled differential equations separately using a nested Newton procedure, we propose a scheme to solve the local and global differential equations simultaneously. In the process we also derive closed-form expressions used to update the internal/constitutive state variables, and unify the popular closest-point and cutting plane methods with our formulas. Numerical testing indicates that our method allows for larger thermomechanical loading steps and provides increased computational efficiency, over the standard return mapping algorithm.

© 2021 Elsevier B.V. All rights reserved.

Keywords: Shape-memory alloys; Computational inelasticity; Finite element analysis; Return mapping algorithm; Parallel projection algorithm

1. Introduction

In the past three decades, shape memory alloys (SMAs) have become one of the most widely used active materials. The actuating features of SMAs come from the Two-Way Shape Memory Effect (TWSME) and pseudoelasticity (superelasticity). These terms refer to recoverable deformations over temperature and mechanical loading cycles, respectively. These properties, as well as the characteristics of high energy density, medium-to-high actuating frequency, and favorable mechanical properties (strength, stiffness, etc.), have made SMAs a frequent choice in a variety of applications [1].

Given these advantages, computational modeling of SMAs is an important topic of investigation. Specifically, constitutive modeling and finite element analysis of polycrystalline SMAs are of critical importance, since many

* Corresponding author.

E-mail address: zkang@mit.edu (Z. Kang).

polycrystalline SMAs tend to exhibit stable TWSMEs and superelasticity [2]. However, the complex constitutive relationship of these multifunctional materials substantially increases the difficulty of the modeling task. Various thermomechanical constitutive models for SMAs have been developed and these models are generally categorized into two groups—micromechanical models and phenomenological models. Micromechanical models focus more on describing the microscopic behaviors of different SMA crystal variants and lattice structures within the SMA family, whereas phenomenological models concentrate on their macroscopic constitutive behaviors. Sun and Hwang produced pioneering work on constitutive relationships based on micromechanical models [3,4]. Further study on the micromechanical dynamics of phase boundary motion was then conducted by Bhattacharya [5,6]. In the past decade, researchers in this area have further investigated the lattice structure of SMAs, and have sought to develop models that accurately capture the phenomena of twinning, detwinning, and single crystallization [7,8]. Micromechanics models are computationally intense, and produce a level of details that may not be necessary for all applications. This gives rise to phenomenological models, whose main challenge is to accurately model hardening during phase transformation. The exponential hardening rule is the first proposed model for NiTi SMAs [9], followed by the cosine model [10,11], the quadratic function model [12–14] and the smooth transformation model [15]. These phenomenological models formed the basis of improved constitutive models considering plasticity [16,17], creep [18–20], tension–compression asymmetry [21,22] and large deformation [23]. Efforts in this area are also directed toward introducing different representations for thermodynamic potentials, which contain internal state variables that represent the transformation state of SMAs. As a result of these phenomenological models, finite element analysis of SMAs has become more computationally efficient and accurate, with fewer tuned parameters. Indeed, experimental validation of Helmholtz free and Gibbs free potential energies, and phase diagrams have been reported [24–28].

The highly nonlinear finite element analysis requires iterative techniques to evaluate the dynamic thermal energy state of the material [29] at each time step within the simulation. Two sets of coupled differential equations (DEs) need to be solved concurrently. One set contains the global partial differential equations (PDEs) based on the momentum balance, and the other set contains the local ordinary differential equations (ODEs), which are based on the evolution relation [30]. Extensive applied mathematics research has been focused on efficiently solving these large coupled systems of DEs. The return mapping algorithm is the most widely adopted approach [31]. In each iteration of this nested approach, one first updates the internal/constitutive state variables for the given state of strain by fully solving local constitutive DEs at each Gauss point. Then the displacement is updated by solving the linear system derived from the global equilibrium PDEs. This procedure treats the global–local DEs as two coupled convex mathematical programming problems, adding an extra layer of computational complexity and inefficiency. In an effort to improve the efficiency of the classical return mapping method at each iteration, Simo et al. solved the Gauss point DE and global PDE updates as one monolithic system thereby they did not have to fully solve local DEs [32]. Kulkarni et al. then used a Schur-complement procedure to solve the monolithic system [33]. Similar Newton Schur procedures are also reported to solve other problems with history dependent material responses [34–39].

While similar procedures have been implemented for elastoplastic systems, researchers have yet to apply the Newton Schur tool to SMAs due to the complicated schemes required for updating the internal, i.e. constitutive, state variables. To this end, we develop what we deem the *parallel projection* algorithm. The algorithm provides identical results as the classical return mapping algorithm, with measurable benefits, notably the ability to take of larger load steps and achieve computational savings. Further, we propose schemes for updating the internal state variables that unify the popular closest-point and cutting plane methods. In the sections that follow, we explain the implementation and mathematical rationale behind the parallel projection algorithm, and demonstrate the algorithm via a series of example problems over one- and three-dimensional domains.

2. Phenomenological constitutive relationship of SMAs

The unique properties (TWSME and superelasticity) of SMAs are triggered by a non-diffusional phase transformation, which is caused by latent heat exchange between two stable phases. These two phases, martensite (M) and austenite (A), are each characterized by their distinctive molecular lattice structures. In this paper, we use the phenomenological constitutive models derived by Boyd and Lagoudas [12] and later described by Lagoudas in 2008 [14]. This model uses the Gibbs-free energy,

$$G = u - \frac{1}{\rho} \boldsymbol{\sigma} : \boldsymbol{\varepsilon} - sT \quad (1)$$

to derive the constitutive relationship. Here, $\boldsymbol{\epsilon}$ is the total small strain tensor, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, T is the temperature, ρ is the density, u is the specific internal energy and s is the specific entropy which define the energy state of the thermodynamic system. The operator “:” refers to double dot product of tensors. The choice of Gibbs free energy makes it easier to represent the constitutive model in the global momentum balance relationship used in the finite element analysis. We assume that the energy function G depends on the martensite volume fraction ξ , the stress $\boldsymbol{\sigma}$, the temperature T , and the transformation strain $\boldsymbol{\epsilon}^t$. It is further assumed to be expressed as the sum of bulk G_b and mixing G_m energies such that

$$G = G_b + G_m \tag{2}$$

where

$$G_b(\xi, \boldsymbol{\sigma}, T) = -\frac{1}{2\rho} \boldsymbol{\sigma} : \boldsymbol{S} : \boldsymbol{\sigma} - \frac{1}{\rho} \boldsymbol{\sigma} : \boldsymbol{\alpha} (T - T_0) + c \left[(T - T_0) - T \ln \frac{T}{T_0} \right] - s_0 T + u_0 \tag{3}$$

and

$$G_m(\xi, \boldsymbol{\sigma}, T, \boldsymbol{\epsilon}^t) = -\frac{1}{\rho} \boldsymbol{\sigma} : \boldsymbol{\epsilon}^t + \frac{1}{\rho} f(\xi) \tag{4}$$

In the above f is the experimentally obtained transformation hardening function that defines the specific energy due to mixing, T_0 refers to reference temperature for thermal expansion, \boldsymbol{S} is the compliance tensor and $\boldsymbol{\alpha}$ is the thermal expansion tensor. For isotropic SMAs, the Young’s modulus E is the only variant in the evolution of the compliance tensor \boldsymbol{S} . As such, we express \boldsymbol{S} (in Voigt notation) via the compliance modulus $S = 1/E$ and the Poisson’s ratio ν as

$$\boldsymbol{S} = S \boldsymbol{\mathcal{C}} \tag{5}$$

$$\boldsymbol{\mathcal{C}} = \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix}$$

Consistent with our isotropic assumption, the thermal expansion tensor $\boldsymbol{\alpha}$ is given by

$$\boldsymbol{\alpha} = \alpha \times \text{diag} (1, 1, 1, 0, 0, 0) \tag{6}$$

where α is the thermal expansion coefficient.

The symbols c , s_0 and u_0 in Eq. (3) represent the effective specific heat, effective specific entropy and effective specific internal energy. The values of the above physical quantities are expressed as a volume average of their values in their martensite (M) and austenite (A) phases, i.e.

$$\begin{aligned} S &= S^A + \xi(S^M - S^A) = S^A + \xi \Delta S \\ \alpha &= \alpha^A = \alpha^M \\ c &= c^A = c^M \\ s_0 &= s_0^A + \xi(s_0^M - s_0^A) = s_0^A + \xi \Delta s_0 \\ u_0 &= u_0^A + \xi(u_0^M - u_0^A) = u_0^A + \xi \Delta u_0 \end{aligned} \tag{7}$$

Note that α and c are assumed to be constant

We derive the strain $\boldsymbol{\epsilon}$ by substituting Eqs. (1) and (2) into the second law of thermodynamics.

$$\frac{1}{\rho} \boldsymbol{\sigma} : \dot{\boldsymbol{\epsilon}} - (\dot{u} + \dot{s}T) \geq 0 \tag{8}$$

Applying the Coleman–Noll approach [30], we find that the total strain is composed of three parts, i.e.

$$\boldsymbol{\epsilon} = \boldsymbol{\epsilon}^e + \boldsymbol{\epsilon}^{th} + \boldsymbol{\epsilon}^t \tag{9}$$

In the above we have the usual elastic strain $\boldsymbol{\epsilon}^e = \boldsymbol{S} : \boldsymbol{\sigma}$ and pure thermal expansion strain $\boldsymbol{\epsilon}^{th} = \boldsymbol{\alpha}(T - T_0)$, and the transformation strain $\boldsymbol{\epsilon}^t$, which is defined via a phenomenological model. Now we invert the relation to obtain

the familiar looking result

$$\begin{aligned} \boldsymbol{\sigma} &= \mathbf{S}^{-1} : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^{th} - \boldsymbol{\varepsilon}^t) \\ &= \mathbf{S}^{-1} : (\boldsymbol{\varepsilon} - \boldsymbol{\alpha}(T - T_0) - \boldsymbol{\varepsilon}^t) \end{aligned} \tag{10}$$

An evolution relation (flow rule) [40] is proposed to build a relationship between the transformation strain $\boldsymbol{\varepsilon}^t$ and the martensite volume fraction ξ . Thus we only have three internal variables ξ , $\boldsymbol{\sigma}$ and T to consider in the phenomenological model. The $\xi - \boldsymbol{\varepsilon}^t$ relation is governed by the experimentally obtained transformation tensor \mathbf{A} ; we use the isotropic relationship proposed by Boyd and Lagoudas [12].

$$\begin{aligned} \dot{\boldsymbol{\varepsilon}}^t &= \mathbf{A} \dot{\xi} \\ \mathbf{A} &= \begin{cases} \frac{3}{2} H \frac{\boldsymbol{\sigma}_s}{\sigma_s^{eff}} & \dot{\xi} > 0 \\ H \frac{\boldsymbol{\varepsilon}_{t-r}}{\varepsilon_{t-r}^{eff}} & \dot{\xi} < 0 \end{cases} \end{aligned} \tag{11}$$

where H is the maximum transformation strain of the SMA material. In the forward transformation, i.e. austenite (A) to martensite (M) transformation for which $\dot{\xi} > 0$, $\boldsymbol{\sigma}_s$ is the deviatoric stress tensor and σ_s^{eff} is its associated effective (von Mises) stress. In the reverse transformation, i.e. martensite (M) to austenite (A) transformation for which $\dot{\xi} < 0$, $\boldsymbol{\varepsilon}_{t-r}$ is the transformation strain tensor at the reversal point, and ε_{t-r}^{eff} is its associated effective strain.

Using the aforementioned thermodynamic relationships, the second law of thermodynamics reduces to the Clausius–Planck inequality [41].

$$\Pi \dot{\xi} \geq 0 \tag{12}$$

where

$$\Pi(\xi, \boldsymbol{\sigma}, T) = \boldsymbol{\sigma} : \mathbf{A} + \frac{1}{2} \boldsymbol{\sigma} : \Delta \mathbf{S} : \boldsymbol{\sigma} + \rho \Delta s_0 T - \rho \Delta u_0 - \frac{\partial f(\xi)}{\partial \xi} \tag{13}$$

and we recall that the thermal expansion coefficient and specific heat do not change with the phase transformation.

Note that the Clausius–Planck inequality must be satisfied for all admissible thermomechanical loading paths $\dot{\xi}$. In our study, the loading paths are constrained such that $\Phi \leq 0$, where $\Phi = |\Pi| - Y$ can be viewed as a type of yield function in which Y is a type of yield strength (transformation threshold) determined by the transformation hardening function. When $\Phi = 0$, we have two possibilities. If $\Pi - Y = 0$ then to satisfy Eq. (12), $\dot{\xi} > 0$, indicating the forward transformation; otherwise if $-\Pi - Y = 0$ then $\dot{\xi} < 0$, indicating the inverse transformation. For all other cases, i.e. for $\Phi < 0$, we have $\dot{\xi} = 0$.

$$\Phi \begin{cases} = \Pi - Y = 0 & \dot{\xi} > 0 \text{ (A} \rightarrow \text{M)} \\ = -\Pi - Y = 0 & \dot{\xi} < 0 \text{ (M} \rightarrow \text{A)} \\ < 0 & \dot{\xi} = 0 \end{cases} \tag{14}$$

We assume the martensite volume fraction ξ evolves so as to maximize the dissipation of Eq. (13), subject to the $\Phi \leq 0$ constraint. As such, we must satisfy the Kuhn–Tucker conditions

$$\Phi \leq 0, \quad \Phi \dot{\xi} = 0, \quad \dot{\xi} \begin{cases} \geq 0 & \text{if } \Pi - Y = 0 \text{ (A} \rightarrow \text{M)} \\ \leq 0 & \text{if } -\Pi - Y = 0 \text{ (M} \rightarrow \text{A)} \\ = 0 & \text{if } \Phi < 0 \end{cases} \tag{15}$$

Additionally, during either forward or reverse transformation, i.e. $\dot{\xi} \neq 0$, we must satisfy the consistency condition to stay on the “loading surface”, i.e. to maintain the $\Phi = 0$ equality [42]

$$\dot{\Phi} = \frac{\partial \Phi}{\partial \boldsymbol{\sigma}} : \dot{\boldsymbol{\sigma}} + \frac{\partial \Phi}{\partial T} : \dot{T} + \frac{\partial \Phi}{\partial \xi} : \dot{\xi} = 0 \tag{16}$$

To satisfy the Karush–Kuhn–Tucker (KKT) and the consistency conditions, the following inelastic constitutive relationship must hold

$$\begin{aligned}
 d\boldsymbol{\sigma} &= \boldsymbol{\mathcal{L}} : d\boldsymbol{\varepsilon} + \boldsymbol{\Theta} : dT \\
 \boldsymbol{\mathcal{L}} &= \begin{cases} \mathbf{S}^{-1} - \frac{\mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi \otimes \mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi}{\partial_{\boldsymbol{\sigma}} \Phi : \mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi - \partial_{\xi} \Phi} & \dot{\xi} > 0 \\ \mathbf{S}^{-1} - \frac{\mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi \otimes \mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi}{\partial_{\boldsymbol{\sigma}} \Phi : \mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi + \partial_{\xi} \Phi} & \dot{\xi} < 0 \end{cases} \\
 \boldsymbol{\Theta} &= \begin{cases} -\boldsymbol{\mathcal{L}} : \boldsymbol{\alpha} - \partial_T \Phi \frac{\mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi}{\partial_{\boldsymbol{\sigma}} \Phi : \mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi - \partial_{\xi} \Phi} & \dot{\xi} > 0 \\ -\boldsymbol{\mathcal{L}} : \boldsymbol{\alpha} - \partial_T \Phi \frac{\mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi}{\partial_{\boldsymbol{\sigma}} \Phi : \mathbf{S}^{-1} : \partial_{\boldsymbol{\sigma}} \Phi + \partial_{\xi} \Phi} & \dot{\xi} < 0 \end{cases}
 \end{aligned} \tag{17}$$

where \otimes refers to the tensor product operator, $\boldsymbol{\mathcal{L}}$ and $\boldsymbol{\Theta}$ are the continuum tangent stiffness and tangent thermal moduli.

3. Finite element analysis of SMAs

3.1. The discrete model

We now describe the combined equilibrium and constitutive DE problem as finding the kinematically admissible displacement \mathbf{d} , martensite volume fraction ξ , transformation strain $\boldsymbol{\varepsilon}^t$ and stress tensor $\boldsymbol{\sigma}$ such that

$$\begin{aligned}
 \text{Global Level :} & \quad \text{force equilibrium} & \int_{\Omega} \boldsymbol{\varepsilon}(\delta \mathbf{d}) \boldsymbol{\sigma}(\mathbf{d}, \xi) d\Omega - \int_{\Gamma^p} \delta \mathbf{d} \mathbf{p} d\Gamma^p = 0 \quad \forall \delta \mathbf{d} \\
 \text{Local Level :} & \quad \text{KKT - condition} & \Phi \leq 0 \text{ in } \Omega \\
 & & \dot{\xi} \Phi = 0 \text{ in } \Omega \\
 & \quad \text{Consistency} & \dot{\Phi} = \frac{\partial \Phi}{\partial \boldsymbol{\sigma}} : \dot{\boldsymbol{\sigma}} + \frac{\partial \Phi}{\partial T} : \dot{T} + \frac{\partial \Phi}{\partial \xi} : \dot{\xi} = 0 \text{ in } \Omega \\
 & \quad \text{flow rule} & \dot{\boldsymbol{\varepsilon}}^t = \boldsymbol{\Lambda} \dot{\xi},
 \end{aligned} \tag{18}$$

where the boundary conditions include prescribed displacements \mathbf{d}^c over Γ_d , prescribed traction \mathbf{p} over Γ^p , and prescribed temperature T over Ω . Note that the local-level problem is directly solved as an algebraic equation when SMAs behave elastically, i.e. $\dot{\xi} = 0$. When SMAs behave inelastically, i.e. $\dot{\xi} \neq 0$, the initial conditions for the local DEs are the latest internal state variables solved before SMAs start transformation.

From Eq. (18), two sets of DEs are observed — one set of PDEs is applied at the global level, to enforce the equilibrium equation. The other set of ODEs is applied at the local level to enforce the constitutive relationship. Generally, for such complicated systems of differential equations, numerical methods must be used to approximate the response. We discretize the system in the spatial domain via the finite element method and in the time domain via a backward-Euler scheme. Here each pseudo-time step corresponds to a temperature increment or a load increment. The discretized form of Eq. (18) is shown in Eq. (19).

$$\begin{aligned}
 \text{Global Level :} & \quad \mathbf{R}_{n+1} = \bigwedge_{\text{el}} [\sum_i (w \mathbf{B}^T(\mathfrak{G}_i^d) \boldsymbol{\sigma}_{n+1}) - \varpi \mathbf{N}(\mathfrak{G}_i^p) \mathbf{p}_{n+1}] \det \mathbf{J} = \mathbf{0} \\
 \text{Local Level :} & \quad \left. \begin{aligned} \mathbf{H}_{n+1} &= \begin{cases} H_{\Phi} = \Phi(\boldsymbol{\sigma}_{n+1}, T_{n+1}, \xi_{n+1}) \\ H_{\boldsymbol{\varepsilon}^t} = \boldsymbol{\varepsilon}_{n+1}^t + \boldsymbol{\Lambda}(\xi_{n+1} - \xi_n) - \boldsymbol{\varepsilon}_{n+1}^t \\ H_S = S_n + \Delta S(\xi_{n+1} - \xi_n) - S_{n+1} \\ H_{\boldsymbol{\sigma}} = \mathbf{S}_{n+1}^{-1} : [\boldsymbol{\varepsilon}_{n+1} - \boldsymbol{\alpha}(T_{n+1} - T_0) - \boldsymbol{\varepsilon}_{n+1}^t] - \boldsymbol{\sigma}_{n+1} \end{cases} \\ \mathbf{H}_{n+1} &= \mathbf{S}^{-1} : [\boldsymbol{\varepsilon}_{n+1} - \boldsymbol{\alpha}(T_{n+1} - T_0) - \boldsymbol{\varepsilon}^t] - \boldsymbol{\sigma}_{n+1} = \mathbf{0} \end{aligned} \right\} = \mathbf{0} \quad \begin{aligned} \dot{\xi} &\neq 0 \\ \dot{\xi} &= 0 \end{aligned}
 \end{aligned} \tag{19}$$

Here, \mathfrak{G}_i^d (\mathfrak{G}_i^p) are the 2D (1D) Gauss point coordinates for the area(line) integrals and w (ϖ) are the corresponding product of the differential area (length) and Gauss weights at the Gauss points, respectively. $\det \mathbf{J}$ is the determinant of the Jacobian matrix. \mathbf{N} and \mathbf{B} are shape functions used to interpolate the displacement and strain fields such that, e.g., in Voigt notation $\boldsymbol{\varepsilon}_{n+1} = \mathbf{B} \mathbf{d}_{n+1}$.¹

There are two sets of unknown states in the problem, the global state variable $\mathbf{u} = (\mathbf{d}^f, \mathbf{F}^c)$, and the local state variables $\mathbf{v} = (\xi, \boldsymbol{\varepsilon}^t, S, \boldsymbol{\sigma})$ for the inelastic case, and $\mathbf{v} = \boldsymbol{\sigma}$ for the elastic case. Here, the superscript f refers to the free, i.e. unprescribed displacement degrees-of-freedom, \mathbf{d}^f ; and the superscript c represents the unknown reaction forces \mathbf{F}^c of the prescribed, i.e. constrained displacement degrees-of-freedom.

¹ Note that H_S is the rate form of Eq. (7) and could be simplified to $H_S = S_A + \Delta S \xi_{n+1} - S_{n+1}$. However, to maintain consistency with [14] and consider experimental data may only provide the value of ΔS , we use the rate equation shown.

3.2. The parallel projection algorithm

The aforementioned discrete model yields two sets of DEs that are defined in residual form; $\mathbf{R} = 0$ is the global equilibrium residual, and $\mathbf{H} = 0$ is the local residual. The two sets of DEs are coupled through the Cauchy stress tensor $\boldsymbol{\sigma}$ and displacement \mathbf{d} . This results in a large system of nonlinear equations. For instance, for a two-dimensional (2D) domain with N nodes and G Gauss points, the FEM problem contains $2 \times N$ global equations and up to $6 \times G$ local equations, all of which must be solved concurrently using the Newton–Raphson method.

To lessen the high computational cost, one’s first instinct might be to partition the system, and hence shrink the size of the tangent matrix in the Newton–Raphson scheme. In the popular return-mapping algorithm [31], the resulting system is uncoupled as

$$\begin{aligned} \mathbf{R}(\mathbf{u}, \mathbf{v}(\mathbf{u})) &= 0 \\ \mathbf{H}(\mathbf{u}, \mathbf{v}(\mathbf{u})) &= 0 \end{aligned} \tag{20}$$

The uncoupled system is solved separately via a nested Newton–Raphson iteration consisting of two loops,

$$\begin{aligned} \text{inner loop for } \mathbf{v}^{(l+1)} : & \left\{ \begin{aligned} & \left[\frac{\partial \mathbf{H}}{\partial \mathbf{v}}(\mathbf{u}^{(k)}, \mathbf{v}^{(l)}(\mathbf{u}^{(k)})) \right] \delta \mathbf{v}^{(l)} = -\mathbf{H}(\mathbf{u}^{(k)}, \mathbf{v}^{(l)}(\mathbf{u}^{(k)})) \\ & \mathbf{v}^{(l+1)} = \mathbf{v}^{(l)} + \delta \mathbf{v}^{(l)}, l = 1, 2, 3, \dots \end{aligned} \right. \\ \text{outer loop for } \mathbf{u}^{(k+1)} : & \left\{ \begin{aligned} & \left[\frac{\partial \mathbf{R}}{\partial \mathbf{u}}(\mathbf{u}^{(k)}, \mathbf{v}^{(l+1)}) - \frac{\partial \mathbf{R}}{\partial \mathbf{v}} \left(\frac{\partial \mathbf{H}}{\partial \mathbf{v}} \right)^{-1} \frac{\partial \mathbf{H}}{\partial \mathbf{u}}(\mathbf{u}^{(k)}, \mathbf{v}^{(l+1)}) \right] \delta \mathbf{u}^{(k)} = -\mathbf{R}(\mathbf{u}, \mathbf{v}^{(l+1)}) \\ & \mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \delta \mathbf{u}^{(k)}, k = 1, 2, 3, \dots \end{aligned} \right. \end{aligned} \tag{21}$$

where l and k are the iteration counters of the inner and outer Newton–Raphson loops. The derivative $d\mathbf{v}/d\mathbf{u} = -(\partial \mathbf{H}/\partial \mathbf{v})^{-1}(\partial \mathbf{H}/\partial \mathbf{u})$ in the outer loop is obtained by differentiating the local residual in Eq. (20).

$$\frac{\partial \mathbf{H}}{\partial \mathbf{u}}(\mathbf{u}, \mathbf{v}(\mathbf{u})) + \frac{\partial \mathbf{H}}{\partial \mathbf{v}}(\mathbf{u}, \mathbf{v}(\mathbf{u})) \frac{d\mathbf{v}}{d\mathbf{u}}(\mathbf{u}) = 0 \tag{22}$$

As seen above, the local residual equation $\mathbf{H} = 0$ in the inner loop is first solved for \mathbf{v} via Newton’s method for the fixed \mathbf{u} . After convergence of the inner problem, i.e. after $|\mathbf{H}| \approx 0$, we update the displacement \mathbf{u} by solving the linear system in the outer loop. We repeat the process, i.e. of solving the inner loop residual and updating the displacement until $|\mathbf{R}| \approx 0$. Unfortunately, while the return mapping algorithm reduces the size of the tangent matrix in the outer loop, it may require more iterations than simultaneously updating \mathbf{u} and \mathbf{v} as one monolithic system. We seek a method that combines the benefits of both approaches.

In response to the aforementioned concern, we introduce the parallel projection algorithm, which updates the inner and outer equations in parallel. In this algorithm, we still partition the equation as shown in Eq. (20) to diminish the size of the tangent matrix. However, we solve the coupled equations in a cross-iterative Newton–Raphson scheme, which also contains inner and outer computations:

$$\begin{aligned} \text{inner} : & \left\{ \begin{aligned} & \left[\frac{\partial \mathbf{H}}{\partial \mathbf{v}}(\mathbf{u}^{(k)}, \mathbf{v}^{(k)}(\mathbf{u}^{(k)})) \right] \delta \mathbf{v}^{(k)} = -\mathbf{H}(\mathbf{u}^{(k)}, \mathbf{v}^{(k)}(\mathbf{u}^{(k)})) - \frac{\partial \mathbf{H}}{\partial \mathbf{u}}(\mathbf{u}^{(k)}, \mathbf{v}^{(k)}(\mathbf{u}^{(k)})) \delta \mathbf{u}^{(k-1)} \\ & \mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \delta \mathbf{v}^{(k)}, k = 1, 2, 3, \dots \end{aligned} \right. \\ \text{outer} : & \left\{ \begin{aligned} & \left[\frac{\partial \mathbf{R}}{\partial \mathbf{u}}(\mathbf{u}^{(k)}, \mathbf{v}^{(k+1)}) - \frac{\partial \mathbf{R}}{\partial \mathbf{v}} \left(\frac{\partial \mathbf{H}}{\partial \mathbf{v}} \right)^{-1} \frac{\partial \mathbf{H}}{\partial \mathbf{u}}(\mathbf{u}^{(k)}, \mathbf{v}^{(k+1)}) \right] \delta \mathbf{u}^{(k)} \\ & = -\mathbf{R}(\mathbf{u}^{(k)}, \mathbf{v}^{(k)}) + \frac{\partial \mathbf{R}}{\partial \mathbf{v}} \left(\frac{\partial \mathbf{H}}{\partial \mathbf{v}} \right)^{-1} \mathbf{H}(\mathbf{u}^{(k)}, \mathbf{v}^{(k)}) \\ & \mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \delta \mathbf{u}^{(k)} \end{aligned} \right. \end{aligned} \tag{23}$$

To begin the algorithm, i.e. for $k = 1$, we assign $\delta \mathbf{u}^{(0)} = \mathbf{0}$. Notably, (1) we do not iterate to fully resolve the inner equation for each update of the outer equation, (2) the inner and outer equations are updated sequentially such that the outer problem is solved in *parallel* with the inner problem, (3) the coefficient matrices are identical in the return-mapping and parallel projection algorithms. Fig. 1 highlights the difference between the return-mapping and parallel projection algorithms.

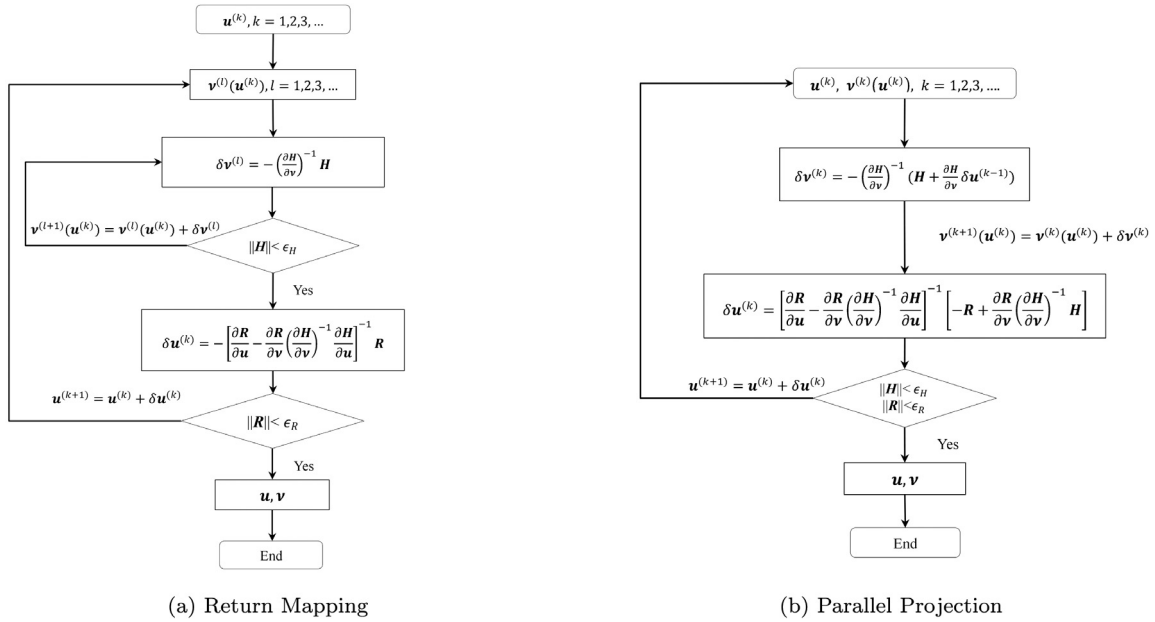


Fig. 1. Comparison between the return mapping and parallel projection algorithms.

3.3. Internal state variables update

In this section, we provide the analytical solution of the updating scheme for the internal state variable \mathbf{v} . As seen above in Eqs. (21) and (23), we need to calculate the inverse of the tangent matrix $\partial \mathbf{H} / \partial \mathbf{v}$. Depending on the thermomechanical state of the material, i.e. whether $\dot{\xi} \approx \xi_{n+1} - \xi_n$ is zero or nonzero we have,

$$\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n} = \begin{cases} \begin{bmatrix} \frac{\partial H_n^\Phi}{\partial \xi_n} & \frac{\partial H_n^\Phi}{\partial \mathbf{e}_n^t} & \frac{\partial H_n^\Phi}{\partial S_n} & \frac{\partial H_n^\Phi}{\partial \sigma_n} \\ \frac{\partial H_n^{\mathbf{e}^t}}{\partial \xi_n} & \frac{\partial H_n^{\mathbf{e}^t}}{\partial \mathbf{e}_n^t} & \frac{\partial H_n^{\mathbf{e}^t}}{\partial S_n} & \frac{\partial H_n^{\mathbf{e}^t}}{\partial \sigma_n} \\ \frac{\partial H_n^S}{\partial \xi_n} & \frac{\partial H_n^S}{\partial \mathbf{e}_n^t} & \frac{\partial H_n^S}{\partial S_n} & \frac{\partial H_n^S}{\partial \sigma_n} \\ \frac{\partial H_n^\sigma}{\partial \xi_n} & \frac{\partial H_n^\sigma}{\partial \mathbf{e}_n^t} & \frac{\partial H_n^\sigma}{\partial S_n} & \frac{\partial H_n^\sigma}{\partial \sigma_n} \end{bmatrix} = \begin{bmatrix} \partial_\xi \Phi_n & \mathbf{0}_{6 \times 6} & 0 & \partial_\sigma \Phi_n^T \\ \mathbf{A}_n & -\mathbf{I}_{6 \times 6} & \mathbf{0}_{6 \times 1} & \partial_\sigma \mathbf{A}_n : (\xi_n - \xi_{n-1}) \\ \Delta S & \mathbf{0}_{1 \times 6} & -\mathbf{I}_{1 \times 1} & \mathbf{0}_{1 \times 6} \\ \mathbf{0}_{6 \times 1} & -\mathbf{S}_n^{-1} & -\mathbf{S}_n^{-1} : \mathbf{I}_{6 \times 6} : \sigma_n & -\mathbf{I}_{6 \times 6} \end{bmatrix} \\ \begin{bmatrix} \frac{\partial H_n}{\partial \sigma_n} \end{bmatrix} = [-\mathbf{I}_{6 \times 6}] \end{cases} \quad \begin{matrix} \dot{\xi} \neq 0 \\ \dot{\xi} = 0 \end{matrix} \quad (24)$$

where the superscript T denotes the transpose of vectors and matrices.

From Eq. (24), we notice that the magnitudes S^{-1} and \mathbf{S}^{-1} are much larger than the other internal variables. This results in numerical difficulties, since $\partial \mathbf{H} / \partial \mathbf{v}$ is close to singular. To lessen the effect of the ill-conditioning, we use Schur-complement to derive the analytical formula for the internal state variable update $\delta \mathbf{v}$ of Eqs. (21) and (23). Similar Schur procedures are also reported in [33,43–45] for plasticity and ductile fracture problems. To do this, we break the update into two parts. First we compute $\Delta \mathbf{v}_{n+1}^{(k)} = -(\partial \mathbf{H}_{n+1}^{(k)} / \partial \mathbf{v}_{n+1}^{(k)})^{-1} \mathbf{H}_{n+1}^{(k)}$, which appears in both the return mapping and parallel projection algorithms.

$$\Delta \mathbf{v}_{n+1}^{(k)} = \begin{bmatrix} \delta \xi_{n+1}^{(k)} \\ \delta \mathbf{e}_{n+1}^{t*(k)} \\ \delta S_{n+1}^{(k)} \\ \delta \sigma_{n+1}^{(k)} \end{bmatrix} = \begin{bmatrix} \Delta \xi_{n+1}^{*(k)} + \vartheta_{n+1}^{(k)} \\ \Delta \mathbf{e}_{n+1}^{t*(k)} + \tilde{\mathbf{A}}_{n+1}^{(k)} : \vartheta_{n+1}^{(k)} + \tilde{\Psi}_{n+1}^{(k)} \\ \Delta S_{n+1}^{*(k)} + H_{S_{n+1}}^{(k)} + \Delta S \cdot \vartheta_{n+1}^{(k)} \\ \Delta \sigma_{n+1}^{*(k)} + \mathfrak{L}_{n+1}^{(k)} : \mathbf{S}_{n+1}^{(k)} : \Psi_{n+1}^{(k)} \end{bmatrix} \quad (25)$$

where

$$\begin{aligned}
 \Delta \xi_{n+1}^{*(k)} &= \frac{\Phi_{n+1}^{(k)} - \partial_\sigma \Phi_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : \mathbf{H} \mathbf{e}_{n+1}^{(k)}}{\pm \partial_\sigma \Phi_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} - \partial_\xi \Phi_{n+1}^{(k)}} & (+ : \dot{\xi} > 0, - : \dot{\xi} < 0) \\
 \Delta \sigma_{n+1}^{*(k)} &= \zeta_{n+1}^{(k)-1} : [-\mathbf{H} \mathbf{e}_{n+1}^{(k)} \mp \Delta \xi_{n+1}^{*(k)} : \partial_\sigma \Phi_{n+1}^{(k)}] & (- : \dot{\xi} > 0, + : \dot{\xi} < 0) \\
 \Delta \mathbf{e}_{n+1}^{*(k)} &= -\mathbf{S}_{n+1}^{(k)} : \Delta \sigma_{n+1}^{*(k)} - (\Delta S : \mathbf{I} : \sigma_{n+1}^{(k)}) \Delta \xi_{n+1}^{*(k)} & (26) \\
 \Delta S_{n+1}^{*(k)} &= \Delta S \Delta \xi_{n+1}^{*(k)} \\
 \Psi_{n+1}^{(k)} &= \mathbf{H} \sigma_{n+1}^{(k)} - \mathbf{S}_{n+1}^{(k)-1} : \mathbf{I} : \sigma_{n+1}^{(k)} : \mathbf{H} S_{n+1}^{(k)} \\
 \tilde{\Psi}_{n+1}^{(k)} &= \mathbf{S}_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : \partial_\sigma \mathbf{A}_{n+1}^{(k)} : (\xi_{n+1}^{(k)} - \xi_n) : \Psi_{n+1}^{(k)} \\
 \vartheta_{n+1}^{(k)} &= \frac{\partial_\sigma \Phi_{n+1}^{(k)}}{\pm \partial_\sigma \Phi_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} - \partial_\xi \Phi_{n+1}^{(k)}} : \zeta_{n+1}^{(k)-1} : \mathbf{S}_{n+1}^{(k)} : \Psi_{n+1}^{(k)} & (+ : \dot{\xi} > 0, - : \dot{\xi} < 0) \\
 \zeta_{n+1}^{(k)} &= \begin{cases} \mathbf{S}_{n+1}^{(k)} + \partial_\sigma \mathbf{A}_{n+1}^{(k)} : (\xi_{n+1}^{(k)} - \xi_n) & \dot{\xi} > 0 \\ \mathbf{S}_{n+1}^{(k)} & \dot{\xi} < 0 \end{cases}
 \end{aligned}$$

Next we evaluate $\Delta \mathbf{v}_{n+1}^{*(k)} = -(\partial \mathbf{H}_{n+1}^{(k)} / \partial \mathbf{v}_{n+1}^{(k)})^{-1} (\partial \mathbf{H}_{n+1}^{(k)} / \partial \mathbf{u}_{n+1}^{(k)}) \delta \mathbf{u}_{n+1}^{(k-1)}$, which only appears in the parallel projection algorithm.

$$\Delta \mathbf{v}_{n+1}^{*(k)} = \begin{bmatrix} \frac{\partial_\sigma \Phi_{n+1}^{(k)T} : \zeta_{n+1}^{(k)-1}}{\pm \partial_\sigma \Phi_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} - \partial_\xi \Phi_{n+1}^{(k)}} \\ \frac{\tilde{\mathbf{A}}_{n+1}^{(k)} : \partial_\sigma \Phi_{n+1}^{(k)T}}{\pm \partial_\sigma \Phi_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} - \partial_\xi \Phi_{n+1}^{(k)}} : \zeta_{n+1}^{(k)-1} - \mathbf{S}_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} + \mathbf{I}_{6 \times 6} \\ \frac{\Delta S : \partial_\sigma \Phi_{n+1}^{(k)T}}{\pm \partial_\sigma \Phi_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} - \partial_\xi \Phi_{n+1}^{(k)}} : \zeta_{n+1}^{(k)-1} \\ \mathbf{S}_{n+1}^{(k-1)} \end{bmatrix} \mathbf{B} : \delta \mathbf{u}_{n+1}^{(k-1)} \quad (27)$$

where we use + if $\dot{\xi} > 0$ and - if $\dot{\xi} < 0$. For all updates, we define

$$\tilde{\mathbf{A}}_{n+1}^{(k)} = \mathbf{S}_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : (\mathbf{A}_{n+1}^{(k)} - \partial_\sigma \mathbf{A}_{n+1}^{(k)} : (\xi_{n+1}^{(k)} - \xi_n) : \mathbf{S}_{n+1}^{(k)-1} : \mathbf{I}_{6 \times 6} : \sigma_{n+1}^{(k)} : \Delta S) \quad (28)$$

Note that during the inverse transformation, i.e. $\dot{\xi} < 0$, $\tilde{\mathbf{A}}_{n+1}^{(k)}$ degenerates to the transformation tensor $\mathbf{A}_{n+1}^{(k)}$, since $\zeta_{n+1}^{(k)} = \mathbf{S}_{n+1}^{(k)}$ and $\partial_\sigma \mathbf{A}_{n+1}^{(k)} = 0$. Ultimately, the local Newton–Raphson updated increment $\delta \mathbf{v}_n^{(k+1)}$ shown in Eqs. (21) and (23) is evaluated as

$$\delta \mathbf{v}_{n+1}^{(k+1)} = \begin{cases} \Delta \mathbf{v}_{n+1}^{(k)} & \text{(Return Mapping)} \\ \Delta \mathbf{v}_{n+1}^{(k)} + \Delta \mathbf{v}_{n+1}^{*(k)} & \text{(Parallel Projection)} \end{cases} \quad (29)$$

A detailed derivation of these quantities is provided in [Appendix](#).

With these formulae, we obtain the consistent tangent stiffness modulus $\mathfrak{L}_{n+1}^{(k)}$ to update the displacement, i.e. $(\partial \mathbf{R} / \partial \mathbf{u}) - (\partial \mathbf{R} / \partial \mathbf{v})(\partial \mathbf{H} / \partial \mathbf{v})^{-1} (\partial \mathbf{H} / \partial \mathbf{u}) = \sum_{\mathfrak{S}_i} w \mathbf{B}^T \mathfrak{L}_{n+1}^{(k)} \mathbf{B} \det \mathbf{J}$, where

$$\mathfrak{L}_{n+1}^{(k)} = \zeta_{n+1}^{(k)-1} - \frac{\zeta_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} \otimes \zeta_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)}}{\partial_\sigma \Phi_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} \mp \partial_\xi \Phi_{n+1}^{(k)}} \quad (+ : \dot{\xi} > 0, - : \dot{\xi} < 0) \quad (30)$$

and ζ_n is defined in Eq. (26). Note the similarity in the form of the ‘‘continuum’’ tangent operator of Eq. (17) and the ‘‘consistent’’ tangent operator of (30).

Now, we compare our Newton–Raphson updating scheme for solving the local residual \mathbf{H} to the radial return method for plasticity [31], and with the popular closest-point [40] and cutting plane methods [14]. The Newton–Raphson scheme can be simplified by strictly enforcing some of the local residual equations in $\mathbf{H} = [H_\phi, \mathbf{H}_{\mathbf{e}^t}, H_S, \mathbf{H}_\sigma]^T = 0$ to eliminate their associated internal variables ξ , \mathbf{e}^t , S , σ , cf. [Table 1](#).

Summarizing all of the above methods, i.e. the Newton–Raphson, radial return, closest-point and cutting plane methods, are applicable to both the return mapping and parallel projection algorithms. The difference being that local iterations are performed in the return mapping algorithm, but not in the parallel projection algorithm.

Table 1

Comparison of techniques to update internal state variables in Δv .

Newton–Raphson Method	Radial return Method
	Strictly enforced: $\mathbf{H}_\sigma = 0$ to eliminate σ
Increment : $\delta \xi_{n+1}^{(k)} = \Delta \xi_{n+1}^{*(k)} + \vartheta_{n+1}^{(k)}$ $\delta \mathbf{e}_{n+1}^{(k)} = \Delta \mathbf{e}_{n+1}^{*(k)} + \mathbf{\Lambda}_{n+1}^{(k)} : \vartheta_{n+1}^{(k)} + \tilde{\Psi}_{n+1}^{(k)}$ $\delta S_{n+1}^{(k)} = \Delta S_{n+1}^{*(k)} + H_{S_{n+1}} + \Delta S \cdot \vartheta_{n+1}^{(k)}$ $\delta \sigma_{n+1}^{(k)} = \Delta \sigma_{n+1}^{*(k)} + \mathfrak{L}_{n+1}^{(k)} : S_{n+1}^{(k)} : \Psi_{n+1}^{(k)}$ Consistent tangent operator : $\mathfrak{L}_n = \begin{cases} \zeta_n^{-1} - \frac{\zeta_n^{-1} : \partial_\sigma \Phi_n \otimes \zeta_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : \zeta_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi_n} & \dot{\zeta} > 0 \\ S_n^{-1} - \frac{S_n^{-1} : \partial_\sigma \Phi_n \otimes S_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : S_n^{-1} : \partial_\sigma \Phi_n + \partial_\xi \Phi_n} & \dot{\zeta} < 0 \end{cases}$	Increment : $\delta \xi_{n+1}^{(k)} = \Delta \xi_{n+1}^{*(k)} + \vartheta_{n+1}^{(k)}$ $\delta \mathbf{e}_{n+1}^{(k)} = \Delta \mathbf{e}_{n+1}^{*(k)} + \mathbf{\Lambda}_{n+1}^{(k)} : \vartheta_{n+1}^{(k)} + \tilde{\Psi}_{n+1}^{(k)}$ $\delta S_{n+1}^{(k)} = \Delta S_{n+1}^{*(k)} + H_{S_{n+1}} + \Delta S \cdot \vartheta_{n+1}^{(k)}$ $\vartheta_{n+1}^{(k)} = \frac{\partial_\sigma \Phi_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : S_{n+1}^{(k)} : \Psi_{n+1}^{(k)}}{\pm \partial_\sigma \Phi_{n+1}^{(k)} : \zeta_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} - \partial_\xi \Phi_{n+1}^{(k)}}$ $\Psi_{n+1}^{(k)} = -S_{n+1}^{(k)-1} : \sigma_{n+1}^{(k)} : H_{S_{n+1}}^{(k)}$ $\sigma_{n+1}^{(k+1)} = S_{n+1}^{(k+1)-1} : [\mathbf{e}_{n+1} - \alpha(T_{n+1} - T_0) - \mathbf{e}_{n+1}^{(k+1)}]$ Consistent tangent operator : $\mathfrak{L}_n = \begin{cases} \zeta_n^{-1} - \frac{\zeta_n^{-1} : \partial_\sigma \Phi_n \otimes \zeta_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : \zeta_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi_n} & \dot{\zeta} > 0 \\ S_n^{-1} - \frac{S_n^{-1} : \partial_\sigma \Phi_n \otimes S_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : S_n^{-1} : \partial_\sigma \Phi_n + \partial_\xi \Phi_n} & \dot{\zeta} < 0 \end{cases}$
Closest-Point Method [40]	Cutting Plane Method [14]
Strictly enforced : $H_S = 0$, $\mathbf{H}_\sigma = 0$ to eliminate S and σ	Strictly enforced : $H_S = 0$, $\mathbf{H}_{\mathbf{e}^t} = 0$, $\mathbf{H}_\sigma = 0$ to eliminate S , \mathbf{e}^t and σ
Increment : $\delta \xi_{n+1}^{(k)} = \Delta \xi_{n+1}^{*(k)}$ $\delta \mathbf{e}_{n+1}^{(k)} = \Delta \mathbf{e}_{n+1}^{*(k)}$ $\delta S_{n+1}^{(k)} = \Delta S \Delta \xi_{n+1}^{*(k)}$ $\sigma_{n+1}^{(k+1)} = S_{n+1}^{(k+1)-1} : [\mathbf{e}_{n+1} - \alpha(T_{n+1} - T_0) - \mathbf{e}_{n+1}^{(k+1)}]$ Consistent tangent operator : $\mathfrak{L}_n = \begin{cases} \zeta_n^{-1} - \frac{\zeta_n^{-1} : \partial_\sigma \Phi_n \otimes \zeta_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : \zeta_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi_n} & \dot{\zeta} > 0 \\ S_n^{-1} - \frac{S_n^{-1} : \partial_\sigma \Phi_n \otimes S_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : S_n^{-1} : \partial_\sigma \Phi_n + \partial_\xi \Phi_n} & \dot{\zeta} < 0 \end{cases}$	Increment : $\delta \xi_{n+1}^{(k)} = \frac{\Phi_{n+1}^{(k)}}{\pm \partial_\sigma \Phi_{n+1}^{(k)} : S_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} - \partial_\xi \Phi_{n+1}^{(k)}}$ $\delta \mathbf{e}_{n+1}^{(k)} = \mathbf{\Lambda}_{n+1}^{(k)} : \delta \xi_{n+1}^{(k)}$ $\delta S_{n+1}^{(k)} = \Delta S \delta \xi_{n+1}^{(k)}$ $\sigma_{n+1}^{(k)} = S_{n+1}^{(k+1)-1} : [\mathbf{e}_{n+1} - \alpha(T_{n+1} - T_0) - \mathbf{e}_{n+1}^{(k+1)}]$ Consistent tangent operator : $\mathfrak{L}_n = \begin{cases} S_n^{-1} - \frac{S_n^{-1} : \partial_\sigma \Phi_n \otimes S_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : S_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi_n} & \dot{\zeta} > 0 \\ S_n^{-1} - \frac{S_n^{-1} : \partial_\sigma \Phi_n \otimes S_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : S_n^{-1} : \partial_\sigma \Phi_n + \partial_\xi \Phi_n} & \dot{\zeta} < 0 \end{cases}$

To better understand the implementation of the parallel projection algorithm, and its ability to be combined with various local updating techniques, we provide a pseudo-code implementation with the closest-point updating technique in Table 2.

4. Example problems

To illustrate the capability of the parallel projection algorithm and the updating schemes for the local residuals, various simulations of TWSMEs and superelasticity of SMAs are provided. All of the simulations are based on NiTi50, whose properties are provided in Table 3. The finite element analysis in all cases is carried out via the parallel projection and return mapping algorithms. For both algorithms, the global residual \mathbf{R} convergence tolerance is $e_R = 10^{-6}$, and the local residual \mathbf{H} convergence tolerance is $e_H = 10^{-6}$. The simulations are conducted via commercial software MATLAB(R) 2016b, on a workstation platform equipped with an Intel(R) Core i5-6500 CPU and 8 GB memory (RAM).

4.1. 1D SMA problem

We start the discussion with 1D SMA problems, because their constitutive model can be analytically evaluated, which enables a direct verification of different algorithms. The boundary conditions and parameters of the simulated 1D bar are shown in Fig. 2. The length of the structure is divided into a mesh of ten equally-sized bar finite elements, with a cross section of 0.1 m².

Table 2

Parallel projection — closest-point algorithm.

Initialization: initialize external and internal state variables at pseudo-time step $n + 1$

I. Let $k = 1, T_{n+1} = T_n + n \cdot dT, \mathbf{F}_{n+1} = \mathbf{F}_n + n \cdot d\mathbf{F}, \mathbf{d}_{n+1}^{(1)} = \mathbf{d}_n$
 $\xi_{n+1}^{(1)} = \xi_n, \mathbf{e}_{n+1}^{t(1)} = \mathbf{e}_n^t, S_{n+1}^{(1)} = S_n$

II. Loop over Gauss points, calculate total strain $\mathbf{e}_{n+1}^{(k)} = \mathbf{B}\mathbf{d}_{n+1}^{(k)}$ and go to inner loop

Inner Update: Update internal state variables for one iteration step

1. Calculate trial stress $\boldsymbol{\sigma}_{n+1}^{(k)}$ using $S_{n+1}^{(k)}, \mathbf{e}_{n+1}^{(k)}$ and $\mathbf{e}_{n+1}^{t(k)}$ from previous iteration, and evaluate the local residual $\mathbf{H}^{(k)} = [\mathbf{H}_\Phi^{(k)}, \mathbf{H}_{\mathbf{e}^t}^{(k)}]$
 $\boldsymbol{\sigma}_{n+1}^{(k)} = S_{n+1}^{(k)-1} : [\mathbf{e}_{n+1}^{(k)} - \alpha(T_{n+1} - T_0) - \mathbf{e}_{n+1}^{t(k)}]$
 $\mathbf{H}_\Phi^{(k)} = \Phi[\boldsymbol{\sigma}_{n+1}^{(k)}, T_{n+1}, \xi_{n+1}^{(k)}]$
 $\mathbf{H}_{\mathbf{e}^t}^{(k)} = \mathbf{e}_n^t + \boldsymbol{\Lambda}_{n+1}^{(k)} [\xi_{n+1}^{(k)} - \xi_n] - \mathbf{e}_{n+1}^{t(k)}$
 If $|\mathbf{H}_\Phi^{(k)}| \leq e_1^H$ and $\|\mathbf{H}_{\mathbf{e}^t}^{(k)}\| \leq e_2^H$
 Compute consistent tangent stiffness modulus $\mathfrak{L}^{(k+1)}$, update internal variables with the current values and go to Outer Loop
 Else
 Continue to step 2

2. Compute increment of inner state variables $-(\partial\mathbf{H}/\partial\mathbf{v})^{-1}\mathbf{H}$
 $\delta\xi_{n+1}^{(k)} = \frac{\Phi_{n+1}^{(k)} - \partial_\sigma \Phi_{n+1}^{(k)} \xi_{n+1}^{(k)-1} : \mathbf{H}_{\mathbf{e}^t}^{(k)}}{\pm \partial_\sigma \Phi_{n+1}^{(k)} : \xi_{n+1}^{(k)-1} : \partial_\sigma \Phi_{n+1}^{(k)} - \partial_\xi \Phi_{n+1}^{(k)}} \quad (+ : \dot{\xi} > 0, - : \dot{\xi} < 0)$
 $\Delta\boldsymbol{\sigma}_{n+1}^{*(k)} = S_{n+1}^{(k)-1} : [-\mathbf{H}_{\mathbf{e}^t}^{(k)} \mp \Delta\xi_{n+1}^{(k)} \partial_\sigma \Phi_{n+1}^{(k)}] \quad (- : \dot{\xi} > 0, + : \dot{\xi} < 0)$
 $\delta\mathbf{e}_{n+1}^{t(k)} = -S_{n+1}^{(k)} : \Delta\boldsymbol{\sigma}_{n+1}^{*(k)} - (\Delta S : \boldsymbol{\sigma}_{n+1}^{(k)}) \delta\xi_{n+1}^{(k)}$

3. Update martensite volume fraction, transformation strain and compliance modulus
 $\xi_{n+1}^{(k+1)} = \xi_{n+1}^{(k)} + \delta\xi_{n+1}^{(k)}$
 $\mathbf{e}_{n+1}^{t(k+1)} = \mathbf{e}_{n+1}^{t(k)} + \delta\mathbf{e}_{n+1}^{t(k)}$
 $S_{n+1}^{(k+1)} = S_{n+1}^{(k)} + \Delta S \delta\xi_{n+1}^{(k)}$
 $\boldsymbol{\sigma}_{n+1}^{(k+1)} = S_{n+1}^{(k+1)-1} : [\mathbf{e}_{n+1}^{(k)} - \alpha(T_{n+1} - T_0) - \mathbf{e}_{n+1}^{t(k+1)}]$

4. Compute consistent tangent stiffness modulus $\mathfrak{L}^{(k)}$ and go to Outer Update

Outer Update: Update displacement

III. Assemble global tangent stiffness matrix \mathbf{K} , internal force \mathbf{F}_{int}
 $\mathbf{K} = \bigwedge_{el} \sum_i w \mathbf{B}_{\mathfrak{G}_i}^T \mathfrak{L}_{\mathfrak{G}_i}^{(k)} \mathbf{B}_{\mathfrak{G}_i} \det \mathbf{J}, \mathbf{F}_{int}^{(k)} = \bigwedge_{el} \sum_i \varpi \mathbf{B}_{\mathfrak{G}_i}^T \boldsymbol{\sigma}_{\mathfrak{G}_i}^{(k)} \det \mathbf{J}$

IV. Calculate the global residual
 $\mathbf{R}^{(k)} = \mathbf{F}_{int}^{(k)} - \mathbf{F}_{n+1}$

V. Evaluate the convergence condition
 If $\|\mathbf{R}^{(k)}\| \leq e^R$ and for all Gaussian points $\|\mathbf{H}_{\mathfrak{G}}^{(k)}\| \leq e_G^H$
 Finalize the external variables \mathbf{u}_{n+1} and internal variables \mathbf{v}_{n+1}
 Let $n = n + 1$, return to step I until the end of pseudo-time step
 Else
 Update the displacement field, $\mathbf{d}_{n+1}^{(k+1)} = \mathbf{d}_{n+1}^{(k)} - \mathbf{K}^{-1} [\mathbf{R}^{(k)} + \frac{\partial \mathbf{R}}{\partial \mathbf{v}} (\frac{\partial \mathbf{H}}{\partial \mathbf{v}})^{-1} \mathbf{H}^{(k)}]$
 Let $k = k + 1$, return to Inner Update

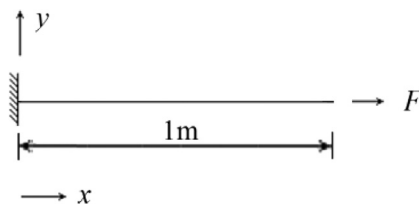


Fig. 2. Boundary conditions for the 1D bar problem with various hardening models.

For the simulation of TWSME, a temperature cycle ranging from 180 to 300 K is applied to the structure; the structure is stress-free as no external loads are applied. When simulating the superelasticity of SMAs, the structure

Table 3
Material properties of NiTi50.

	Austenite (A)	Martensite (M)
Young's modulus E (Pa)	32.5×10^9	23.0×10^9
Thermal coefficient α (K^{-1})	22.0×10^{-6}	22.0×10^{-6}
Specific heat c (J/kgK)	400.0	400.0
Transformation start temperature A_s and M_s (K)	241	226
Transformation finish A_f and M_f temperature (K)	290	194
Highest transformation strain H	0.033	
Material density ρ (kg/m^3)	6500	
Reference temperature T_0 (K)	300	
Entropy difference $\rho\Delta s_0$ ($\text{J}/\text{m}^3\text{K}$)	-11.55×10^4	

Table 4
Approximate maximum loading step for simulating superelasticity.

Methods	Newton–Raphson	Radial Return	Closest-Point	Cutting Plane
Return Mapping	9×10^8 N	9×10^8 N	9×10^8 N	9×10^8 N
Parallel Projection	3×10^9 N	3×10^9 N	3×10^9 N	3×10^9 N

is subjected to an axial loading cycle ranging from 5×10^6 N to 5×10^9 N, at a uniform temperature 310 K. Note that the phase transformation evolves uniformly, since the structure undergoes uniaxial stress.

4.1.1. Accuracy of local residual updating scheme implemented in the parallel projection algorithm

To compare the accuracy and stability of the four different updating schemes for the local residuals, i.e. the Newton Raphson, radial return, closest-point and cutting plane methods, we compare their convergences when embedded in the return mapping and parallel projection algorithms, cf. Figs. 3 and 4. The comparison is conducted in the TWSME case with the analytic solution [14]

$$\begin{aligned}\xi &= \frac{1}{\rho b^M} [|\sigma|H + \frac{1}{2}\Delta S\sigma^2 + \rho\Delta s_0(T - M_s)] \\ \epsilon^t &= \frac{H\text{sgn}(\sigma)}{\rho b^M} [|\sigma|H + \frac{1}{2}\Delta S\sigma^2 + \rho\Delta s_0(T - M_s)]\end{aligned}\quad (31)$$

which uses the quadratic polynomial hardening function, and $b^M = -\Delta s_0(M_s - M_f)$. The convergence is compared at the temperature of 220 K, where the error is computed as

$$\begin{aligned}\epsilon_\xi &= |\xi_{\text{ANA}} - \xi_{\text{NUM}}| \\ \epsilon_{\epsilon^t} &= |\epsilon^t_{\text{ANA}} - \epsilon^t_{\text{NUM}}|\end{aligned}\quad (32)$$

Here, the subscripts ANA and NUM refer to the analytical and numerical solutions, respectively.

In Figs. 3(a) and 4(a), the arrow refers to the maximum temperature increment that we use with each method to attain convergence. These methods solve the same equations but their radii of convergence vary. Indeed, as seen in Figs. 3 and 4, eliminating variables to simplify the local DEs hinders the convergence, e.g. the cutting plane method. In addition, we also notice that the parallel projection algorithm generally allows for larger temperature increments than the return mapping algorithm which strictly enforces the residual equation $\mathbf{H} = 0$ at each global iteration. This finding is consistent in simulating the superelasticity, cf. Table 4

4.1.2. Adaptability of the parallel projection algorithm with various constitutive models

To demonstrate that the Parallel Projection algorithm is able to handle different constitutive relationships (i.e. hardening models) of SMAs, we repeat the above study. However, we only use the closest-point scheme and replace the hardening function (f) with the most popular models proposed in the past two decades. Due to differences in fitting techniques for Differential Scanning Calorimetry (DSC)-obtained experimental data [12], various hardening functions have been created to model the hysteresis curve of TWSMEs and superelasticity. These models, cf. Eq. (33), include the quadratic model proposed by Lagoudas [14], the cosine model proposed by

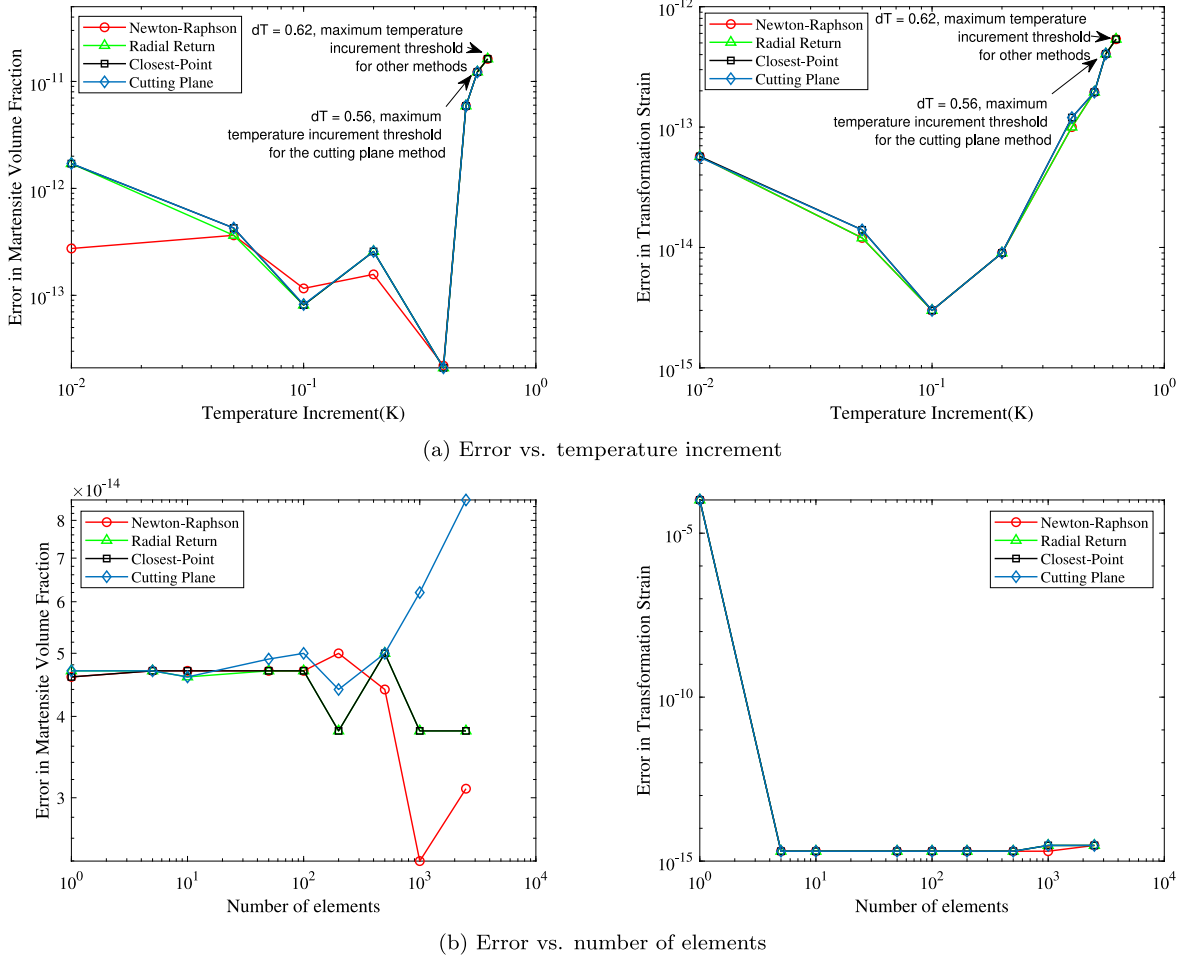


Fig. 3. Error between the 1D analytic solution and different return mapping schemes for updating internal state variables at $T = 220$ K.

Liang [10,11], the exponential model proposed by Tanaka [9], and the smooth model proposed by Lagoudas [14].

$$\begin{aligned}
 \text{Quadratic : } f(\xi) &= \begin{cases} \frac{1}{2}\rho b^M \xi^2 + (\mu_1 + \mu_2)\xi & (\dot{\xi} > 0) \\ \frac{1}{2}\rho b^A \xi^2 + (\mu_1 - \mu_2)\xi & (\dot{\xi} < 0) \end{cases} \\
 \text{Cosine : } f(\xi) &= \begin{cases} \int_{\xi}^0 -\frac{\rho \Delta s_0}{a_c^M} [\pi - \cos^{-1}(2\xi - 1)] d\xi + (\mu_1^c + \mu_2^c)\xi & (\dot{\xi} > 0) \\ \int_{\xi}^0 -\frac{\rho \Delta s_0}{a_c^A} [\pi - \cos^{-1}(2\xi - 1)] d\xi + (\mu_1^c - \mu_2^c)\xi & (\dot{\xi} < 0) \end{cases} \\
 \text{Exponential : } f(\xi) &= \begin{cases} \frac{\rho \Delta s_0}{a_c^M} [(1 - \xi) \ln(1 - \xi) + \xi] + (\mu_1^e + \mu_2^e)\xi & (\dot{\xi} > 0) \\ -\frac{\rho \Delta s_0}{a_c^A} [\xi \ln(\xi) - \xi] + (\mu_1^e - \mu_2^e)\xi & (\dot{\xi} < 0) \end{cases} \\
 \text{Smooth : } f(\xi) &= \begin{cases} \frac{1}{2}\rho b^M \left[\xi + \frac{\xi^{n_1+1}}{n_1+1} + \frac{(1-\xi)^{n_2+1}}{n_2+1} \right] & (\dot{\xi} > 0) \\ \frac{1}{2}\rho b^A \left[\xi + \frac{\xi^{n_3+1}}{n_3+1} + \frac{(1-\xi)^{n_4+1}}{n_4+1} \right] & (\dot{\xi} < 0) \end{cases}
 \end{aligned} \tag{33}$$

the parameters a, b, n and μ in Eq. (33), are defined in [14]. In Fig. 5, we present the evolution of strain and martensite volume fraction under a temperature cycle, and the evolution of stress and martensite volume fraction under a loading cycle, at the left-most Gauss point of the left-most element.

These results agree with the return mapping computations and the analytical solutions presented in [14].

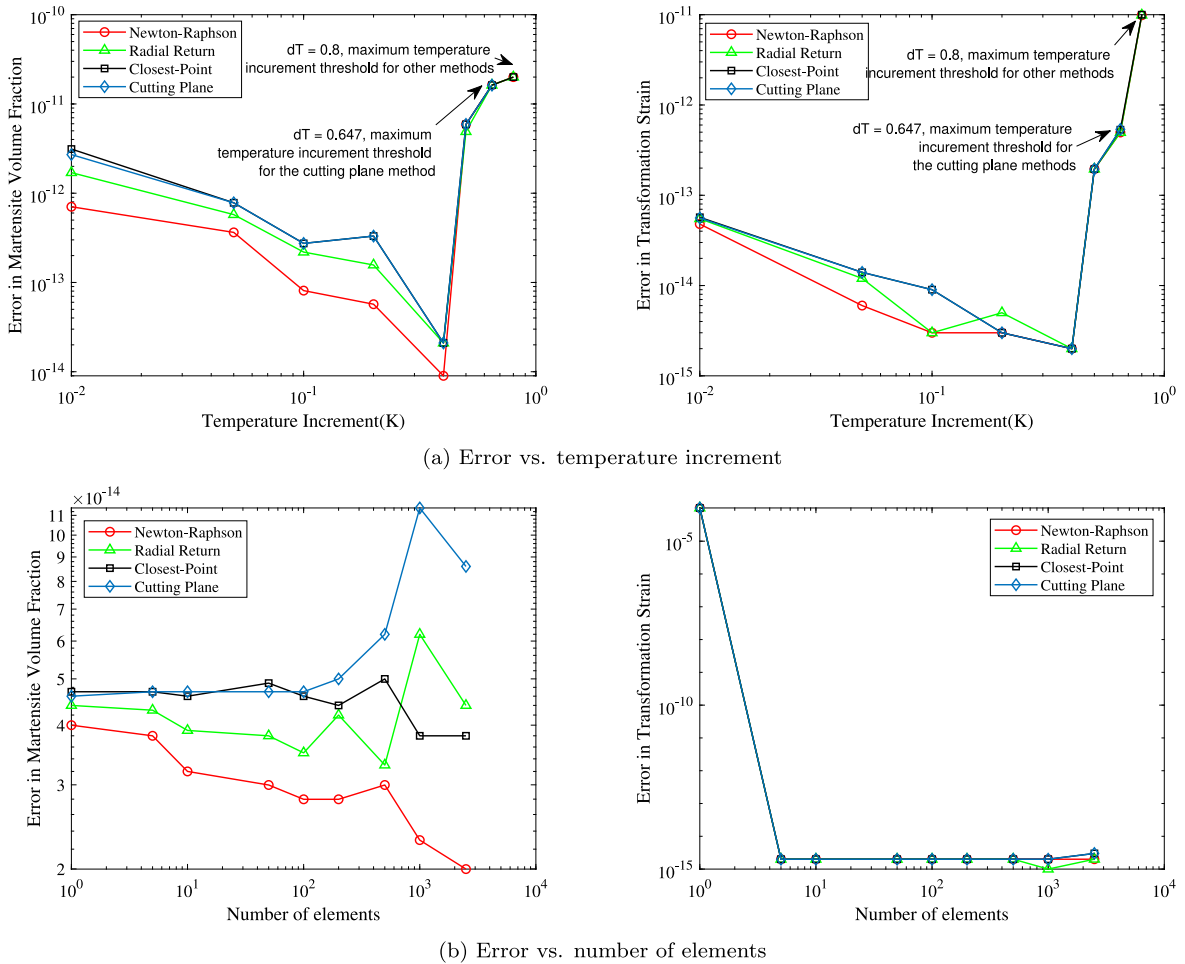


Fig. 4. Error between the 1D analytic solution and different parallel projection schemes for updating internal state variables at $T = 220$ K.

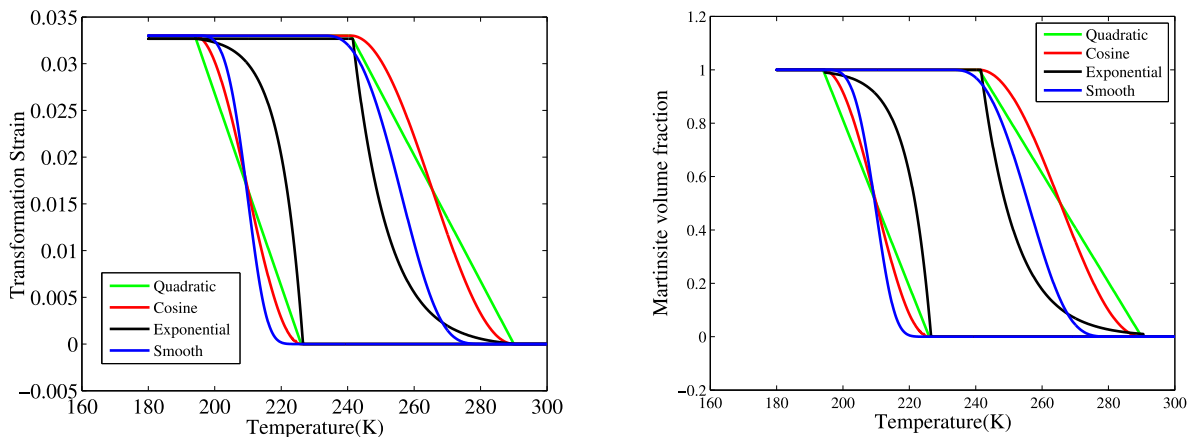
4.2. 3D SMA examples

4.2.1. Computation efficiency of the parallel projection algorithm in 3-Dimensions

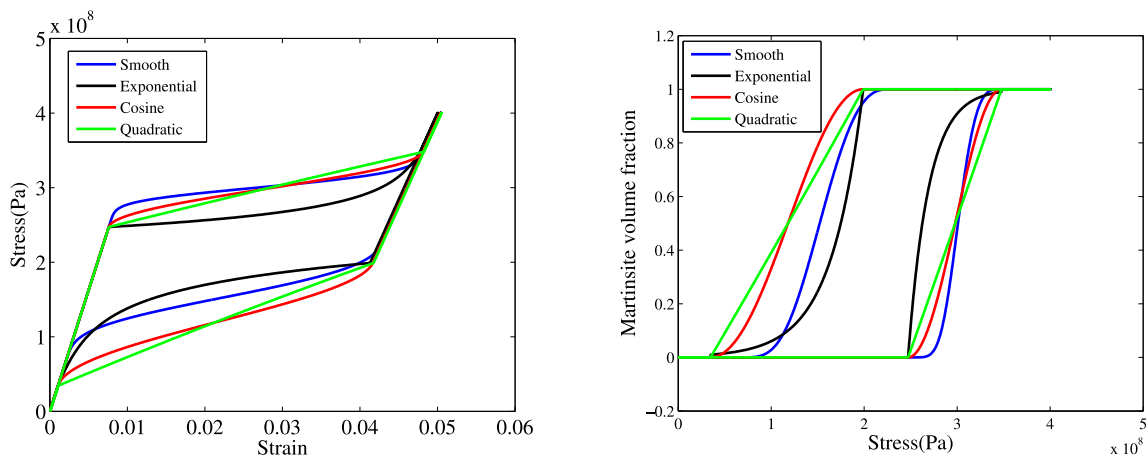
In this section, we compare the parallel projection algorithm with the return mapping algorithm in a 3D simulation, to exemplify the computation efficiency of the parallel projection algorithm. The 3D SMA bar that is meshed with $2 \times 10 \times 2$ hexahedral elements, is subject to a coupled temperature and mechanical loading cycle, with amplitude ranges 210–310 K and 2×10^6 – 2.2×10^7 N/m², cf. Fig. 6

In Fig. 7, we see that the bar deforms in all three directions due to the Poisson effects

Fig. 8 compares the evolution of the axial strain $\varepsilon_{xx}^{\mathcal{G}_A}$, stress $\sigma_{xx}^{\mathcal{G}_A}$ and martensite volume fraction $\xi_{\mathcal{G}_A}$ for the two methods. Here, point A is the right tip point of the structure, shown in Fig. 6. \mathcal{G}_A is the closest Gauss point to the point A. Note that the phase transformation is influenced by both the temperature and load. From the results, we see an agreement between the parallel projection and return mapping computations. Fig. 9 shows the convergence rate of the global residual R at thermomechanical loading step size of $dT = 0.1$ K and $dF = 1 \times 10^5$ N/m². We conjecture that the radius of quadratic convergence of the return mapping algorithm is adversely influenced due to enforcing H to zero at each outer iteration. We also noticed that all four methods exhibit quadratic convergence for the parallel projection algorithm, although their radii of convergence vary. Indeed, divergence was occasionally observed. We suggest using a line-search method to mitigate divergence, as proposed for the cutting plane method described in [31,46].



(a) Two-way shape memory effect



(b) Superelasticity

Fig. 5. Simulating TWSMEs and superelasticity with the parallel projection algorithm using different hardening models.

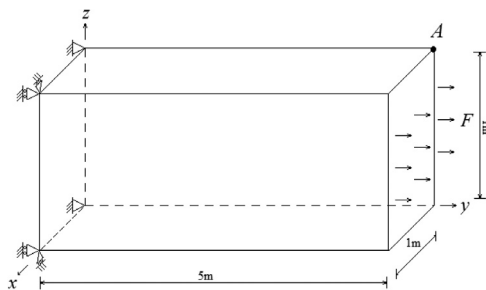


Fig. 6. Geometry and boundary conditions of the 3D SMA bar.

Tables 5 and 6 present the maximum thermomechanical loading increment allowed for the return mapping and parallel projection algorithms for this 3D problem. We observe that the parallel projection algorithm again exhibits larger increments than the return mapping algorithm, which is consistent with our findings in the 1D problem.

Table 7 shows the number of inner and outer iterations needed for a single-element at one time step, using the closest-point scheme with thermomechanical loading step size of $dT = 0.1$ K and $dF = 1 \times 10^5$ N/m². Here, k is the global iteration counter and l is the local counter. Since the parallel projection uses a cross-iteration scheme,

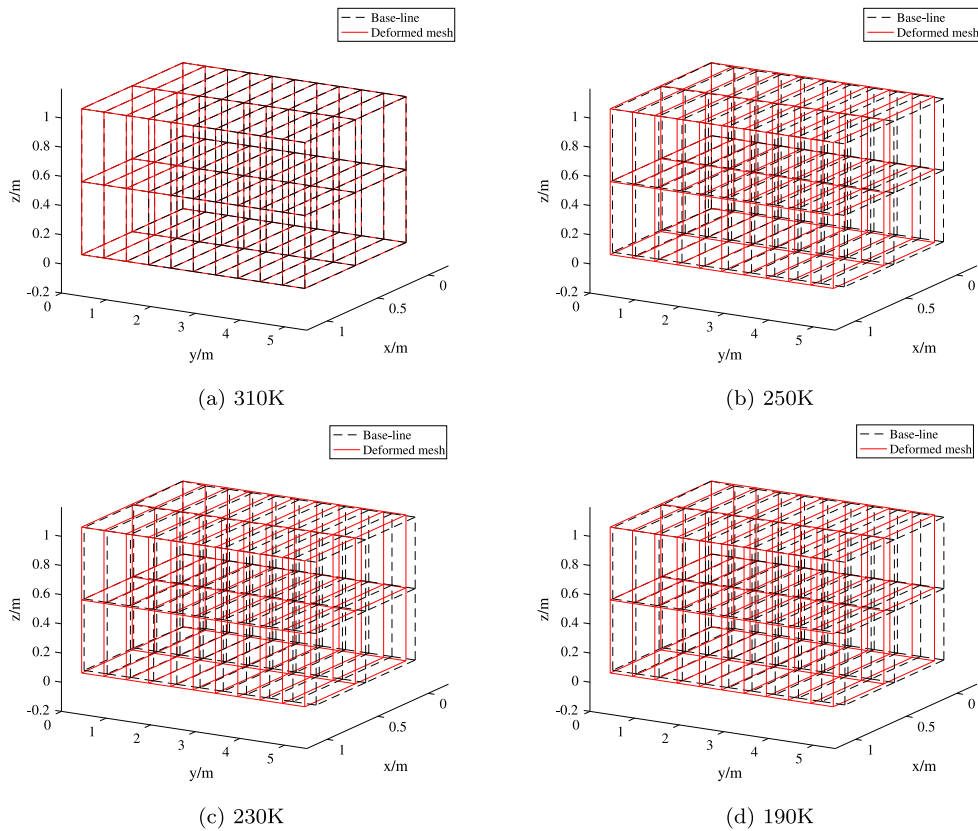
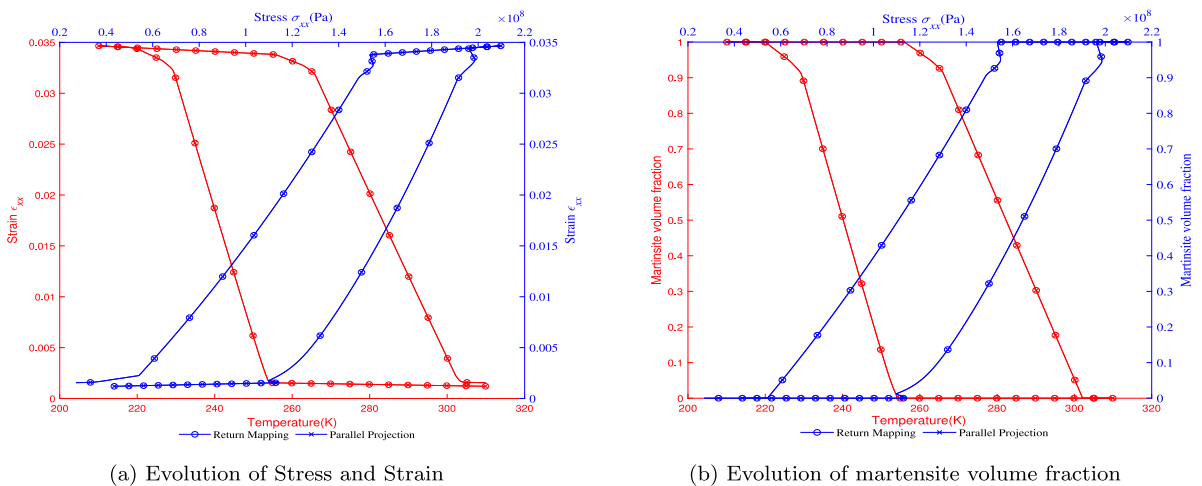


Fig. 7. Deformation of a 3D bar under coupled thermomechanical loading.



(a) Evolution of Stress and Strain (b) Evolution of martensite volume fraction

Fig. 8. Simultaneous presence of TWSMEs and superelasticity in 3D bar under a temperature and loading cycle.

there is only a global counter k . As seen in the table, the parallel projection scheme requires 6 iterations, as opposed to the 11 *outer* iterations for the return mapping algorithm. In total the return mapping algorithm requires 11 *outer* iterations and 4 inner iterations for *each* of the 8 Gauss point for a total of $N = 11 \cdot 4 \cdot 8 = 352$ local computations versus $N = 6 \cdot 8 = 48$ for the parallel projection method. Ultimately, the computed state variables for the two methods are identical (within the specified tolerances) as expected, since the same equations are solved.

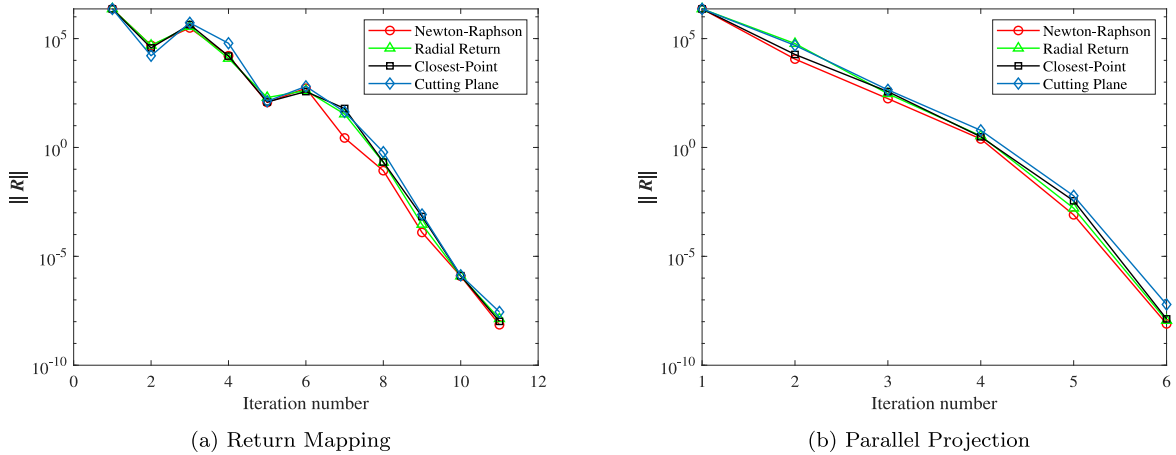


Fig. 9. Convergence history of different schemes for updating internal state variables at $T = 270$ K.

Table 5

Approximate maximum temperature step, with fixed $dF = 1 \times 10^5$ N/m².

Methods	Newton–Raphson	Radial Return	Closest-Point	Cutting Plane
Return Mapping	1.04 K	1.04 K	1.04 K	0.98 K
Parallel Projection	4.7 K	4.7 K	4.7 K	4.7 K

Table 6

Approximate maximum loading step, with fixed $dT = 0.1$ K.

Methods	Newton–Raphson	Radial Return	Closest-Point	Cutting Plane
Return Mapping	1×10^7 N	1×10^7 N	1×10^7 N	1×10^7 N
Parallel Projection	5×10^7 N	5×10^7 N	5×10^7 N	5×10^7 N

Table 7

Gauss point computation iteration count using the closet-point scheme, $T = 230$ K.

Methods	k	//Gauss Point	N	d_y^A	ξ_{G_A}	$\epsilon_{yy}^{t_{G_A}}$
Return Mapping	11	4	352	0.178241294	0.9711878	0.0320491973
Parallel Projection	6	/	48	0.178241294	0.9711880	0.0320491924

Now, we investigate the effect of the mesh size on the computation time for the two algorithms, cf. Fig. 10. In all cases, the parallel projection method yields computation savings of greater than 50% without losing accuracy.

The computational savings of the parallel projection algorithm is due to the need for fewer global equilibrium iterations, and the fact that we do not have to satisfy the local evolution equations. Despite these promising results, it cannot be stated conclusively that the parallel projection approach will be more computationally efficient than the traditional return mapping method for any mesh, SMA constitutive law or loading condition.

5. Conclusions

We present a novel framework to improve upon the computational efficiency of the classical return mapping algorithm in finite element analysis of SMAs. Instead of using a nested iteration, the proposed approach—the parallel projection algorithm—uses a sequential iterative scheme, which is able to significantly reduce computation time while not increasing the size of tangent matrix within the Newton–Raphson iterations. In addition, we provide the analytical update for the internal state variables. To test the robustness of the parallel projection algorithm, various numerical analysis examples have been studied. The examples show that parallel projection is able to achieve stable convergence, and obtain results that are consistent with those simulated using established methods.

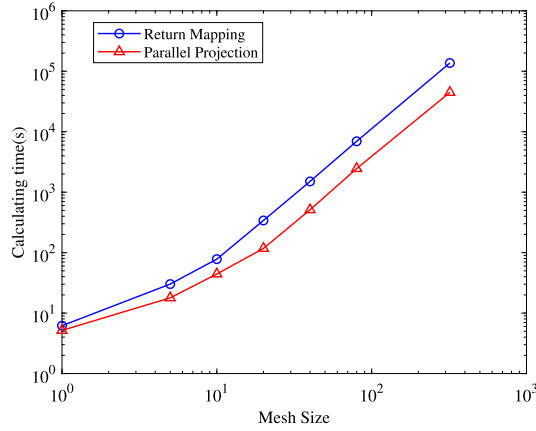


Fig. 10. Comparison between the parallel projection and return mapping algorithms.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research was supported by the National Science Foundation, USA through Grant Number CMMI1663566.

Appendix. Analytical derivation of Newton–Raphson updating scheme

In this section, we present the derivation of the analytical solution of the increment $-\left(\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n}\right)^{-1} \mathbf{H}_n$ for updating the internal state variable vector \mathbf{v} during the Newton–Raphson procedure. Due to the complexity of deriving the analytical solution of the aforementioned operator, previous research generally simplifies the form of local residual \mathbf{H} to get approximate derivations, i.e. the closest-point and cutting plane methods [14]. This may lead to stability issues when using the results to update the global state variable \mathbf{u} , since the tangent stiffness modulus \mathfrak{L} gradually becomes more inaccurate as the residual is simplified. Another option is to numerically calculate the inverse of $\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n}$ using QR or LU decomposition. However, due to the large magnitude of the compliance modulus S in an off-diagonal position, the matrix $\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n}$ is ill-conditioned.

Here, we use the Schur algorithm to provide the analytical solution of the inverse of $\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n}$. For simplicity, we only present the derivation for the forward transformation, i.e. $\dot{\xi} > 0$. The derivation for the reverse transformation, i.e. ($\dot{\xi} < 0$), can be obtained similarly following the derivation presented here.

To start, we first represent the inverse of $\partial \mathbf{H}_n / \partial \mathbf{v}_n$ using the Schur formulation [47].

$$\left(\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n}\right)^{-1} = \begin{bmatrix} (\mathbf{A} + \mathbf{B}\mathbf{C})^{-1} & (\mathbf{A} + \mathbf{B}\mathbf{C})^{-1}\mathbf{B} \\ \mathbf{C}(\mathbf{A} + \mathbf{B}\mathbf{C})^{-1} & -\mathbf{I} + \mathbf{C}(\mathbf{A} + \mathbf{B}\mathbf{C})^{-1}\mathbf{B} \end{bmatrix} \tag{A.1}$$

with

$$\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n} = \left[\begin{array}{ccc|ccc} \partial_{\xi} \Phi_n & \mathbf{0}_{1 \times 6} & \mathbf{0}_{1 \times 1} & \partial_{\sigma_n} \Phi_n^T & & \\ \mathbf{A}_n & -\mathbf{I}_{6 \times 6} & \mathbf{0}_{6 \times 1} & \partial_{\sigma} \mathbf{A}_n : \Delta \xi_n & & \\ \Delta S & \mathbf{0}_{1 \times 6} & -\mathbf{I}_{1 \times 1} & \mathbf{0}_{1 \times 6} & & \\ \hline \mathbf{0}_{6 \times 1} & -\mathbf{S}_n^{-1} & -\mathbf{S}_n^{-1} : \mathbf{I}_{6 \times 6} : \sigma_n & -\mathbf{I}_{6 \times 6} & & \end{array} \right] = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \tag{A.2}$$

where $\Delta \xi_n = \xi_n - \xi_{n-1}$ and $\partial_{\sigma} \mathbf{A}$ have nonzero value in the forward transformation, cf. Eq. (11)

Meanwhile, we define the local residual \mathbf{H} in block matrix form as

$$\mathbf{H}_n = \begin{Bmatrix} H_\phi \\ \mathbf{H}_{\mathbf{e}^t} \\ H_S \\ \mathbf{H}_\sigma \end{Bmatrix} = \begin{Bmatrix} \tilde{\mathbf{H}} \\ \mathbf{H}_\sigma \end{Bmatrix} \tag{A.3}$$

Then the increment of the internal state variable for Newton–Raphson iterations is expressed as

$$\Delta \mathbf{v} = - \left(\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n} \right)^{-1} \mathbf{H}_n = \begin{Bmatrix} -(\mathbb{A} + \mathbb{B}\mathbb{C})^{-1} \tilde{\mathbf{H}} - (\mathbb{A} + \mathbb{B}\mathbb{C})^{-1} \mathbb{B} \mathbf{H}_\sigma \\ -\mathbb{C}(\mathbb{A} + \mathbb{B}\mathbb{C})^{-1} \tilde{\mathbf{H}} - [-\mathbf{I} + \mathbb{C}(\mathbb{A} + \mathbb{B}\mathbb{C})^{-1} \mathbb{B}] \mathbf{H}_\sigma \end{Bmatrix} \tag{A.4}$$

Note that the solution shown in Eq. (A.4) does not change the ill-conditioned characteristics of the matrix $(\mathbb{A} + \mathbb{B}\mathbb{C})$, since \mathbb{A}^{-1} is nearly singular due to the fact that the large number S is still in an off-diagonal position. Hence, the key to obtaining an accurate solution of the operators lies in computing the inverse of $(\mathbb{A} + \mathbb{B}\mathbb{C})$. Again, we use the Schur formulation to calculate the inverse of the matrix.

$$(\mathbb{A} + \mathbb{B}\mathbb{C})^{-1} = \begin{bmatrix} (\mathbb{A}' - \mathbb{B}'\mathbb{D}'^{-1}\mathbb{C}')^{-1} & -(\mathbb{A}' - \mathbb{B}'\mathbb{D}'^{-1}\mathbb{C}')^{-1}\mathbb{B}'\mathbb{D}'^{-1} \\ -\mathbb{D}'^{-1}\mathbb{C}'(\mathbb{A}' - \mathbb{B}'\mathbb{D}'^{-1}\mathbb{C}')^{-1} & \mathbb{D}'^{-1} + \mathbb{D}'^{-1}\mathbb{C}'(\mathbb{A}' - \mathbb{B}'\mathbb{D}'^{-1}\mathbb{C}')^{-1}\mathbb{B}'\mathbb{D}'^{-1} \end{bmatrix} \tag{A.5}$$

with

$$\mathbb{A} + \mathbb{B}\mathbb{C} = \left[\begin{array}{c|cc} \frac{\partial_\xi \Phi_n}{\mathbf{A}_n} & -\partial_\sigma \Phi_n : \mathbf{S}_n^{-1} & -\partial_\sigma \Phi_n : \mathbf{S}_n^{-1} : \mathbf{I}_{6 \times 6} : \boldsymbol{\sigma}_n \\ \hline \Delta S & -(\mathbf{S}_n + \partial_\sigma \mathbf{A}_n : \Delta \xi_n) : \mathbf{S}_n^{-1} & -\partial_\sigma \mathbf{A}_n : \Delta \xi_n : \mathbf{S}_n^{-1} : \mathbf{I}_{6 \times 6} : \boldsymbol{\sigma}_n \\ & \mathbf{0}_{1 \times 6} & -\mathbf{I}_{1 \times 1} \end{array} \right] \tag{A.6}$$

$$= \begin{bmatrix} \mathbb{A}' & \mathbb{B}' \\ \mathbb{C}' & \mathbb{D}' \end{bmatrix}$$

Here, \mathbb{D}'^{-1} and $(\mathbb{A}' - \mathbb{B}'\mathbb{D}'^{-1}\mathbb{C}')^{-1}$ need to be evaluated. Note that $\partial_\sigma \mathbf{A}$ belongs to the null space of $\boldsymbol{\sigma}$, hence is not invertible [48]. However, $(\mathbf{S}_n + \partial_\sigma \mathbf{A}_n : \Delta \xi_n)$ is non-singular. Defining $\boldsymbol{\zeta}_n = \mathbf{S}_n + \partial_\sigma \mathbf{A}_n : \Delta \xi_n$, \mathbb{D}'^{-1} is expressed using the Schur form as

$$\mathbb{D}'^{-1} = \begin{bmatrix} -\mathbf{S}_n : \boldsymbol{\zeta}_n^{-1} & \mathbf{S}_n : \boldsymbol{\zeta}_n^{-1} : \partial_\sigma \mathbf{A}_n : \Delta \xi_n : \mathbf{S}_n^{-1} : \mathbf{I}_{6 \times 6} : \boldsymbol{\sigma}_n \\ \mathbf{0}_{1 \times 6} & -\mathbf{I}_{1 \times 1} \end{bmatrix} \tag{A.7}$$

With the above solution, one also obtains that

$$\mathbb{B}'\mathbb{D}'^{-1} = \begin{bmatrix} \partial_\sigma \Phi_n : \boldsymbol{\zeta}_n^{-1} & \partial_\sigma \Phi_n : \boldsymbol{\zeta}_n^{-1} : \mathbf{S}_n^{-1} : \mathbf{S}_n^{-1} : \mathbf{I}_{6 \times 6} : \boldsymbol{\sigma}_n \end{bmatrix} \tag{A.8}$$

$$\mathbb{D}'^{-1}\mathbb{C}' = \begin{bmatrix} -\mathbf{S}_n : \boldsymbol{\zeta}_n^{-1} : (\mathbf{A}_n - \partial_\sigma \mathbf{A}_n : \Delta \xi_n : \mathbf{S}_n^{-1} : \mathbf{I}_{6 \times 6} : \boldsymbol{\sigma}_n : \Delta S) \\ -\Delta S \end{bmatrix} = \begin{bmatrix} -\tilde{\mathbf{A}}_n \\ -\Delta S \end{bmatrix}$$

Then $(\mathbb{A}' - \mathbb{B}'\mathbb{D}'^{-1}\mathbb{C}')^{-1}$ is calculated as

$$(\mathbb{A}' - \mathbb{B}'\mathbb{D}'^{-1}\mathbb{C}')^{-1} = -\mathbf{Q}, \quad \mathbf{Q} = \partial_\sigma \Phi_n : \boldsymbol{\zeta}_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi_n \tag{A.9}$$

and \mathbf{Q} is a scalar. Hence

$$\mathbb{C}(\mathbb{A} + \mathbb{B}\mathbb{C})^{-1} = \begin{bmatrix} \frac{\boldsymbol{\zeta}_n^{-1} : \partial_\sigma \Phi_n}{\mathbf{Q}} & \boldsymbol{\mathfrak{L}}_n & \boldsymbol{\mathfrak{L}}_n : \mathbf{S}_n : \mathbf{S}_n^{-1} : \boldsymbol{\sigma}_n \end{bmatrix} \tag{A.10}$$

$$\mathbb{C}(\mathbb{A} + \mathbb{B}\mathbb{C})^{-1}\mathbb{B} = \frac{\boldsymbol{\zeta}_n^{-1} : \partial_\sigma \Phi_n : \partial_\sigma \Phi_n^T}{\mathbf{Q}} + \boldsymbol{\mathfrak{L}}_n : \partial_\sigma \mathbf{A}_n : \Delta \xi_n$$

Ultimately, we obtain the analytical expressions of the internal state variable increment for the forward transformation as

$$\Delta \mathbf{v} = \begin{Bmatrix} \delta \xi \\ \delta \mathbf{e}^t \\ \delta S \\ \delta \boldsymbol{\sigma} \end{Bmatrix} = \begin{Bmatrix} \Delta \xi^* + \vartheta \\ \Delta \mathbf{e}^{t*} + \tilde{\mathbf{A}}_n : \vartheta + \tilde{\boldsymbol{\Psi}} \\ \Delta S^* + H_S + \Delta S \cdot \vartheta \\ \Delta \boldsymbol{\sigma}^* + \boldsymbol{\mathfrak{L}}_n : \mathbf{S}_n : \boldsymbol{\Psi} \end{Bmatrix} \tag{A.11}$$

where

$$\begin{aligned}
 \Delta \xi^* &= \frac{\Phi_n - \partial_\sigma \Phi_n : \zeta_n^{-1} : H_{\epsilon t}}{\partial_\sigma \Phi_n : \zeta_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi_n} & (\dot{\xi} > 0) \\
 \Delta \sigma^* &= \zeta_n^{-1} : [-H_{\epsilon t} - \Delta \xi^* : \partial_\sigma \Phi_n] & (\dot{\xi} > 0) \\
 \Delta \mathbf{e}^{f*} &= -\mathbf{S}_n : \Delta \sigma^* - (\Delta S : \mathbf{I} : \sigma_n) \Delta \xi^* \\
 \Delta S^* &= \Delta S \Delta \xi^* \\
 \tilde{\Lambda}_n &= \mathbf{S}_n : \zeta_n^{-1} : (\Lambda_n - \partial_\sigma \Lambda_n : \Delta \xi_n : S_n^{-1} : \mathbf{I}_{6 \times 6} : \sigma_n : \Delta S) \\
 \Psi &= \mathbf{H}_\sigma - S_n^{-1} : \mathbf{I} : \sigma_n : H_S \\
 \tilde{\Psi} &= \mathbf{S}_n : \zeta_n^{-1} : \partial_\sigma \Lambda_n : \Delta \xi_n : \Psi \\
 \vartheta &= \frac{\partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : \zeta_n^{-1} : \partial_\sigma \Phi_n^{(k)} - \partial_\xi \Phi_n} : \zeta_n^{-1} : \mathbf{S}_n : \Psi & (\dot{\xi} > 0)
 \end{aligned} \tag{A.12}$$

For the inverse transformation, we have $\zeta_n = S_n$, $\tilde{\Lambda}_n = \Lambda_n$ and $\tilde{\Psi} = 0$, since $\partial_\sigma \Lambda_n = 0$. According to the definition of Φ in Eq. (14), we shall only modify the formulations for $\Delta \xi^*$, $\Delta \sigma^*$ and ϑ

$$\begin{aligned}
 \Delta \xi^* &= \frac{\Phi_n - \partial_\sigma \Phi_n : S_n^{-1} : H_{\epsilon t}}{-\partial_\sigma \Phi_n : S_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi_n} & (\dot{\xi} < 0) \\
 \Delta \sigma^* &= S_n^{-1} : [-H_{\epsilon t} + \Delta \xi^* : \partial_\sigma \Phi_n] & (\dot{\xi} < 0) \\
 \vartheta &= \frac{\partial_\sigma \Phi_n}{-\partial_\sigma \Phi_n : S_n^{-1} : \partial_\sigma \Phi_n^{(k)} - \partial_\xi \Phi_n} : \Psi & (\dot{\xi} < 0)
 \end{aligned} \tag{A.13}$$

Then, we obtain the consistent tangent operator $(\partial \mathbf{R} / \partial \mathbf{v})(d\mathbf{v} / d\mathbf{u})$ using the aforementioned expression of $(\partial \mathbf{H} / \partial \mathbf{v})^{-1}$. Given that

$$\begin{aligned}
 \frac{\partial \mathbf{R}_{el,n}}{\partial \mathbf{v}_n} &= [0 \quad 0 \quad 0 \quad w \mathbf{B}^T \det \mathbf{J}] \\
 \frac{\partial \mathbf{H}_{G,n}}{\partial \mathbf{u}_{el,n}} &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ S_n^{-1} : \mathbf{B} \end{bmatrix}
 \end{aligned} \tag{A.14}$$

and $d\mathbf{v} / d\mathbf{u} = -(\partial \mathbf{H} / \partial \mathbf{v})^{-1} (\partial \mathbf{H} / \partial \mathbf{u})$ from Eq. (22), the consistent tangent operator is derived as

$$\begin{aligned}
 \frac{\partial \mathbf{R}_n}{\partial \mathbf{v}_n} \frac{d\mathbf{v}}{d\mathbf{u}} &= - \bigwedge_{el} \sum_{\mathfrak{G}} \frac{\partial \mathbf{R}_n}{\partial \mathbf{v}_{\mathfrak{G},n}} \left(\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_{\mathfrak{G},n}} \right)^{-1} \frac{\partial \mathbf{H}_{\mathfrak{G},n}}{\partial \mathbf{u}_{el,n}} \\
 &= \bigwedge_{el} \sum_{\mathfrak{G}} w \mathbf{B}_{\mathfrak{G}}^T [\mathbf{I} - \mathbb{C}(\mathbb{A} + \mathbb{B}\mathbb{C})^{-1} \mathbb{B}]_{\mathfrak{G}} : S_{\mathfrak{G},n}^{-1} \mathbf{B}_{\mathfrak{G}} \det \mathbf{J}_{\mathfrak{G}} \\
 &= \bigwedge_{el} \sum_{\mathfrak{G}} w \mathbf{B}_{\mathfrak{G}}^T \mathfrak{L}_n \mathbf{B}_{\mathfrak{G}} \det \mathbf{J}_{\mathfrak{G}}
 \end{aligned} \tag{A.15}$$

where

$$\mathfrak{L}_n = \begin{cases} \zeta_n^{-1} - \frac{\zeta_n^{-1} : \partial_\sigma \Phi_n \otimes \zeta_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : \zeta_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi_n} & \dot{\xi} > 0 \\ S_n^{-1} - \frac{S_n^{-1} : \partial_\sigma \Phi_n \otimes S_n^{-1} : \partial_\sigma \Phi_n}{\partial_\sigma \Phi_n : S_n^{-1} : \partial_\sigma \Phi_n + \partial_\xi \Phi_n} & \dot{\xi} < 0 \end{cases} \tag{A.16}$$

In addition, we obtain that $(\partial \mathbf{H} / \partial \mathbf{v})^{-1} (\partial \mathbf{H} / \partial \mathbf{u})$

$$\left(\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n} \right)^{-1} \frac{\partial \mathbf{H}_n}{\partial \mathbf{u}_n} = \begin{cases} \left[\begin{array}{c} -\frac{\partial_\sigma \Phi_n^T : \xi_n^{-1}}{\partial_\sigma \Phi_n : \xi_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi} \\ -\frac{\tilde{\Lambda}_n : \partial_\sigma \Phi_n^T}{\partial_\sigma \Phi_n : \xi_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi} : \xi_n^{-1} + \mathbf{S}_n : \xi_n^{-1} - \mathbf{I}_{6 \times 6} \\ -\frac{\Delta S : \partial_\sigma \Phi_n^T}{\partial_\sigma \Phi_n : \xi_n^{-1} : \partial_\sigma \Phi_n - \partial_\xi \Phi} : \xi_n^{-1} \\ -\mathfrak{L}_n \end{array} \right] \mathbf{B} \quad \dot{\xi} > 0 \\ \left[\begin{array}{c} -\frac{\partial_\sigma \Phi_n^T : S_n^{-1}}{\partial_\sigma \Phi_n : S_n^{-1} : \partial_\sigma \Phi_n + \partial_\xi \Phi_n} \\ \left[\begin{array}{c} \mathbf{A}_n \\ \Delta S \end{array} \right] : \partial_\sigma \Phi_n^T : S_n^{-1} \\ -\mathfrak{L}_n \end{array} \right] \mathbf{B} \quad \dot{\xi} < 0 \end{cases} \quad (\text{A.17})$$

where

$$\tilde{\Lambda}_n = \mathbf{S}_n : \xi_n^{-1} : (\mathbf{A}_n - \partial_\sigma \mathbf{A}_n : \Delta \xi_n : S_n^{-1} : \mathbf{I}_{6 \times 6} : \sigma_n : \Delta S) \quad (\text{A.18})$$

Further, $(\partial \mathbf{R} / \partial \mathbf{v})(\partial \mathbf{H} / \partial \mathbf{v})^{-1}$ is obtained as

$$\frac{\partial \mathbf{R}_n}{\partial \mathbf{v}_n} \left(\frac{\partial \mathbf{H}_n}{\partial \mathbf{v}_n} \right)^{-1} = \begin{cases} w \mathbf{B}^T \left[\begin{array}{ccc} \frac{\xi_n^{-1} : \partial_\sigma \Phi_n}{Q} & \mathfrak{L}_n & \mathfrak{L}_n : \mathfrak{C}^{-1} : \sigma_n & -\mathfrak{L}_n : S_n \end{array} \right] \det \mathbf{J} \quad \dot{\xi} > 0 \\ w \mathbf{B}^T \left[\begin{array}{ccc} \frac{S_n^{-1} : \partial_\sigma \Phi_n}{Q} & \mathfrak{L}_n & \mathfrak{L}_n : \mathfrak{C}^{-1} : \sigma_n & -\mathfrak{L}_n : S_n \end{array} \right] \det \mathbf{J} \quad \dot{\xi} < 0 \end{cases} \quad (\text{A.19})$$

References

- [1] T.W. Duerig, K. Melton, D. Stöckel, Engineering Aspects of Shape Memory Alloys, Butterworth-Heinemann, 2013.
- [2] K. Otsuka, C.M. Wayman, Shape Memory Materials, Cambridge University Press, 1999.
- [3] Q.P. Sun, K.C. Hwang, Micromechanics modelling for the constitutive behavior of polycrystalline shape memory alloys—I. Derivation of general relations, J. Mech. Phys. Solids 41 (1) (1993) 1–17.
- [4] Q.P. Sun, K.C. Hwang, Micromechanics modelling for the constitutive behavior of polycrystalline shape memory alloys—II. Study of the individual phenomena, J. Mech. Phys. Solids 41 (1) (1993) 19–33.
- [5] K. Bhattacharya, R.V. Kohn, Symmetry, texture and the recoverable strain of shape-memory polycrystals, Acta Mater. 44 (2) (1996) 529–542.
- [6] K. Bhattacharya, et al., Microstructure of Martensite: Why It Forms and how It Gives Rise to the Shape-Memory Effect, Vol. 2, Oxford University Press, 2003.
- [7] C. Cisse, W. Zaki, T.B. Zineb, A review of constitutive models and modeling techniques for shape memory alloys, Int. J. Plast. 76 (2016) 244–284.
- [8] M. Frost, B. Benešová, P. Sedláč, A microscopically motivated constitutive model for shape memory alloys: formulation, analysis and computations, Math. Mech. Solids 21 (3) (2016) 358–382.
- [9] K. Tanaka, F. Nishimura, T. Hayashi, H. Tobushi, C. LExcellent, Phenomenological analysis on subloops and cyclic behavior in shape memory alloys under mechanical and/or thermal loads, Mech. Mater. 19 (4) (1995) 281–292.
- [10] C. Liang, C. Rogers, A multi-dimensional constitutive model for shape memory alloys, J. Eng. Math. 26 (3) (1992) 429–443.
- [11] C. Liang, C.A. Rogers, One-dimensional thermomechanical constitutive relations for shape memory materials, J. Intell. Mater. Syst. Struct. 8 (4) (1997) 285–302.
- [12] J.G. Boyd, D.C. Lagoudas, A thermodynamical constitutive model for shape memory materials. Part I. The monolithic shape memory alloy, Int. J. Plast. 12 (6) (1996) 805–842.
- [13] J.G. Boyd, D.C. Lagoudas, A thermodynamical constitutive model for shape memory materials. Part II. The SMA composite material, Int. J. Plast. 12 (7) (1996) 843–873.
- [14] D.C. Lagoudas, Shape Memory Alloys: Modeling and Engineering Applications, Springer, 2008.
- [15] D. Lagoudas, D. Hartl, Y. Chemisky, L. Machado, P. Popov, Constitutive model for the numerical analysis of phase transformation in polycrystalline shape memory alloys, Int. J. Plast. 32 (2012) 155–183.
- [16] C. Yu, G. Kang, Q. Kan, D. Song, A micromechanical constitutive model based on crystal plasticity for thermo-mechanical cyclic deformation of NiTi shape memory alloys, Int. J. Plast. 44 (2013) 161–191.
- [17] C. Yu, G. Kang, Q. Kan, Crystal plasticity based constitutive model of NiTi shape memory alloy considering different mechanisms of inelastic deformation, Int. J. Plast. 54 (2014) 132–162.
- [18] C. LExcellent, J. Rejzner, Modeling of the strain rate effect, creep and relaxation of a Ni-ti shape memory alloy under tension (compression)-torsional proportional loading in the pseudoelastic range, Smart Mater. Struct. 9 (5) (2000) 613.
- [19] D.C. Lagoudas, G. Chatzigeorgiou, P.K. Kumar, Modeling and experimental study of simultaneous creep and transformation in polycrystalline high-temperature shape memory alloys, J. Intell. Mater. Syst. Struct. 20 (18) (2009) 2257–2267.

- [20] D.J. Hartl, G. Chatzigeorgiou, D.C. Lagoudas, Three-dimensional modeling and numerical analysis of rate-dependent irrecoverable deformation in shape memory alloys, *Int. J. Plast.* 26 (10) (2010) 1485–1507.
- [21] A. Paiva, M.A. Savi, A.M.B. Braga, P.M.C.L. Pacheco, A constitutive model for shape memory alloys considering tensile–compressive asymmetry and plasticity, *Int. J. Solids Struct.* 42 (11–12) (2005) 3439–3457.
- [22] R. Mehrabi, M. Kadkhodaei, M. Elahinia, Constitutive modeling of tension-torsion coupling and tension-compression asymmetry in NiTi shape memory alloys, *Smart Mater. Struct.* 23 (7) (2014) 075021.
- [23] L. Xu, T. Baxevis, D. Lagoudas, A three-dimensional constitutive model for the martensitic transformation in polycrystalline shape memory alloys under large deformation, *Smart Mater. Struct.* 28 (7) (2019) 074004.
- [24] L.C. Brinson, One-dimensional constitutive behavior of shape memory alloys: thermomechanical derivation with non-constant material functions and redefined martensite internal variable, *J. Intell. Mater. Syst. Struct.* 4 (2) (1993) 229–242.
- [25] S. Leclercq, C. LExcellent, A general macroscopic description of the thermomechanical behavior of shape memory alloys, *J. Mech. Phys. Solids* 44 (6) (1996) 953–980.
- [26] S. Reese, D. Christ, Finite deformation pseudo-elasticity of shape memory alloys—constitutive modelling and finite element implementation, *Int. J. Plast.* 24 (3) (2008) 455–482.
- [27] J. Arghavani, F. Auricchio, R. Naghdabadi, A. Reali, S. Sohrabpour, A 3-D phenomenological constitutive model for shape memory alloys under multiaxial loadings, *Int. J. Plast.* 26 (7) (2010) 976–991.
- [28] J. Wang, Z. Moumni, W. Zhang, Y. Xu, W. Zaki, A 3D finite-strain-based constitutive model for shape memory alloys accounting for thermomechanical coupling and martensite reorientation, *Smart Mater. Struct.* 26 (6) (2017) 065006.
- [29] L. Brinson, R. Lammering, Finite element analysis of the behavior of shape memory alloys and their applications, *Int. J. Solids Struct.* 30 (23) (1993) 3261–3280.
- [30] M. Qidwai, D. Lagoudas, Numerical implementation of a shape memory alloy thermomechanical constitutive model using return mapping algorithms, *Internat. J. Numer. Methods Engrg.* 47 (6) (2000) 1123–1168.
- [31] J.C. Simo, T.J. Hughes, *Computational Inelasticity*, Vol. 7, Springer Science & Business Media, 2006.
- [32] J. Simo, J. Kennedy, R. Taylor, Complementary mixed finite element formulations for elastoplasticity, *Comput. Methods Appl. Mech. Engrg.* 74 (2) (1989) 177–206.
- [33] D.V. Kulkarni, D.A. Tortorelli, M. Wallin, A Newton–Schur alternative to the consistent tangent approach in computational plasticity, *Comput. Methods Appl. Mech. Engrg.* 196 (7) (2007) 1169–1177.
- [34] C. McAuliffe, H. Waisman, Mesh insensitive formulation for initiation and growth of shear bands using mixed finite elements, *Comput. Mech.* 51 (5) (2013) 807–823.
- [35] C. McAuliffe, H. Waisman, A Pian–Sumihara type element for modeling shear bands at finite deformation, *Comput. Mech.* 53 (5) (2014) 925–940.
- [36] C. McAuliffe, H. Waisman, A unified model for metal failure capturing shear banding and fracture, *Int. J. Plast.* 65 (2015) 131–151.
- [37] L. Berger-Vergiat, C. McAuliffe, H. Waisman, Isogeometric analysis of shear bands, *Comput. Mech.* 54 (2) (2014) 503–521.
- [38] C. McAuliffe, H. Waisman, A coupled phase field shear band model for ductile–brittle transition in notched plate impacts, *Comput. Methods Appl. Mech. Engrg.* 305 (2016) 173–195.
- [39] K.A. James, H. Waisman, Failure mitigation in optimal topology design using a coupled nonlinear continuum damage model, *Comput. Methods Appl. Mech. Engrg.* 268 (2014) 614–631.
- [40] D.C. Lagoudas, Z. Bo, M.A. Qidwai, A unified thermodynamic constitutive model for SMA and finite element analysis of active metal matrix composites, *Mech. Compos. Mater. Struct.* 3 (2) (1996) 153–179.
- [41] C. Kittel, H. Kroemer, *Thermal Physics*, American Association of Physics Teachers, 1998.
- [42] M. Qidwai, D. Lagoudas, On thermomechanics and transformation surfaces of polycrystalline NiTi shape memory alloy material, *Int. J. Plast.* 16 (10–11) (2000) 1309–1343.
- [43] L. Berger-Vergiat, C. McAuliffe, H. Waisman, Parallel preconditioners for monolithic solution of shear bands, *J. Comput. Phys.* 304 (2016) 359–379.
- [44] L. Berger-Vergiat, H. Waisman, An overlapping domain decomposition preconditioning method for monolithic solution of shear bands, *Comput. Methods Appl. Mech. Engrg.* 318 (2017) 33–60.
- [45] L. Svolos, L. Berger-Vergiat, H. Waisman, Updating strategy of a domain decomposition preconditioner for parallel solution of dynamic fracture problems, *J. Comput. Phys.* 422 (2020) 109746.
- [46] M. Ortiz, J.B. Martin, Symmetry-preserving return mapping algorithms and incrementally extremal paths: A unification of concepts, *Internat. J. Numer. Methods Engrg.* 28 (8) (1989) 1839–1853.
- [47] F. Zhang, *The Schur Complement and Its Applications*, Vol. 4, Springer Science & Business Media, 2006.
- [48] A.S. Householder, *Principles of Numerical Analysis*, Courier Corporation, 2006.