Comput. Methods Appl. Mech. Engrg. 198 (2009) 2286–2296

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

Return mapping for nonsmooth and multiscale elastoplasticity

Xuxin Tu, José E. Andrade *, Qiushi Chen

Theoretical and Applied Mechanics, Northwestern University, Evanston, IL 60208, USA

ARTICLE INFO

Article history:
Received 22 July 2008
Received in revised form 27 November 2008
Accepted 10 February 2009
Available online 20 February 2009

Keywords:
Nonsmooth
Semi-implicit
Stress integration
Micromechanics
Multiscale
Elastoplasticity

ABSTRACT

We present a semi-implicit return mapping algorithm for integrating generic nonsmooth elastoplastic models. The semi-implicit nature of the algorithm stems from "freezing" the plastic internal variables at their previous state, followed by implicitly integrating the stresses and plastic multiplier. The plastic internal variables are incrementally updated once convergence is achieved (*a posteriori*). Locally, the algorithm behaves as a classic return mapping for perfect plasticity and, hence, inherits the stability of implicit integrators. However, it differs from purely implicit integrators by keeping the plastic internal variables locally constant. This feature affords the method the ability to integrate nonsmooth (C_0) evolution laws that may not be integrable using implicit methods. As a result, we propose and use the algorithm as the backbone of a semi-concurrent multiscale framework, in which nonsmooth constitutive relationships can be directly extracted from the underlying micromechanical processes and faithfully incorporated into elastoplastic continuum models. Though accuracy of the proposed algorithm is step size-dependent, its simplicity and its remarkable ability to handle nonsmooth relations make the method promising and computationally appealing.

1. Introduction

Elastoplasticity is perhaps the most widely utilized and reliable framework used to capture material nonlinearities and inelastic behavior [1]. From metals to composites to aggregates, most solids can be simulated using elastoplastic models. Furthermore, many elastoplastic models make use of nonsmooth functions (in general, C_0 functions) to either represent yield surfaces, e.g. [2–4] or hardening (evolution) laws, e.g. [5,6]. In the case of cohesive-frictional materials, C_0 yield surfaces have been proposed to model two salient properties. On the one hand, the yield surface is generally dependent on the third invariant of stress. Using multiple smooth functions to describe the third invariant dependency [2,7] constituted one major source for nonsmoothness. On the other hand, cohesive-frictional materials feature very distinct responses under deviatoric and volumetric stresses. These two features have been accounted for by proposing models with two distinct yield surfaces, providing a potential source for discontinuities in the gradient function [3,4,8]. Nevertheless, the past decades have seen a great advance in the development of smooth yield surfaces aimed at capturing the behavior of complex geomaterials [5,9–13]. Naturally, smooth plastic potentials can also be derived based on their similarity to the yield surfaces.

In contrast, C_0 functions are very much still used to describe the evolution of internal plastic variables via nonsmooth hardening laws. It is well-known that the evolution of plastic internal variables (PIVs) is difficult to obtain and is mostly based on phenomenology. Hardening laws that conform well to experimental data may not necessarily yield smooth evolutions. Unsurprisingly, many nonsmooth hardening laws have been proposed to capture the behavior of complex elastoplastic materials accurately [5,6]. In this paper, we refer to a relation defining the variation of a PIV as an *evolution law*. Nonsmooth evolution laws permeate the plasticity literature. Accurately handling these C_0 evolution laws within a computational framework is not a trivial task and defines the objective of this work.

From a physics standpoint, one limitation of plasticity models emanates from the underlying phenomenology. Essentially, plasticity relations, especially evolution laws, are determined from limited experiments or simply based on empirical intuition. Furthermore, a plasticity model only describes an average behavior at the macroscopic scale but fails to account for the underlying microscale mechanisms. In contrast, multiscale computational approaches can derive the constitutive relationship from a fundamental level 'on-the-fly' [14–18]. In particular, for granular matter this fundamental level corresponds to the grain scale, from which the micromechanical phenomena—including particle geometry, force chains, fabric arrangement—intrinsically govern the macroscopic response of the material. These grain-scale phenomena can be explicitly simulated using micromechanical models [19–21]. An alternative and recently proposed technique is to link micromechanical models with elastoplasticity using a multiscale framework [18]. The main idea is to replace phenomenological

* Corresponding author. Tel.: +1 847 491 5884.
E-mail address: j-andrade@northwestern.edu (J.E. Andrade).

0045-7825/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2009.02.014

evolution laws with direct extraction of physically meaningful PIVs from the micromechanics. The resulting micromechanically-based evolution of PIVs is nonsmooth and falls within the realm of C_0 evolution laws tackled in this work.

Among the few previous efforts to address nonsmooth elastoplasticity problems, the nonsmooth Newton method [22] is responsible for laying down an important theoretical foundation for integrating nonsmooth plasticity relations. However, as pointed out in [22], the derivation of the method relies on the assumption of J_2-plasticity and, hence, the applicability of this method to other type of plasticity models (e.g. pressure-dependent models) remains to be determined. There have been other semi-implicit algorithms proposed in the literature (see [23,24] for example), but these have been aimed at explicitly integrating the hardening or evolution law and the flow rule, while the rest of the algorithm is fully implicit.

In this work, a simple semi-implicit algorithm is proposed to effectively combine the strengths of implicit and explicit architectures. Implicit integration algorithms easily lose their advantages when integrating nonsmooth relations. At the same time, though explicit algorithms have the ability of accommodating nonsmoothness, they may suffer shortcomings such as drifting and small critical time steps [24,25]. Further, combining explicit stress integrators with implicit FE schemes may be problematic [26]. We unveil a semi-implicit algorithm that conserves all the features of the implicit schemes except for integration of the plastic internal variables. Specifically, the method 'freezes' the plastic internal variables (PIVs) incrementally. Hence, the method resembles implicit perfect plasticity integrators at the local level and therefore inherits unconditional stability. The PIVs are then updated *a posteriori* at every time increment. The combination of local freezing and the *a posteriori* update of PIVs affords the method the ability to handle nonsmooth (C_0) evolution laws. It is also shown that incremental updating is efficient computationally and its application to recent multiscale techniques will be clearly demonstrated. The robustness and accuracy of the proposed algorithm is investigated using several numerical examples.

This paper is organized as follows. Section 2 summarizes the rate elastoplasticity formulation and presents the classic implicit return mapping scheme. In Section 3, the proposed semi-implicit algorithm is presented based on the implicit return mapping algorithm. Section 4 presents a detailed verification of the semi-implicit algorithm where we focus on boundary value problems to assess accuracy and robustness (convergence) of the algorithm against the backdrop of the fully implicit return mapping integrator. We conclude that the incrementally updated semi-implicit algorithm furnishes an appropriate balance between accuracy and robustness and, as a result, we utilize this method to perform proof-of-concept micromechanically-based semi-concurrent multiscale computations in Section 5. We summarize our findings and make some closing remarks in the last section.

As for notations and symbols used in this paper, bold-faced letters denote tensors or vectors; the symbol '$\cdot$' denotes an inner product of two vectors (e.g. $\boldsymbol{a}\cdot\boldsymbol{b}=a_ib_i$), or a single contraction of adjacent indices of two tensors (e.g. $\boldsymbol{c}\cdot\boldsymbol{d}=c_{ij}d_{jk}$); the symbol '$\otimes$' denotes a tensorial (or dyadic) product (e.g. $\boldsymbol{a}\otimes\boldsymbol{b}=a_ib_j$, or $\boldsymbol{\alpha}\otimes\boldsymbol{\beta}=\alpha_{ij}\beta_{kl}$); the symbol ':' denotes an inner product of two second-order tensors (e.g. $\boldsymbol{c}:\boldsymbol{d}=c_{ij}d_{ij}$); the symbol '$||\cdot||$' denotes an L_2 norm of a vector, e.g. $||\boldsymbol{e}||=(\boldsymbol{e}\cdot\boldsymbol{e})^{1/2}$ or a tensor $\|\boldsymbol{A}\|=(\boldsymbol{A}:\boldsymbol{A})^{1/2}$. Stress and strain are expressed in Voigt notation, and as a result, the associated stiffness/compliance are expressed as matrices.

2. Infinitesimal elastoplasticity and implicit integrators

The most salient ingredients of the infinitesimal elastoplasticity theory are [27]:

- Additive decomposition of strain rate into elastic and plastic components, i.e., $\dot{\boldsymbol{\epsilon}}=\dot{\boldsymbol{\epsilon}}^{\mathrm{e}}+\dot{\boldsymbol{\epsilon}}^{\mathrm{p}}$.
- Generalized Hooke's law, i.e., $\dot{\boldsymbol{\sigma}}=\boldsymbol{c}^{\mathrm{e}}:\dot{\boldsymbol{\epsilon}}^{\mathrm{e}}$, where $\boldsymbol{c}^{\mathrm{e}}$ is the elastic constitutive tensor.
- Elastic domain and yield condition such that the yield surface $F=0$ defines the limit of the elastic domain.
- Non-associative plastic flow rule, i.e., $\dot{\boldsymbol{\epsilon}}^{\mathrm{p}}=\dot{\lambda}\boldsymbol{g}$, where $\dot{\lambda}\geqslant 0$ is the consistency parameter or plastic multiplier and $\boldsymbol{g}:=\partial G/\partial\boldsymbol{\sigma}$ is the direction of the plastic flow, where G is the plastic potential function.
- Evolution laws for the PIVs involved in F and G. In this paper, we use a vector $\boldsymbol{\alpha}$ to represent the set of PIVs. In classical infinitesimal elastoplasticity, the evolution relations for the PIVs are typically cast in rate-form, $\dot{\boldsymbol{\alpha}}=\dot{\lambda}\hat{\boldsymbol{\alpha}}(\boldsymbol{\sigma},\boldsymbol{\alpha})$.
- The Kuhn–Tucker optimality condition, $\dot{\lambda}F=0$, which induces the consistency requirement $\dot{\lambda}\dot{F}=0$.

The aforementioned ingredients are the foundation for most plasticity models available, which are typically integrated numerically into a finite element (FE) or finite difference code. Numerical integration of these models is crucial for successful modeling of boundary value problems in engineering. A well-established integration technique is the implicit return mapping algorithm. A schematic showing the role of the implicit return mapping in the material subroutine inside a FE code is shown in Fig. 1. As shown in this flowchart, the material subroutine is at the heart of the FE code and its main purpose is to compute, given an increment in the strain $\Delta\boldsymbol{\epsilon}$, the resulting incremental change in state, i.e., $\Delta\boldsymbol{\sigma}$ and $\Delta\boldsymbol{\alpha}$. Here we use the incremental notation $\Delta\square:=\square_{n+1}-\square_n$, where $\square_{n+1}$ corresponds to the value of the function evaluated at time station t_{n+1}. In addition, the material subroutine computes the consistent tangent algorithm defined as $\boldsymbol{c}=\partial\boldsymbol{\sigma}_{n+1}/\partial\boldsymbol{\epsilon}_{n+1}$. The consistent tangent is available in closed-form when implicit integrators are invoked and this is one of the reasons that make implicit algorithms appealing. Consistent tangent operators afford implicit nonlinear FE codes asymptotic rates of convergence, a key feature for efficient engineering analyses.

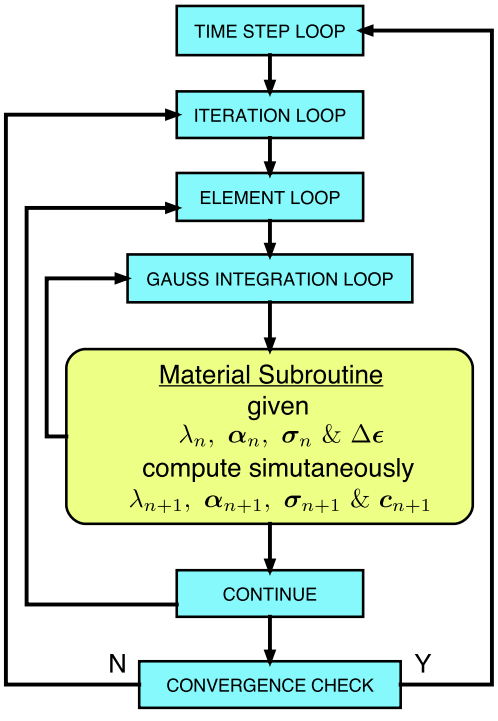

Fig. 1. Flowchart for an implicit return mapping algorithm within a FE code.

Implicit return mappings rely heavily on Newton–Raphson schemes to iteratively arrive at a solution [12,28,29]. These schemes typically construct residual vectors $\boldsymbol{r}$ as a function of the unknowns $\boldsymbol{x}$, i.e.,

$$\boldsymbol{r}(\boldsymbol{x}) = \begin{Bmatrix} \boldsymbol{c}^{e-1} : \Delta\boldsymbol{\sigma} + \Delta\lambda G_{,\boldsymbol{\sigma}} - \Delta\boldsymbol{\epsilon} \\ \Delta\boldsymbol{\alpha} - \Delta\lambda\hat{\boldsymbol{\alpha}}(\boldsymbol{\sigma},\boldsymbol{\alpha}) \\ F(\boldsymbol{\sigma},\boldsymbol{\alpha}) \end{Bmatrix}; \quad \boldsymbol{x} = \begin{Bmatrix} \boldsymbol{\sigma} \\ \boldsymbol{\alpha} \\ \Delta\lambda \end{Bmatrix}, \tag{2.1}$$

where $\boldsymbol{c}^e$ is the linear elastic stiffness matrix and $\Delta\lambda$ is the discrete consistency parameter. Solution to the local system of generally nonlinear equations is achieved when $\boldsymbol{r}(\boldsymbol{x}) = \boldsymbol{0}$ and the rate of convergence is intimately dependent on the consistent local tangent (Jacobian) such that

$$\boldsymbol{r}_{,\boldsymbol{x}} = \begin{bmatrix} \boldsymbol{c}^{e-1} + \Delta\lambda G_{,\boldsymbol{\sigma}\boldsymbol{\sigma}} & \Delta\lambda G_{,\boldsymbol{\sigma}\boldsymbol{\alpha}} & G_{,\boldsymbol{\sigma}} \\ -\Delta\lambda\hat{\boldsymbol{\alpha}}_{,\boldsymbol{\sigma}} & \boldsymbol{\delta} - \Delta\lambda\hat{\boldsymbol{\alpha}}_{,\boldsymbol{\alpha}} & -\hat{\boldsymbol{\alpha}} \\ F_{,\boldsymbol{\sigma}} & F_{,\boldsymbol{\alpha}} & 0 \end{bmatrix}, \tag{2.2}$$

where $\boldsymbol{\delta}$ is the second-order identity tensor. The above Jacobian underscores the potential issues related to accommodating nonsmooth evolution laws for $\boldsymbol{\alpha}$. If these functions are only C_0, the required derivatives appearing in the local Jacobian may not be continuous or may not even be defined. By way of example, we will show that this lack of continuity in the derivatives of the evolution laws could be detrimental in the convergence of the local integration algorithm and, as a result, that of the global computation. The next section describes a plausible alternative to fully implicit algorithms where the Jacobian matrix does not require evaluation of the derivatives of the evolution laws, making it possible to handle C_0 evolution functions.

Remark 1. If the formulation is isotropic, the yield surface F and plastic potential G can be expressed as a function of the stress invariants and the spectral decomposition can be exploited. These algorithms are efficient since they reduce the number of unknowns from full stress space to principal stress space. The interested reader is referred to [29] for an elaboration of these types of algorithms.

3. The semi-implicit return mapping algorithm

The implicit algorithm introduced in the foregoing section is a classic approach to integrate plasticity models. Under optimal conditions, this algorithm is able to achieve asymptotic quadratic convergence rates, first order accuracy, while featuring unconditional stability. However, in the presence of nonsmoothness, the implicit approach may not be suitable. As shown in Eq. (2.2), the local Jacobian, and hence the convergence of the algorithm, depend crucially on the computability of the necessary gradients. In the case of C_0 evolution laws, it is clear that convergence rates could be severely affected and the algorithm may diverge altogether. It is well-known that the Newton–Raphson scheme will have serious issues converging near inflection points. Hence, it is often difficult, sometimes almost impossible, to use the conventional implicit method to integrate plasticity models with nonsmooth evolution relations (e.g. emanating from complex micromechanical substructures) [25,30].

To ameliorate the shortcomings of fully implicit schemes in the context of C_0 evolution laws, we propose a simple semi-implicit scheme. The main procedure is simple and it involves freezing the plastic internal variables (PIVs) in the model at their previous, converged value. If the solution at time station t_{n+1} is being pursued, the PIVs in the model are fixed at their value at time station t_n, or $\boldsymbol{\alpha}_n$. This strategy of freezing the PIVs is different from previous semi-implicit algorithms such as those presented in [23,24], where the plastic flow and moduli are explicitly integrated.

A flowchart explicating the semi-implicit return mapping algorithm is given in Fig. 2. Comparing the new semi-implicit scheme in Fig. 2 with the fully implicit algorithm in Fig. 1, it is clear that the material subroutine only updates the stresses $\boldsymbol{\sigma}$ and the plastic increment $\Delta\lambda$ at t_{n+1}, while keeping the PIVs fixed at their previous t_n value. Accordingly, the unknown vector $\boldsymbol{x}$ and the corresponding residual $\boldsymbol{r}$ read

$$\boldsymbol{x} = \begin{Bmatrix} \boldsymbol{\sigma} \\ \Delta\lambda \end{Bmatrix}, \quad \boldsymbol{r}(\boldsymbol{x}) = \begin{Bmatrix} (\boldsymbol{c}^e)^{-1} \cdot \Delta\boldsymbol{\sigma} + \Delta\lambda G_{,\boldsymbol{\sigma}} - \Delta\boldsymbol{\epsilon} \\ F(\boldsymbol{\sigma}) \end{Bmatrix}. \tag{3.1}$$

Note the reduction in the number of unknowns and the resulting disappearance of the derivatives of the PIVs, cf., Eq. (2.2). In general, it is still necessary to invoke the Newton–Raphson locally to solve for $\boldsymbol{x}$. Hence, the local Jacobian is defined such that

$$\boldsymbol{r}_{,\boldsymbol{x}} = \begin{bmatrix} \boldsymbol{a} & \boldsymbol{g} \\ \boldsymbol{f} & 0 \end{bmatrix}; \quad \boldsymbol{a} := (\boldsymbol{c}^e)^{-1} + \Delta\lambda\boldsymbol{\Lambda}; \quad \boldsymbol{\Lambda} := G_{,\boldsymbol{\sigma}\boldsymbol{\sigma}}; \quad \boldsymbol{g} := G_{,\boldsymbol{\sigma}}; \quad \boldsymbol{f} := F_{,\boldsymbol{\sigma}}. \tag{3.2}$$

The consistent tangent operator $\boldsymbol{c} = \partial\boldsymbol{\sigma}_{n+1}/\partial\boldsymbol{\epsilon}_{n+1}$ is obtained in the standard form, similar to the fully implicit algorithms, by exploiting the converged residual function [12,31,32], i.e.,

$$\boldsymbol{c} = \boldsymbol{a}^{-1} - \frac{1}{\bar{\chi}}\boldsymbol{a}^{-1} : \boldsymbol{g} \otimes \boldsymbol{f} : \boldsymbol{a}^{-1}; \quad \bar{\chi} = \boldsymbol{g} : \boldsymbol{a}^{-1} : \boldsymbol{f}, \tag{3.3}$$

where one can show that $\boldsymbol{c}$ corresponds to the upper fourth-order tensor of the inverse of the local jacobian $\boldsymbol{r}_{,\boldsymbol{x}}$. It is interesting to note

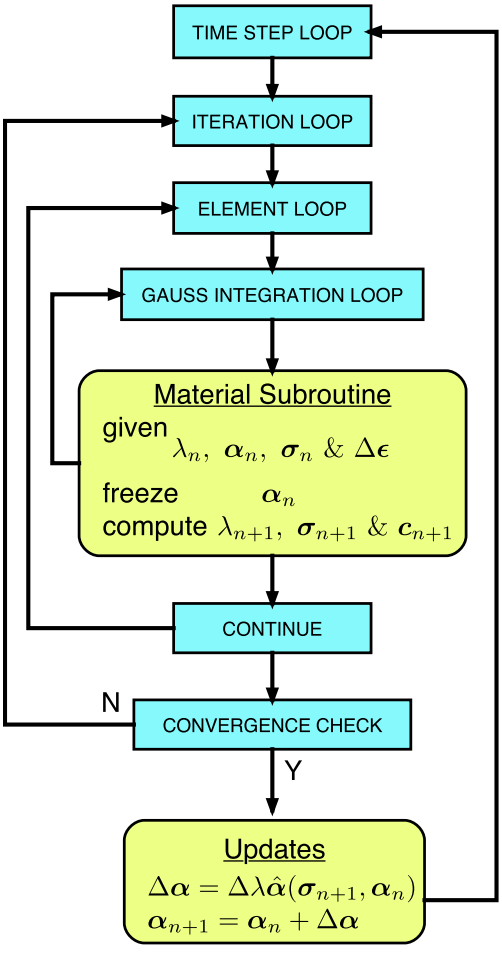

Fig. 2. Flowchart for the semi-implicit return mapping algorithm within a FE code.

the similarity between the consistent tangent and the continuum elastoplastic tangent for perfect plasticity, i.e.,

$$\boldsymbol{c}^{\mathrm{ep}} = \boldsymbol{c}^{\mathrm{e}} - \frac{1}{\chi}\boldsymbol{c}^{\mathrm{e}} : \boldsymbol{g} \otimes \boldsymbol{f} : \boldsymbol{c}^{\mathrm{e}}; \quad \chi = \boldsymbol{g} : \boldsymbol{c}^{\mathrm{e}} : \boldsymbol{f}. \tag{3.4}$$

For the case of a two-invariant model, such as Drucker–Prager, the above return mapping converges in one iteration and the state is obtained directly such that,

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^{\mathrm{tr}} - \Delta\lambda \boldsymbol{g}; \quad \boldsymbol{\sigma}^{\mathrm{tr}} = \boldsymbol{\sigma}_n + \boldsymbol{c}^{\mathrm{e}} : \Delta\boldsymbol{\epsilon} \tag{3.5}$$

and

$$\Delta\lambda = \frac{F^{\mathrm{tr}}}{\chi}; \quad F^{\mathrm{tr}} = F(\boldsymbol{\sigma}^{\mathrm{tr}}). \tag{3.6}$$

These equations of state for the stress $\boldsymbol{\sigma}$ and the plastic multiplier $\Delta\lambda$ are obtained departing from a trial state, i.e., $\boldsymbol{\sigma} = \boldsymbol{\sigma}^{\mathrm{tr}}$ and $\Delta\lambda = 0$. The isotropy of the linear elastic model and the yield and plastic potential functions imply coaxiality, which leads to $\boldsymbol{f} = \boldsymbol{f}^{\mathrm{tr}}$ and $\boldsymbol{g} = \boldsymbol{g}^{\mathrm{tr}}$ in the Drucker–Prager model, where the trial gradients are simply the gradients of the yield function and plastic potential evaluated at the trial stress $\boldsymbol{\sigma}^{\mathrm{tr}}$. A geometrical interpretation for the algorithm in stress space is given in Fig. 3. From this figure and from Eqs. (3.5) and (3.6), it can be appreciated that the converged state is only a function of the trial state and therefore can be obtained without iterations. Finally, based on Eq. (3.5) a simplified closed-form expression for the consistent tangent operator is obtained, i.e.,

$$\boldsymbol{c} = \underbrace{\boldsymbol{c}^{\mathrm{ep}}}_{\text{continuum tangent}} - \underbrace{\Delta\lambda \boldsymbol{c}^{\mathrm{e}} : \boldsymbol{\Lambda} : \boldsymbol{c}^{\mathrm{e}}}_{\text{algorithmic tangent}}, \tag{3.7}$$

where one can observe the $O(\Delta\lambda)$ contribution from the algorithmic tangent.

Remark 2. The plastic internal variables (PIVs) are not updated until the global equations of motion have been satisfied at the global level. Fig. 2 shows the updating procedure. In essence, the PIVs are direct functions of the converged values of stress $\boldsymbol{\sigma}$ and the plastic multiplier $\Delta\lambda$. These PIVs are used for the next time step calculation and kept frozen until the subsequent converged state is achieved.

To bypass potential problems with nonsmooth evolution (C_0) functions, the semi-explicit algorithm presented above, freezes the plastic internal variables at the previous time station, effectively behaving as a perfectly plastic material for a given time step. Similarly, truly explicit algorithms, e.g. [33,34] will also be able to bypass issues related to C_0 functions for the evolution laws. However, the explicit algorithms have two potential shortcomings. First, explicit algorithms generally need to be corrected to prevent yield surface from 'drifting', i.e., a violation of the consistency condition [24,25]. Furthermore, explicit stress integration is better employed within an explicit FE framework, e.g. [26], as there is no closed-form solution for the consistent tangent operator. In fact, it has been shown that the derivation of such a CTO can be quite tedious [35] and could necessitate numerical differentiation [30,36], which is computationally expensive. In contrast, the semi-implicit algorithm presented above combines the advantages of implicit and explicit methods.

Remark 3. It can be seen that the main shortcoming of the semi-implicit method will be potential lack of accuracy stemming from the frozen plasticity. However, as Figs. 2 and 3 show, the stress is corrected to enforce consistency, i.e., $F_{n+1} = F(\boldsymbol{\sigma}_{n+1}, \boldsymbol{\alpha}_n) = 0$, where the PIVs are frozen at their values at t_n. This inaccuracy should not be confused with drifting, which is typically defined as $F_{n+1} \neq 0$ in explicit schemes (see [24, p. 277]).

4. Verification: application to cohesive-frictional plasticity

In this section, and without loss of generality, we apply the semi-implicit return mapping to a Mohr–Coulomb-type model exemplified by the classic linear elastic–plastic Drucker–Prager model with nonlinear hardening/softening [37]. Naturally, we will demonstrate the robustness of the method within the context of C_0 evolution laws for the plastic internal variables involved. The elastic region of the model is furnished by the linear tangent such that

$$\boldsymbol{c}^{\mathrm{e}} = K\boldsymbol{\delta} \otimes \boldsymbol{\delta} + 2\mu\left(\boldsymbol{I} - \frac{1}{3}\boldsymbol{\delta} \otimes \boldsymbol{\delta}\right), \tag{4.1}$$

where K and μ are the constant elastic bulk and shear moduli, $\boldsymbol{\delta}$ is the second-order identity tensor, and $\boldsymbol{I}$ is its fourth-order counterpart. Within this context, we can define two invariants of the stress tensor such that

$$p = \frac{1}{3}\mathrm{tr}\,\boldsymbol{\sigma}; \quad q = \sqrt{\frac{3}{2}}\|\boldsymbol{s}\|, \tag{4.2}$$

where $\mathrm{tr}\,\square = \square : \boldsymbol{\delta}$ is the trace operator, and $\boldsymbol{s} = \boldsymbol{\sigma} - p\boldsymbol{\delta}$ is the deviatoric component of the stress tensor. Similarly, the invariants of the strain rate tensor (total, elastic, or plastic) are defined as

$$\dot{\epsilon}_{\mathrm{v}} = \mathrm{tr}\,\dot{\boldsymbol{\epsilon}}; \quad \dot{\epsilon}_{\mathrm{s}} = \sqrt{\frac{2}{3}}\|\dot{\boldsymbol{e}}\|, \tag{4.3}$$

where $\dot{\boldsymbol{e}} = \dot{\boldsymbol{\epsilon}} - 1/3\,\dot{\epsilon}_{\mathrm{v}}\boldsymbol{\delta}$ is the deviatoric component of the strain tensor.

Using the aforementioned invariants of the stress tensor, we can define the yield surface and plastic potential for the Drucker–Prager (D–P) model:

$$\begin{aligned} F &= q + \alpha p - c_{\mathrm{f}}, \\ G &= q + \beta p - c_{\mathrm{q}}. \end{aligned} \tag{4.4}$$

Typically, the cohesion parameter $c_{\mathrm{f}} = 0$ for granular materials, while the cohesion-like parameter c_{q} is to be adjusted so that the potential surface G is always attached to the current stress point. Two evolution parameters are involved in the D–P model—the fric-

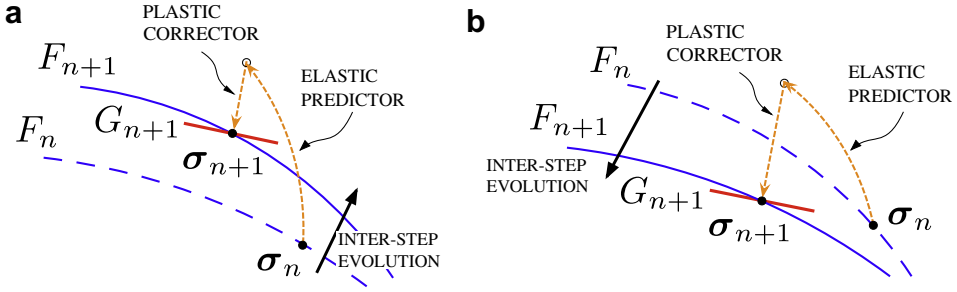

Fig. 3. Two scenarios for the semi-implicit algorithm: (a) hardening and (b) softening.

tion resistance α and the dilatancy parameter β. For $c_f = 0$ (assumed henceforth) and at yielding, the friction parameter takes the form

$$\alpha = -\frac{q}{p}. \tag{4.5}$$

Note that the only allowable states of stress when $c_f = 0$ are compressive, i.e., $p < 0$. The physical interpretation for the plastic internal variable α is that it directly represents the mobilized friction angle of the granular material. Hence, α indicates the mobilized friction resistance at any given state. Invoking the non-associative flow rule, one can show that the volumetric and deviatoric invariants of the plastic strain rate tensor are defined by

$$\dot{\epsilon}_v^p = \dot{\lambda}\frac{\partial G}{\partial p}; \quad \dot{\epsilon}_s^p = \dot{\lambda}\frac{\partial G}{\partial q}. \tag{4.6}$$

For the D–P model presented here, it turns out that the dilatancy β takes the form

$$\beta = \frac{\dot{\epsilon}_v^p}{\dot{\epsilon}_s^p}. \tag{4.7}$$

Similar to the friction coefficient α, the dilatancy β measures the change in volumetric plastic deformations for a given change in deviatoric plastic deformations. Reynolds in 1885 coined the term and pointed out its crucial role in the mechanical behavior of granular media [38]. Finally, the corresponding gradients to the yield surface and plastic potential are given such that

$$\begin{aligned} \boldsymbol{f} &= \frac{1}{3}\alpha\boldsymbol{\delta} + \sqrt{\frac{3}{2}}\hat{\boldsymbol{n}}, \\ \boldsymbol{g} &= \frac{1}{3}\beta\boldsymbol{\delta} + \sqrt{\frac{3}{2}}\hat{\boldsymbol{n}}, \end{aligned} \tag{4.8}$$

where $\hat{\boldsymbol{n}} = \boldsymbol{s}/\|\boldsymbol{s}\|$ is the unit deviatoric tensor. Due to coaxiality, it can be shown that the deviatoric unit tensor can be defined using the trial stress tensor, i.e., $\hat{\boldsymbol{n}} = \boldsymbol{s}^{tr}/\|\boldsymbol{s}^{tr}\|$ and, consequently, $\boldsymbol{f} = \boldsymbol{f}^{tr}$ and $\boldsymbol{g} = \boldsymbol{g}^{tr}$.

In what follows, different evolution laws for the PIVs α and β will be considered to evaluate the accuracy, stability, and efficiency of the proposed semi-implicit algorithm against the backdrop of its fully implicit counterpart.

4.1. Smooth evolution law

The accuracy, stability, and efficiency of the semi-implicit integration technique will be evaluated in this section. A smooth evolution law will be considered to provide both the semi-implicit and fully-implicit algorithms the same datum to make meaningful comparisons.

Consider the following smooth evolution laws for the friction and dilatancy parameters, respectively

$$\begin{aligned} \alpha &= a_0 + a_1\lambda\exp(a_2 p - a_3\lambda), \\ \beta &= \alpha - \beta_0, \end{aligned} \tag{4.9}$$

where a_0, a_1, a_2, a_3 and β_0 are (positive) material constants, and λ is the cumulative plastic multiplier. It is clear that the evolution laws above are highly nonlinear and state-dependent. Note that the friction resistance α and the dilatancy parameter β differ by a constant β_0, which is amenable to the stress-dilatancy relation widely observed in granular media [18,39,40]. The evolution laws defined above were introduced in [29] to test the robustness of fully-implicit return mapping algorithms. Similar to the values used in [29], which apply to soils, we use $a_0 = 0.7$, $a_1 = 50$, $a_2 = 0.0005/\text{kPa}$, $a_3 = 50$ and $\beta_0 = 0.7$. For the elastic parameters, we use $E = 25000$ kPa and $\nu = 0.3$.

Here, we will perform plane-strain compression 'experiments' under constant confinement. These experiments will furnish homogeneous BVPs that can be used to assess accuracy, stability and rate of convergence at the global level. The specimens are initially isotropically consolidated to a hydrostatic state of $p_0 = 50$ kPa. Subsequently, the specimens are sheared under constant lateral confining stress σ_3^* but increasing axial strain ϵ_1. The axial strain is increased by $\Delta\epsilon_1 = 0.3\%$ until the cumulative strain reaches about 10%. This situation allows us to define the global scalar residual function such that $R(\epsilon_3) = \sigma_3^* - \sigma_3(\epsilon_3)$, where we underscore the dependence of the residual function on the unknown lateral strain ϵ_3. Hence, the solution of the problem is $R = 0$ when we have found an appropriate ϵ_3 such that the calculated lateral stress σ_3 equals the prescribed lateral stress σ_3^*, for a given axial strain ϵ_1. The convergence criterion for the BVP is given such that

$$|R|/|R_0| < 10^{-10}, \tag{4.10}$$

where R_0 is the initial residual.

Fig. 4 shows the results of the experiments for both numerical integration techniques. It can be seen that both the stress–strain response and the volume-strain evolution are captured very well by the semi-implicit algorithm. The peak stress is captured correctly with a slight delay due to the PIVs lagging (freezing). Overall, we can conclude that the results for both algorithms are comparable. Similarly, it is important to compare the rate of convergence globally to get a sense for the efficiency of the method in implicit codes where the consistent tangent operator is needed. Fig. 5 shows the semi-log plot of the normalized residual degradation curves in two typical load steps for each integration algorithm. One convergence profile is reported pre-peak and the other post-peak. Clearly, the convergence rates of both algorithms are asymptotically quadratic. Convergence profiles at all other time steps are also asymptotically quadratic. As far as convergence is concerned, these results suggest that the semi-implicit algorithm is capable of delivering the same advantage as its implicit counterpart.

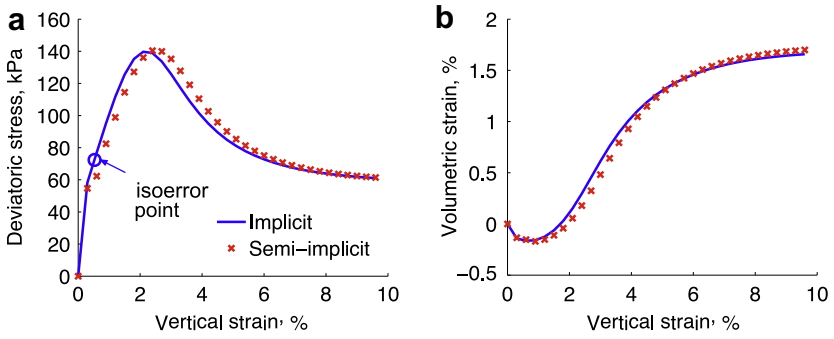

Fig. 4. Integration of the smooth evolution relation under plane-strain compression: (a) stress response and (b) strain response.

Finally, to assess the accuracy of the scheme in a more quantitative fashion, isoerror analysis was performed. This numerical tool is typically employed to quantify the percent error of a solution compared to an 'exact' solution for one time step and under homogeneous conditions [27–29]. Fig. 6 shows an isoerror map generated using various combinations of $(\Delta\epsilon_1, \Delta\epsilon_3)$. The semi-implicit algorithm was used in all computations, starting from the same "isoerror point" shown in Fig. 4a ($\sigma_1 = -131.2$ kPa, $\sigma_2 = -83.0$ kPa, $\sigma_3 = -50.0$ kPa). Each computation of $(\Delta\epsilon_1, \Delta\epsilon_3)$ was first prescribed in a single step, and the computed stress is denoted by $\boldsymbol{\sigma}$. Then, we calculated the 'exact' stress $\boldsymbol{\sigma}^*$ by subdividing the strain increment of $(\Delta\epsilon_1, \Delta\epsilon_3)$ until further refinement produces negligible changes in the resulting stress. The relative error was calculated from the equation

$$ERR := \frac{||\boldsymbol{\sigma} - \boldsymbol{\sigma}^*||}{||\boldsymbol{\sigma}^*||} \times 100\%. \tag{4.11}$$

The step-size dependent error is represented by the isolines in Fig. 6, where negative strain increment is compressive. As expected, accuracy generally deteriorates as the strain increments increase. Nevertheless, increases of up to 0.1% in the strain increment, which is large, yield errors below 2%, which is generally acceptable.

These results suggest an equivalence between the semi-implicit and implicit return mappings under smooth conditions. Generally, implicit methods claim greater stability, good accuracy and quadratic convergence profiles. This example has shown that the semi-implicit return mapping proposed can claim similar properties. In what follows, we will show a case where the semi-implicit algorithm performs much better than its implicit counterpart.

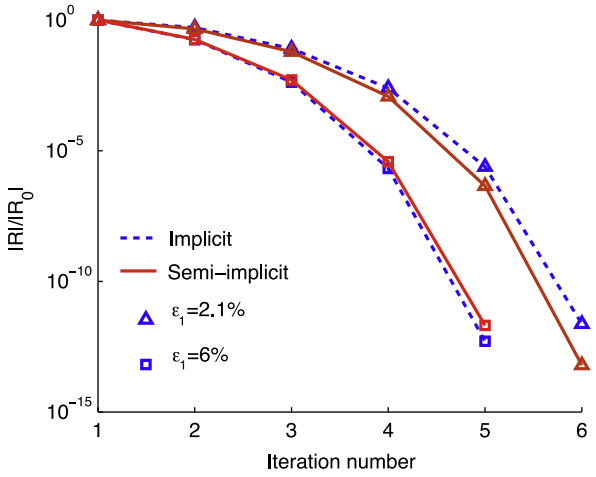

Fig. 5. Residual degradation for plane-strain problem with smooth evolution law.

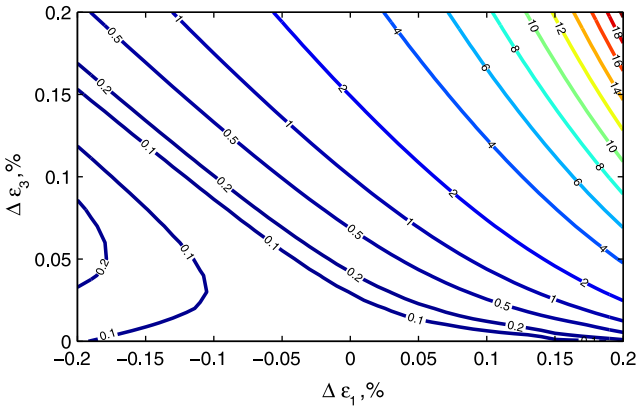

Fig. 6. Isomaps for the semi-implicit algorithm relative to the 'exact' solution.

4.2. Nonsmooth (C_0) evolution law

In this section, the robustness of the semi-implicit method in handling C_0 evolution laws will be demonstrated by way of a numerical example. As mentioned earlier, the complexity of granular materials often requires the use of highly nonlinear and often empirical evolution laws for the plastic internal variables. It is not uncommon for evolution laws to contain ranges over which the evolutions are continuous but introduce kinks at the intersections. One such evolution law was proposed by Lade to simulate the behavior of granular materials [5,6]. Consider the following evolution for the frictional resistance and dilatancy, respectively [5,6],

$$\alpha = \begin{cases} \beta_0 + h_1\lambda & \text{if } \lambda \leqslant l, \\ \beta_0 + h_1 l + h_2(\lambda - l) & \text{if } \lambda > l, \end{cases} \tag{4.12}$$

$$\beta = \alpha - \beta_0. \tag{4.13}$$

Fig. 7 shows the plot of the evolution law proposed for α, labeled as 'imposed' since this function effectively imposes the allowable values for the stress ratio $-q/p$. From Fig. 7 and Eq. (4.12), it can be observed that the evolution law for the friction parameter is bilinear, with a potential change in slope from h_1 to h_2 at $\lambda = l$. Hence, if $h_1 \neq h_2$, as it is usually the case, the derivative function is discontinuous at $\lambda = l$. This discontinuity will make it difficult for fully implicit return mapping to converge.

For this example, we have chosen the following material parameters: $\beta_0 = 0.7, h_1 = 20, h_2 = -20$ and $l = 0.09$. We perform axisymmetric compression simulations using the implicit return mapping and the semi-implicit algorithm within the context of the Drucker–Prager model presented above. The numerical

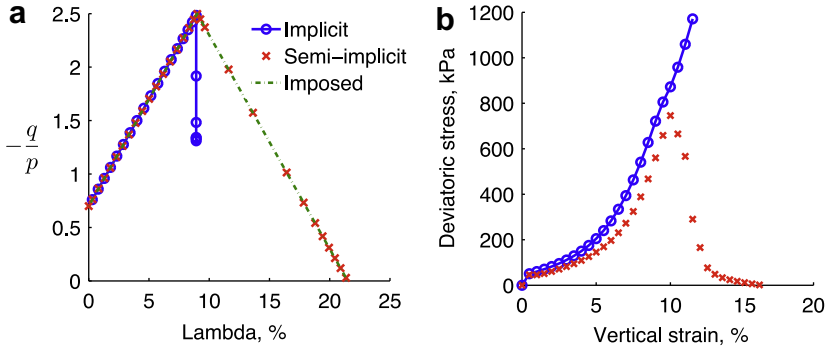

Fig. 7. Integration of nonsmooth evolution law (a) friction evolution and (b) stress–strain curve.

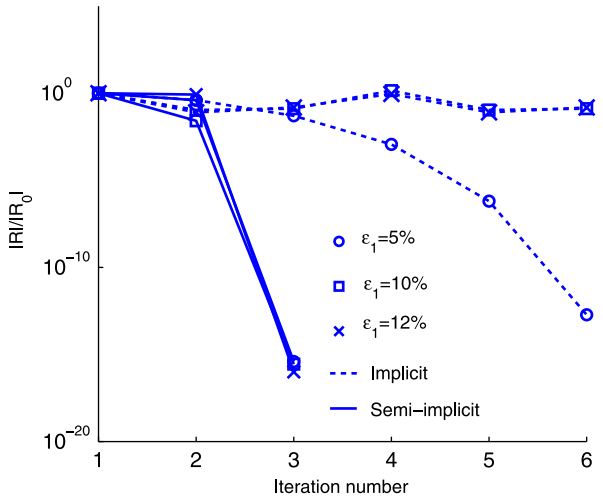

Fig. 8. Convergence profile for nonsmooth evolution law at various time steps.

example is started from a state of hydrostatic compression of $p_0 = 50$ kPa and then the confining stress is held constant with an increasing axial strain at a rate of $\Delta\epsilon_1 = 0.5\%$ in compression. Similar to the previous example, the axisymmetric compression simulation furnishes a BVP with mixed boundary conditions and a global residual where the confining stress is prescribed and must be matched by the computed lateral stress. Of course, the global convergence of the problem depends crucially on the local performance of the integration algorithm.

The results of the simulations are shown in Fig. 7. Clearly, one measure of success, is for the computed stress ratio $-q/p$ to follow the 'imposed' evolution of α. Fig. 7 shows that the semi-implicit algorithm is capable of reproducing the imposed evolution of the friction parameter α before and after the peak. On the other hand, the fully implicit algorithm runs into trouble near the peak, losing convergence and producing spurious results. Part of the problem is explained by the global convergence profiles reported in Fig. 8. It can be seen that both algorithms converge quadratically in the hardening regime. Near the peak, however, the implicit algorithm loses its convergence and finally diverges. In contrast, the convergence profile for the semi-implicit algorithm is undeterred even during the softening regime.

These results clearly show the ability of the semi-implicit method to efficiently handle C_0 functions describing the evolution laws necessitated to perform computations using elastoplastic models. Nevertheless, in the next section, a new class of nonsmooth evolutions for the PIVs will be introduced.

5. Application to multiscale plasticity

In an effort to capture the micromechanical effects governing the behavior of granular media, macroscopic phenomenological models have been introduced. These models have had relative success modeling the behavior of granular materials using plasticity theory and phenomenological evolution laws (e.g. the nonsmooth evolution shown in the previous example [5,6]). However, it is now well accepted that these phenomenological laws break down outside of the realm of the boundary conditions used to develop them. For example, it is not uncommon for an evolution law to break down under plane strain if it was developed under axisymmetric conditions. For this reason, micromechanical models such as the discrete element method (DEM) [41] have been proposed. Unfortunately, micromechanical models such as DEM are very computationally intensive and will not be able to tackle engineering scale problems for the next 20 years [42]. Therefore, similar to Molecular Dynamics computations, these discrete methods have introduced a bottleneck in engineering computations, ameliorated by the advent of multiscale methods.

The key idea of multiscale methods is to retain high fidelity where necessary and use continuum (phenomenological) approximations elsewhere. In general, multiscale methods can be classified as either hierarchical or concurrent [16]. Hierarchical methods use information from the smaller scale as input to the relation for the larger scale. On the other hand, concurrent methods apply models at different scales to different domains and run them simultaneously. In an effort to capture the behavior of granular materials accurately while bypassing phenomenological evolution laws, Andrade and Tu have proposed a semi-concurrent multiscale method for updating Drucker–Prager-type models [18].

The basic idea behind the semi-concurrent multiscale method is to link the granular scale and the continuum scale by extracting the evolution of the basic plastic variables α and β directly from the grain scale computations. Fig. 9 shows the basic recipe for the method. Comparing Figs. 9 and 2, one realizes that the algorithms are form-identical, with the only difference being that the update in the multiscale model is performed directly at the grainscale and then passed back to the continuum plasticity model. Hence, the semi-implicit algorithm presented herein is at the heart of the multiscale computational procedure proposed in [18].

5.1. Unit cell computations and PIV evolution

In the semi-concurrent multiscale scheme, and as shown in Fig. 9, the update of the PIVs is performed at the so-called unit cell

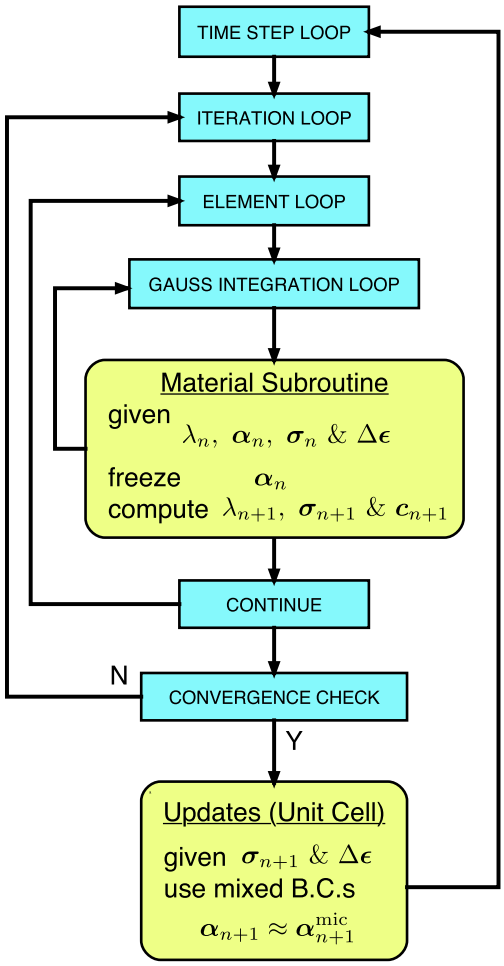

Fig. 9. Flowchart for the hierarchical multiscale scheme.

and then this continuum information is passed to the plasticity model, e.g. [17]. The unit cell contains a certain physical volume of microstructure, from which continuum quantities (the critical parameters) are computed. A closely related concept is the so-called representative volume element (RVE), defined as the smallest possible region representative of the whole heterogeneous media, on average [43]. Unlike the RVE, the unit cell may not necessarily represent the behavior of the entire domain. However, similar to the RVE, the unit cell is a finite physical domain where a continuum description is applicable (high frequency oscillations are not present in a given continuum quantity, e.g. dilatancy). In a multiscale framework using FE, the unit cell can be selected to cover a representative area around a Gauss point, resembling the local Quasi-Continuum strategy [14]. In Fig. 10a, for instance, the unit cell corresponds to the hatched area outlined by the so-called ghost nodes. Alternatively, the whole finite element can be taken as a unit cell, or the unit cell can be allowed to cover multiple elements, resembling the non-local Quasi-Continuum [14].

The unit cell contains a configuration of the microstructure, associated with a specific Gauss point. The usefulness of the unit cell—furnishing the critical parameters necessitated by the macroscopic plasticity model—is realized through probing the microstructure in the *current* configuration. This probing imposes selected components from $\boldsymbol{\sigma}$ and $\Delta\boldsymbol{\epsilon}$ onto the boundary of the unit cell domain. As shown in Fig. 9, the unit cell is invoked at the end of the current load step $n+1$. After the probing is completed, the resulting configuration of the microstructure is recorded, which will be used as the starting configuration, or the current configuration, for the next unit cell computation. More details about the multiscale procedure and the unit cell computation are given in [18] and are outside the scope of this paper.

The basic PIVs in the D–P model are realized by invoking their physical significance, i.e.,

$$\alpha^{\mathrm{mic}} = -\frac{q^{\mathrm{mic}}}{p^{\mathrm{mic}}}, \qquad \beta^{\mathrm{mic}} = \frac{\Delta\epsilon_{\mathrm{v}}^{\mathrm{mic}}}{\Delta\epsilon_{\mathrm{s}}^{\mathrm{mic}}}, \tag{5.1}$$

where the superscript 'mic' signifies that the quantity is computed from the micromechanical model as a means to distinguish it from its continuum counterpart. The micromechanical variables are then passed as approximations to the continuum plastic internal variables, i.e., $\alpha \approx \alpha^{\mathrm{mic}}$ and $\beta \approx \beta^{\mathrm{mic}}$. In the next section, explanation is given in terms of how to compute the stress and strain in a micromechanical model.

5.2. A representative example

To demonstrate the effectiveness of the semi-implicit algorithm in incorporating nonsmooth micromechanical response into the multiscale scheme, we present the results of an axisymmetric compression computation on a granular assembly. We use DEM as the micromechanical model. To extract the stress tensor, equilibrium conditions for a particulate system can be invoked, yielding [21,44],

$$\bar{\boldsymbol{\sigma}} = \frac{1}{V}\sum_{c=1}^{N_c} \boldsymbol{l}^c \otimes \boldsymbol{d}^c, \tag{5.2}$$

where $\boldsymbol{l}^c$ represents the contact force at contact point c, $\boldsymbol{d}^c$ denotes the distance vector connecting the two neighboring particles, N_c is the total number of contacts in the particle assembly and V denotes the volume of the assembly, i.e., the volume of the unit cell domain associated with a specific Gauss point. To compute a homogenized strain tensor, the domain of the DEM-based unit cell can be partitioned into a series of polygonal subdomains, with the corners of each polygon being the centers of participating particles [45]. These polygons are deformed as the particle centers move, and the methods for computing these deformations are given in [46,47]. Consequently, a homogenized strain tensor can be obtained by averaging these polygon-based deformations over the entire domain of the unit cell.

At the continuum level, the sample domain is discretized using one 8-node isoparametric 'brick' element. A single unit cell is used to contain the cubic assembly of 1800 polydisperse spherical particles, shown in Fig. 11. Initially, the assembly was isotropically compressed to $p_0 = 5500$ kPa, with the initial configuration depicted in Fig. 11. The mixed boundary conditions of the unit cell include vertical strain control and horizontal stress control, consistent with the boundary conditions imposed on the finite element. A vertical strain increment $\Delta\epsilon_1 = 0.4\%$ was prescribed on the finite element. Putting the DEM model aside, the multiscale scheme involves only two parameters: $E = 5 \times 10^5$ kPa and $\nu = 0.25$. For comparison purposes, a direct numerical simulation

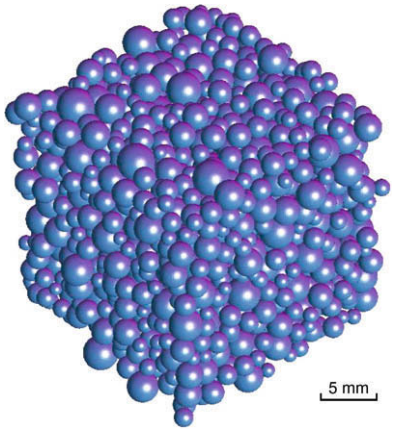

Fig. 11. Initial configuration of the DEM-based unit cell.

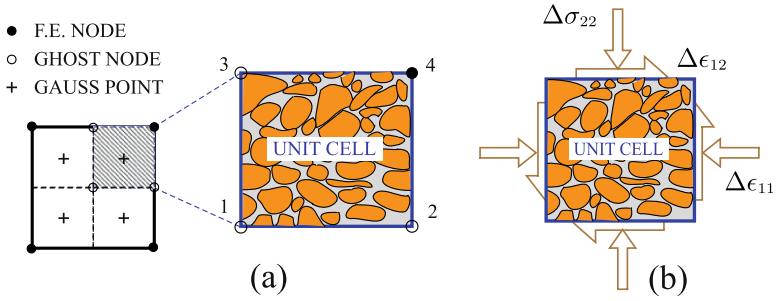

Fig. 10. Unit cell computation: (a) domain and (b) mixed boundary condition.

(DNS) was performed on the same DEM assembly, with identical initial state and identical loading mode. The DNS results are regarded as the 'exact' solution against which the accuracy and performance of the multiscale scheme is evaluated.

Fig. 12 shows the critical parameters (α^{mic} and β^{mic}) obtained from unit cell computation and the resulting friction resistance calculated using the multiscale method, i.e., $-q/p$. Fig. 12b reports the evolution of the micromechanically-based dilatancy β^{mic}, which is later passed onto the macroscopic plasticity model. It is clear that the micromechanical relations for both parameters are nonsmooth, especially in the post-peak, finite deformation regime. These nonsmooth evolutions of α^{mic} and β^{mic} are recast into the semi-implicit return mapping algorithm presented herein as nonsmooth evolution laws for the plastic internal variables α and β. However, these evolutions of the PIVs are not empirical and are rather extracted on-the-fly from the actual microstructure. As shown in Fig. 12a, the semi-implicit return mapping is able to reproduce the nonsmooth evolution of the frictional resistance parameter effectively and accurately.

Remark 4. In this paper, we use infinitesimal elastoplasticity as an example to demonstrate the effectiveness of the proposed algorithm. Extension to finite deformation plasticity is straightforward and will not incur any substantial change in the algorithm. This has been done before in the context of implicit return mapping algorithms (see [32,48]). We recognize the inaccuracy of the small deformation theory in representing the large deformations shown in the previous examples. However, these examples are not shown to capture the physics of deformation per se but to demonstrate the effectiveness of the semi-implicit return mapping algorithm.

Fig. 13 shows results obtained from the multiscale computation compared with those from the DNS. The accuracy of the multiscale method is measured here solely based on how closely it can reproduce the DNS results (verification). It can be seen that both the stress–strain response and the volumetric deformations are captured accurately by the multiscale model. This is remarkable in many levels, but most importantly due to the few parameters necessitated for the multiscale computation. The two elastic parameters are calibrated based on the initial response from the DNS and held constant for the duration of the simulation. Subsequently, the only parameters necessitated by the model are the frictional resistance and the dilatancy, which are allowed to evolve and are directly extracted from the micromechanics. It is remarkable that such a simple model can capture the material response so closely. Finally, Fig. 14 shows the global convergence rates for several different strain levels, highlighting the optimal convergence rate displayed by the algorithm. These results are very promising as they may open the door to more physics-based constitutive models to capture the mechanical behavior of granular media, without resorting to phenomenological evolution laws.

Remark 5. There is a noticeable shift in the responses obtained from the multiscale computation relative to the DNS. This finite gap occurs at the transition from pure elasticity to elastoplasticity and can be reduced by decreasing the time step. The shift is due to the semi-implicit return mapping freezing of the plastic internal variables involved in the multiscale computation.

Remark 6. The unit cell, representing the granular assembly, requires a number of parameters to describe the micromechanical response accurately. For the DEM model, these parameters include particle geometry, grain stiffness, intergranular friction, etc. These parameters substantially determine how accurately the micromechanical model captures the true material behavior, which, however, is not the main focus of this paper. The goal of the

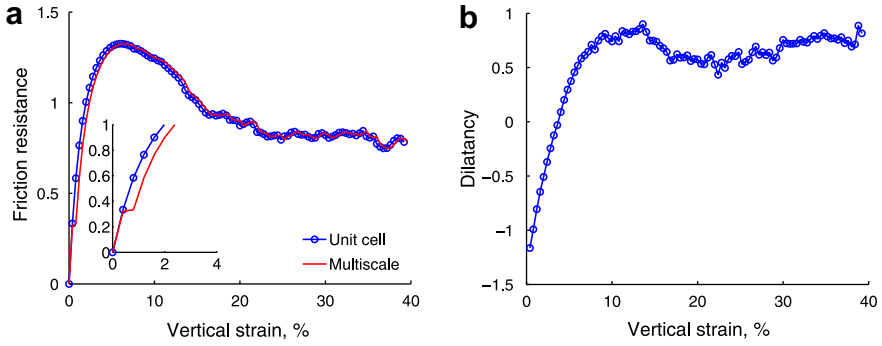

Fig. 12. Nonsmooth evolution of the critical parameters: (a) friction resistance obtained from unit cell vs. $-q/p$ computed by multiscale model and (b) dilatancy parameter obtained from unit cell.

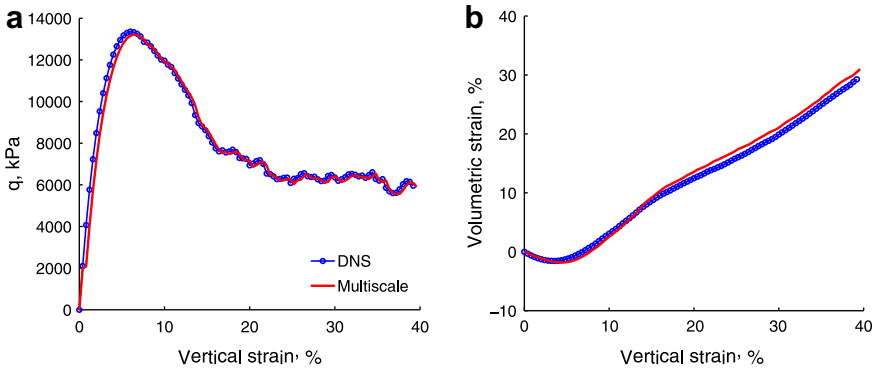

Fig. 13. Comparison of multiscale and DNS results: (a) stress response and (b) strain response.

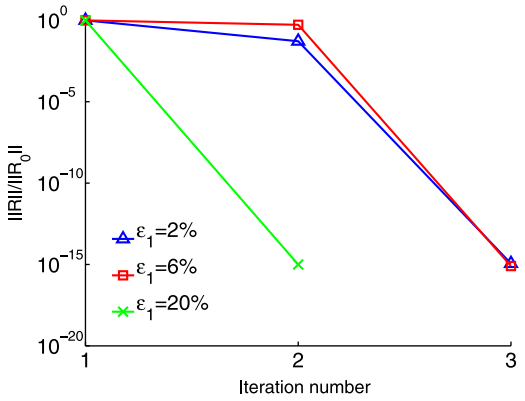

Fig. 14. Convergence profiles at the finite element level for the multiscale simulation.

multiscale scheme is to faithfully reproduce the response of the underlying micromechanical model at the continuum scheme (whatever that micromechanical model is). Hence, the multiscale method provides a bridge from the microscale to the macroscale but it does not provide a micromechanical model. However, it is our belief that this multiscale technique will allow further development of accurate and physics-based micromechanical models in the near future.

Remark 7. There are two key items related to the success of the multiscale technique. The first one is the appropriate selection of the so-called critical parameters—those parameters that are passed back to the macroscopic model. How to select these parameters is key. In the case of granular materials under slow flow (quasi-static deformation) it appears as though the frictional resistance and the dilatancy are sufficient to describe the bulk of the material response. Hence, many models that encapsulate these mechanisms can be used in the multiscale framework. This has been demonstrated elsewhere [18]. The second crucial item is the appropriate selection of the size of the unit cell. In this work, we have not invoked any theoretical basis for the selection of the size, but rather have based our determination on the concept of the unit cell (and RVE for that matter), that it is the minimum size element where high oscillations in continuum properties can be filtered out.

6. Closure

We have presented a semi-implicit return mapping algorithm for integration of the stress response in elastoplastic models with nonsmooth (C_0) evolution laws. The algorithm owes its versatility to the notion of freezing the plastic internal variables and *a posteriori* update of the PIVs. We have demonstrated that the semi-implicit algorithm displays some crucial qualities including good accuracy, stability, and the ability to calculate consistent tangent operators in closed-form, which result in global quadratic convergence. The simple algorithm was verified by way of numerical examples using empirically-based C_0 evolution laws as well as micromechanically-based evolutions of the critical variables. In both instances, it was demonstrated that the semi-implicit algorithm can handle non-smooth evolutions accurately and efficiently. These features make the method promising and computationally appealing.

Acknowledgements

Support for this work was provided in part by NSF Grant Number CMMI-0726908 and AFOSR Grant Number FA9550-08-1-1092 to Northwestern University. This support is gratefully acknowledged. The DEM model used in the multiscale computation is a modification of *Oval*, an open source GNU software that was developed by Prof. Matthew R. Kuhn from the University of Portland. The paper benefited substantially from the suggestions of three anonymous reviewers; their expert opinion is greatly appreciated.

References

[1] R. Hill, The Mathematical Theory of Plasticity, Oxford University Press, New York, NY, 1950.
[2] W.T. Koiter, General theorems for elastic–plastic solids, in: I.N. Sneddon, R. Hill (Eds.), Progress in Solid Mechanics, North-Holland, Amsterdam, 1960, pp. 165–221.
[3] P.V. Lade, Elastoplastic stress–strain theory for cohesionless soil with curved yield surfaces, Int. J. Solids Struct. 13 (1977) 1019–1035.
[4] T. Schanz, P.A. Vermeer, P.G. Boninier, The hardening soil model: formulation and verification, in: R.B.J. Brinkgreve (Ed.), Beyond 2000 in Computational Geotechnics – 10 years of Plaxis International, Balkema, Rotterdam, 1999, pp. 281–296.
[5] P.V. Lade, M.K. Kim, Single hardening constitutive model for frictional materials II. Yield criterion and plastic work contours, Comput. Geotech. 6 (1988) 13–29.
[6] P.V. Lade, K.P. Jakobsen, Incrementalization of a single hardening constitutive model for frictional materials, Int. J. Numer. Anal. Methods Geomech. 26 (2002) 647–659.
[7] I.M. Smith, D.V. Griffith, Programming the Finite Element Method, John Wiley & Sons Ltd., Chichester, UK, 1982.
[8] F.L. DiMaggio, I.S. Sandler, Material model for granular soils, J. Engrg. Mech. Div.-ASCE 97 (1971) 935–950.
[9] H. Matsuoka, T. Nakai, A new failure criterion for soils in three-dimensional stresses, in: Conference on Deformation and Failure of Granular Materials, IUTAM, 1982, pp. 253–263.
[10] A.J. Whittle, M.J. Kavvadas, Formulation of MIT-E3 constitutive model for overconsolidated clays, J. Geotech. Engrg.-ASCE 120 (1994) 173–198.
[11] R.I. Borja, A. Aydin, Computational modeling of deformation bands in granular media. I. Geological and mathematical framework, Comput. Methods Appl. Mech. Engrg. 193 (2004) 2667–2698.
[12] J.E. Andrade, R.I. Borja, Capturing strain localization in dense sands with random density, Int. J. Numer. Methods Engrg. 67 (2006) 1531–1564.
[13] C.D. Foster, R.A. Regueiro, A.F. Fossum, R.I. Borja, Implicit numerical integration of a three-invariant, isotropic/kinematic hardening cap plasticity model for geomaterials, Comput. Methods Appl. Mech. Engrg. 194 (50–52) (2005) 5109–5138.
[14] E. Tadmor, M. Ortiz, R. Phillips, Quasicontinuum analysis of defects in solids, Philos. Mag. A 73 (1996) 1529–1563.
[15] F. Feyel, A multilevel finite element method (FE^2) to describe the response of highly non-linear structures using generalized continua, Comput. Methods Appl. Mech. Engrg. 192 (2003) 3233–3244.
[16] W.K. Liu, E.G. Karpov, H.S. Park, Nano Mechanics and Materials, John Wiley & Sons Ltd., Chichester, West Sussex, UK, 2006.
[17] T. Belytschko, S. Loehnert, J.H. Song, Multiscale aggregating discontinuities: a method for circumventing loss of material stability, Int. J. Numer. Methods Engrg. 73 (2008) 869–894.
[18] J.E. Andrade, X. Tu, Multiscale framework for behavior prediction in granular media, Mech. Mater., in press, doi:10.1016/j.mechmat.2008.12.005.
[19] R.I. Borja, J.R. Wren, Micromechanics of granular media, Part I: Generation of overall constitutive equation for assemblies of circular disks, Comput. Methods Appl. Mech. Engrg. 127 (1995) 13–36.
[20] C. Wellmann, C. Lillie, P. Wriggers, Homogenization of granular material modeled by a three-dimensional discrete element method, Comput. Geotech. 35 (2008) 394–405.
[21] X. Tu, J.E. Andrade, Criteria for static equilibrium in particulate mechanics computations, Int. J. Numer. Methods Engrg. 75 (2008) 1581–1606.
[22] P.W. Christensen, A nonsmooth newton method for elastoplastic problems, Comput. Methods Appl. Mech. Engrg. 191 (2002) 1189–1219.
[23] B. Moran, M. Ortiz, C.F. Shih, Formulation of implicit finite element methods for multiplicative finite deformation plasticity, Int. J. Numer. Methods Engrg. 29 (1990) 483–514.
[24] T. Belytschko, W.K. Liu, B. Moran, Nonlinear Finite Elements for Continua and Structures, John Wiley & Sons Ltd., West Sussex, UK, 2000.
[25] S.W. Sloan, A.J. Abbo, D. Sheng, Refined explicit integration of elastoplastic models with automatic error control, Engrg. Comput. 18 (2001) 121–154.
[26] J. Zhao, D. Sheng, M. Rouainia, S.W. Sloan, Explicit stress integration of complex soil models, Int. J. Numer. Anal. Methods Geomech. 29 (2005) 1209–1229.
[27] J.C. Simo, T.J.R. Hughes, Computational Inelasticity, Prentice-Hall, New York, 1998.
[28] C. Tamagnini, R. Castellanza, R. Nova, A generalized backward euler algorithm for the numerical integration of an isotropic hardening elastoplastic model for mechanical and chemical degradation of bonded geomaterials, Int. J. Numer. Anal. Methods Geomech. 26 (2002) 963–1004.
[29] R.I. Borja, K.M. Sama, P.F. Sanz, On the numerical integration of three-invariant elastoplastic constitutive models, Comput. Methods Appl. Mech. Engrg. 192 (2003) 1227–1258.

[30] C. Miehe, Numerical computation of algorithmic (consistent) tangent moduli in large-strain computational inelasticity, Comput. Methods Appl. Mech. Engrg. 134 (1996) 223–240.
[31] M. Ortiz, J.B. Martin, Symmetry-preserving return mapping algorithms and incrementally extremal paths – a unification of concepts, Int. J. Numer. Methods Engrg. 8 (1989) 1839–1853.
[32] R.I. Borja, J.E. Andrade, Critical state plasticity, Part VI: Meso-scale finite element simulation of strain localization in discrete granular materials, Comput. Methods Appl. Mech. Engrg. 195 (2006) 5115–5140.
[33] S.W. Sloan, Substepping schemes for the numerical integration of elasto-plastic stress–strain relations, Int. J. Numer. Methods Engrg. 24 (1987) 893–911.
[34] K.P. Jakobsen, P.V. Lade, Implementation algorithm for a single hardening constitutive model for frictional materials, Int. J. Numer. Anal. Methods Geomech. 26 (2002) 661–681.
[35] A. Pérez-Foguet, A. Rodríguez-Ferran, A. Huerta, Consistent tangent matrices for substepping schemes, Comput. Methods Appl. Mech. Engrg. 190 (2001) 4627–4647.
[36] A. Pérez-Foguet, A. Rodríguez-Ferran, A. Huerta, Numerical differentiation for local and global tangent operators in computational plasticity, Comput. Methods Appl. Mech. Engrg. 189 (2000) 277–296.
[37] D.C. Drucker, W. Prager, Soil mechanics and plastic analysis or limit design, Quart. Appl. Math. 10 (1952) 157–165.
[38] O. Reynolds, On the dilatancy of media composed of rigid particles in contact, Philos. Mag. J. Sci. 20 (1885) 468–481.
[39] P.W. Rowe, The stress-dilatancy relation for static equilibrium of an assembly of particles in contact, Proc. Royal Soc. London, Ser. A 269 (1962) 500–527.
[40] D. Muir-Wood, Soil Behaviour and Critical State Soil Mechanics, Cambridge University Press, Cambridge, UK, 1990.
[41] P.A. Cundall, O.D.L. Strack, A discrete numerical model for granular assemblies, Géotechnique 29 (1979) 47–65.
[42] P.A. Cundall, A discontinuous future for numerical modelling in geomechanics? Geotech. Engrg., ICE 149 (2001) 41–47.
[43] Z. Hashin, Analysis of composite materials – a survey, J. Appl. Mech. 50 (1983) 481–505.
[44] J. Christoffersen, M.M. Mehrabadi, S. Nemat-Nasser, A micromechanical description of granular material behavior, J. Appl. Mech. 48 (1981) 339–344.
[45] M. Satake, A discrete-mechanical approach to granular materials, Int. J. Engrg. Sci. 30 (1992) 1525–1533.
[46] K. Bagi, Stress and strain in granular assemblies, Mech. Mater. 22 (1996) 165–177.
[47] M.R. Kuhn, Structured deformation in granular materials, Mech. Mater. 31 (1999) 407–429.
[48] J.C. Simo, Algorithms for static and dynamic multiplicative plasticity that preserve the classical return mapping schemes of the infinitesimal theory, Comput. Methods Appl. Mech. Engrg. 99 (1992) 61–112.