Stress update algorithm for elastoplastic models with nonconvex yield surfaces

Dorival M. Pedroso*†, Daichao Sheng and Scott W. Sloan

Centre for Geotechnical and Materials Modelling, The University of Newcastle, University Drive, Callaghan, NSW 2308, Australia

SUMMARY

A stress update algorithm for elastoplastic models with nonconvex yield surfaces is presented. An explicit algorithm based on the Runge–Kutta embedded method of second-order accuracy is developed. The crossing of the yield surface is properly taken into account by means of a robust intersection-finding algorithm. This algorithm is based on a simple multiple-root-finding technique, which requires the yield function, evaluated along a given secant stress path, to be continuously differentiable to the second order. The accuracy of the intersection-finding algorithm is illustrated through examples using simple yield functions similar to the ones adopted in models for unsaturated soils and the modified Cam clay model yield surface with a nonconvex Argyris et al. failure criterion. Isoerror surfaces and finite element simulations are used to investigate the accuracy and efficiency of the stress update algorithm. It is observed that although the algorithm for nonconvex surfaces is slower than its equivalent for convex surfaces, the accuracy can be controlled locally by means of specified tolerances. Copyright © 2008 John Wiley & Sons, Ltd.

Received 7 October 2007; Revised 21 February 2008; Accepted 23 May 2008

KEY WORDS: stress integration; nonconvex yield surface; unsaturated soils; Runge–Kutta method; differential algebraic system; explicit schemes; multiple roots

1. INTRODUCTION

The solution of the initial boundary value problem (IBVP) for a continuum can be achieved incrementally after the discretization of the problem in space and time (see, for example, [1]). This kind of IBVP generates a non-linear differential algebraic system (DAS) (see, for example, [2, 3]) where each point is assigned an incremental constitutive law, according to continuum mechanics theory. This theory establishes basic concepts, such as stress and strain, that allow for the study of the internal response after the application of external boundary conditions [4]. Constitutive equations

*Correspondence to: Dorival M. Pedroso, Centre for Geotechnical and Materials Modelling, The University of Newcastle, University Drive, Callaghan, NSW 2308, Australia.

†E-mail: dorival.pedroso@gmail.com

Copyright © 2008 John Wiley & Sons, Ltd.
based on infinitesimal elastoplasticity often require a numerical scheme for time integration due to their relative complexity (see, for example, [5, 6]). Often analytical solutions cannot be easily deduced. The solution of these equations in the context of finite elements is commonly known as the stress update or stress integration [7, 8].

In the context of finite element solutions to the global DAS, the displacements are the primary variables determined iteratively from the balance laws and hence the total strain increments are considered as input data to the constitutive equations. For unsaturated soil models, the suction increment can also be considered as input data [9–11]. In this sense, the solution is locally strain–suction driven, even if it is load controlled at the global level. Similarly, rate-independent elastoplastic models are described by a set of local differential algebraic equations (DAEs). These equations can be solved accurately by Runge–Kutta (RK) methods with embedded error estimates that can solve a DAE using automatically subdivided steps (see, for example, [12–14]). A comprehensive treatment of the solution of common elastoplastic equations for soils using RK embedded schemes is given in [9, 15–17]. The first reference also considers models for unsaturated soils.

Two classes of RK methods arise: implicit and explicit. Implicit methods use gradients estimated at advanced state positions and solve a DAS by iterations. On the other hand, explicit methods use only gradients evaluated at initial positions. Mixed semi-implicit or semi-explicit methods are also possible by considering gradients calculated at different state positions. Algorithms based on the RK method can be devised with different orders of local approximation, including higher-order methods. Other implicit and explicit methods can be elaborated with higher orders; however, implicit methods of higher order [18] are rare in the literature where the first-order accurate backward Euler (BE) is commonly adopted (see, for example, [1, 7, 8, 19, 20]).

Algorithms for numerical integration of elastoplastic DAEs must deal with the bi-modal characteristic of these equations since the current tangent modulus is selected according to a loading–unloading analysis. This characteristic leads to a discontinuity of the DAE at the transition from elastic to elastoplastic behaviour and vice versa. Moreover, for models with nonconvex yield surfaces, the decision on the loading–unloading regime cannot be made solely by using the sign of the inner product between the tensor normal to the yield surface and the secant or trial stress increment (see the product $\Delta \sigma_{\text{trial}} : n$ illustrated in Figure 1). Therefore, an alternative algorithm must be devised.

Nonconvex yield surfaces are inevitable in models for unsaturated soils if the isotropic internal variable that measures the size of the yield surface is defined in such a way that it increases with

---

Figure 1. Convex and nonconvex yield surfaces, outward normal vectors $n$, and stress path $\Delta \sigma_{\text{trial}}$.
increasing suction [21–24]. This definition is crucial for the consideration of volume collapse due to a decrease in suction. Figure 2 shows the nonconvexity of the yield surfaces in the stress–suction space. The nonconvexity always exists at the transition between saturated and unsaturated states, irrespective of the stress variables adopted. In Figure 2, \( p' = p^\text{tot} - u_w \) is the effective mean stress, \( \bar{p} = p^\text{tot} - u_a \) is the net stress, and \( s = u_a - u_w \) is the matric suction. In addition, \( p^\text{tot} \) is the total mean stress, \( u_a \) is the air pore pressure, \( u_w \) is the water pore pressure, and \( s_{sa} \) is the saturation suction.

All stresses here are effective or net when no indication is made. The total stress will always be represented with a ‘tot’ superscript. Moreover, compressive stresses are taken as positive, following soil mechanics convention, and the strains are considered to be small.

A nonconvex yield surface may also arise when certain failure criteria for granular materials are adopted. For example, as discussed in [25], the Argyris et al. failure criterion [26] becomes nonconvex for friction angles greater than 22°. On the other hand, the criterion proposed in [25] will be nonconvex for friction angles greater than 48.6°. The stress update algorithm for equations with nonconvex yield surfaces must be able to determine at least the first intersection with the yield surface, for any secant (trial) stress path. The algorithm presented here finds all intersections by means of a multiple-root solver and then selects the first intersection point.

As explicit schemes do not require evaluations of the gradients at advanced positions, the procedure that searches for intersections can be simply incorporated into an explicit RK-embedded scheme of any order. In first-order implicit schemes, such as the ‘return mapping’ scheme, the difficulty is twofold (1) the nonconvexity may be overlooked and (2) the return direction may not be unique. Because implicit schemes usually do not compute the intersection points with the current yield surface, an elastic trial stress path starting from and ending inside the yield surface will be considered elastic only. In order to capture the underestimated plastic strain, the load increments must then be kept very small [27]. For certain nonconvex yield surfaces, such as those for unsaturated soils, the nonconvexity occurs only on the suction axis and the suction is held fixed during the plastic corrector phase. In this case, the return direction is unique [27]. However, an implicit scheme that incorporates an intersection-finding algorithm will work similarly to an explicit scheme that proceeds in an incremental manner, seeking intersections and then reducing the substep as appropriate to avoid crossing the yield surface until the next substep.

Here, we extend the explicit schemes presented by Sloan [15], Sloan et al. [17], and Sheng et al. [9] to the case where a nonconvex yield surface is adopted in an elastoplastic model. The
The resulting scheme is based on the RK-embedded method of second order and has the convenient feature of automatic substep and error control. The scheme is presented in a general form that can be applied to the solution of equations for both saturated and unsaturated soils.

The efficiency and accuracy of the new scheme are studied in terms of isoerror maps and finite element analyses. Following Simo and Hughes [8], the isoerror maps are found from the local solution of the stress–strain relations in which a set of strain increments are applied to the stress update algorithm. A simple 3D problem is solved by the finite element method (FEM) and used to check the efficiency of the integration for a model with a nonconvex yield surface in comparison with the modified Cam clay model. Additional results regarding the efficiency and accuracy of the RK schemes for the solution of soil models are available in the literature [6, 9, 10, 15, 17, 28, 29].

Clearly, due to the nonconvexity considerations, the speed of the simulations will depend on the performance of the multiple-intersection algorithm, which requires higher-order derivatives. It is expected that it will be slower than the simulations with conventional elastoplastic models. Based on efficiency concerns only, it is highly recommended that nonconvex yield surfaces in elastoplastic models are avoided where possible.

2. SOLUTION FOR THE INTERSECTION POINT

For given strain and suction increments, the current stress state and internal variables must be updated following a selected constitutive law. This update is generally carried out using numerical schemes as analytical solutions are usually unavailable. For elastoplastic models, the intersection of the stress path with the yield surface must be determined in order to decide which tangent modulus is appropriate during the integration. Before the intersection point is reached, the elastic modulus is used, while after this point the elastoplastic modulus and the hardening law are considered.

This bi-modal characteristic makes elastoplastic equations difficult to integrate when nonconvex yield surfaces are adopted; for example, a variety of possible intersections for a nonconvex surface are illustrated in Figure 3. The most complicated situation occurs when the yield surface is crossed three times. However, it is not possible to know a priori how many times the yield surface will be crossed.

![Figure 3. Generic nonconvex yield surfaces and intersections for different increments.](image-url)
crossed, because the size (and position) of the yield surface may change after the first intersection. Therefore, the numerical algorithm must integrate from the current state to the first intersection point and, afterwards, check again for further intersections. Thus, for nonconvex yield surfaces, possible intersection points must be checked for each increment along the stress path.

Considering a general elastoplastic model, including the effect of suction, a secant (trial) stress increment can be defined in order to determine whether the yield surface is crossed or not:

\[ \Delta \sigma^{\text{trial}} = D^e : \Delta \varepsilon + W^e \Delta s \]

in which \( D^e \) is the fourth-order elastic stiffness tensor and \( W^e \), necessary only for models for unsaturated soils, is a second-order tensor (see, for example, the equations presented in [9]). For saturated soil models, the term \( W^e \Delta s \) depends on the stress variables selected. If the effective stress is adopted, the term \( W^e \Delta s \) vanishes and can be disregarded. On the other hand, if the net stress is used this term will be \( W^e \Delta s = -I \Delta u_w \) with \( I \) being the second-order identity tensor and \( u_w \) being the pore water pressure.

In Equation (1), \( \Delta \varepsilon \) is the known strain increment, which is found from the strain–displacement relations prior to the computation of the residuals (i.e. the difference between the external and internal forces). For unsaturated soils, the increment in suction \( \Delta s \) is also supplied to the stress update algorithm. If the elastic modulus is linear, i.e. it is independent of the stresses, suction, and internal variables, then it is trivial to compute the elastic trial increment. Otherwise, for some non-linear relations, a secant modulus may be adopted.

Finding the intersection between the elastic trial stress increment and the current yield surface reduces to the problem of finding the multiple roots of a nonlinear equation. Here the yield function is evaluated along the secant stress path and is indicated as \( f_z(\varepsilon) = 0 \). The parameter \( \varepsilon \) lies in the interval \([0, 1]\) and represents a fraction of the given strain increment. The function used to compute the roots \( \varepsilon \) is given by

\[ f_z(\varepsilon) = f(\sigma_z, s_z, z_k) \]

where \( f(\sigma_z, s_z, z_k) \) is the yield function, \( z_k \) indicates a set of internal variables, and the intermediate stress–suction states \( \sigma_z \) and \( s_z \) are calculated according to

\[ \sigma_z = \sigma_{\text{current}} + \varepsilon \Delta \sigma^{\text{trial}} \quad \text{and} \quad s_z = s_{\text{current}} + \varepsilon \Delta s \]

Equation (2) the internal variables \( z_k \) must be continuously differentiable to the second order for values of \( \varepsilon \) from \( a \) to \( b \). In Equation (4), \( g_1(\varepsilon) \) and \( h_2(\varepsilon) \) stand for the first and second derivatives of the function \( f_z \) with respect to \( \varepsilon \), respectively, and \( \gamma \) is a small positive constant that does not affect the result computed with the KP formula [30]. The first and second derivatives of \( f_z \) with respect to \( \varepsilon \) can
be directly determined as follows:

\[
g_z(x) = \frac{\partial f_z}{\partial \mathbf{\sigma}^z} : \Delta \mathbf{\sigma}^\text{trial} + \frac{\partial f}{\partial \mathbf{\sigma}_z} : \Delta \mathbf{\sigma}_z + \frac{\partial f_{sz}}{\partial \mathbf{\sigma}_z} : \Delta s^z (5)
\]

\[
h_z(x) = \frac{\partial^2 f_z}{\partial \mathbf{\sigma}^z \partial \mathbf{\sigma}_z} = \Delta \mathbf{\sigma}^\text{trial} : \frac{\partial^2 f}{\partial \mathbf{\sigma}^z \partial \mathbf{\sigma}_z} \bigg|_z + \frac{\partial f_{sz}}{\partial \mathbf{\sigma}_z} : \Delta s^z + \frac{\partial^2 f_{sz}}{\partial \mathbf{\sigma}^z} : \Delta s^z (6)
\]

In the case where the suction state variable is disregarded, i.e. for saturated soils, these derivatives are simply given by

\[
g_z(x) = \frac{\partial f}{\partial \mathbf{\sigma}} : \Delta \mathbf{\sigma}^\text{trial} \quad \text{and} \quad h_z(x) = \Delta \mathbf{\sigma}^\text{trial} : \frac{\partial^2 f}{\partial \mathbf{\sigma}^z} \bigg|_z (7)
\]

The number of roots calculated using Equation (4) is used to divide the interval of \( x \) into subintervals until each subinterval contains at most one root. First, \( N \) is computed for the interval \([a, b]\). If \( N \) is larger than one, the interval \([a, b]\) is divided into two equal subintervals, \([a, (a+b)/2]\) and \([(a+b)/2, b]\). The number of roots for each subinterval is then computed and any subinterval that contains more than one root is further divided into two equal sub-subintervals. This process continues until each subinterval contains at most one root. As shown in [30], the usage of equal-size intervals (equiprobable parts) is not much worse than an algorithm that considers the statistical distribution of the roots inside \([a, b]\).

Once the roots are bracketed, the solution for each root can be found by using well-known numerical methods such as the Newton–Raphson algorithm. It should be noted that the Newton–Raphson method, although fast, may not converge in some circumstances because it does not constrain the solution to lie within specified bounds. Therefore, more advanced methods can be used. For example, the Pegasus method used in [17] is very robust, simple, and competitively fast. Brent’s method [31] also provides another attractive alternative. This method does not use any derivatives, does not require initial guesses, and guarantees convergence as long as the values of the function are computable within a given region containing a root. These characteristics of Brent’s method are due to the combination of the bisection method, the secant method, and inverse quadratic interpolation. Therefore, it has the reliability of the bisection method as well as the efficiency of the less reliable secant method and inverse quadratic interpolation. A detailed description of Brent’s method is given in [32].

The evaluation of the integral in Equation (4) with the KP formula is generally nontrivial. Some numerical integration or quadrature has to be used, such as the Gauss–Legendre method (see, for example, [32, 33]). In addition, for highly non-linear yield functions, an adaptive integration scheme may be beneficial. For the numerical examples presented in this paper, the adaptive integration routine QAGS of Reference [34], which is openly available in the GNU Scientific Library (GSL, www.gnu.org/software/gsl), is used. These routines are based on the QUADPACK Fortran library available at www.netlib.org and can numerically integrate functions with singularities efficiently.

Figure 4 presents the algorithm that locates and computes all simple roots of \( f_z(x)=0 \) for a given interval of \( x \). In the case of elastoplastic models with nonconvex yield surfaces studied here, the interval is \([0, 1]\). For the algorithm given in Figure 4, the functions Integrate and FindRoot are used to perform numerical integration and numerical solution of non-linear equations, respectively. Although all the numerical examples in this paper are solved using the QAGS quadrature for
STRESS UPDATE ELASTOPLASTIC MODELS NONCONVEX YIELD SURFACES 2035

Figure 4. Algorithm for a multiple-root-finding solver.

Integrate and Brent’s method for FindRoot, other methods could be used in their place. The kernel of the algorithm is the recursive function Roots that proceeds by dividing the interval into smaller parts, after the determination of the number of roots. The exit points for the Roots function are (i) when no root is present inside an interval and (ii) when only one root exists in an interval and the root is found by the function FindRoot. The root found is then stored in the array $z_k$ with all other roots. The algorithm is initialized by setting the number of roots $N$ to zero and providing the range $[A, B]$.

3. LOADING–UNLOADING DECISION

One of the difficulties with nonconvex yield surfaces arises when deciding whether the stress path is directed towards the inside (unloading) or outside (loading) of the current yield surface. The inner product between the tensor normal to the yield surface and the secant or trial stress increment, as shown by the product $u \cdot n$ in Figure 1, is no longer sufficient to make this decision. In addition, in computer arithmetic, it is unusual to obtain a yield function value exactly equal to zero and a
small tolerance must be considered instead. This situation is illustrated in Figures 5 and 6. The
unavoidable tolerance for the yield function value introduces additional difficulties in handling the
loading–unloading decision. In this paper, a new procedure for the loading–unloading decision is
presented.

The loading–unloading decision for nonconvex yield surfaces is based on the following:

(i) The intersection algorithm must always be performed in order to confirm that all initial
states (inside or on the yield surface) and all possible paths (crossing, or pointing to the
exterior or to interior of the yield surface) are considered (see Figure 3);

(ii) only the sign of the yield function $f(x)$ is used to detect the loading/unloading condition;

(iii) a scheme to correct the yield surface drift must be adopted just after the stress update.

One question that arises is: where should the sign of $f(x)$ be checked for every stress path? This
location, denoted here by $x_{\text{checkpoint}}$, should be somewhere between the current stress state and the

---

Copyright © 2008 John Wiley & Sons, Ltd. 
DOI: 10.1002/nme
The first intersection must exclude a possible intersection within the tolerance \( \pm \xi_{\text{TOL}} \) as illustrated in Figure 6. With these considerations, \( \xi_{\text{checkpoint}} \) is defined as

\[
\xi_{\text{checkpoint}} = \begin{cases} 
\xi_{\text{TOL}} + \frac{\xi_{\text{INT}}}{2} & \text{if there are intersections} \\
\xi_{\text{TOL}} & \text{otherwise}
\end{cases}
\]  

(8)

Once the checkpoint \( \xi_{\text{checkpoint}} \) is properly identified, \( f_{\text{checkpoint}} > 0 \) will indicate loading and \( f_{\text{checkpoint}} \leq 0 \), unloading.

Figure 6 presents a general situation where an exaggerated tolerance (FTOL) of the yield function is shown to visualize the drift. It is possible to relate this tolerance FTOL to the tolerance \( \xi_{\text{TOL}} \) along the secant trial path by means of

\[
\text{FTOL} \leq |g_z(0)\xi_{\text{TOL}}|
\]  

(9)

in which \( g_z(0) \) is the first derivative of the yield function \( f_z \) with respect to \( z \) and is evaluated at the initial position. Thus, the tolerance FTOL will be computed locally at each position on the yield surface. Hence, this constraint allows the loading–unloading decision to be made based only on the sign of \( f_z \) near the initial yield surface.

The concept underlying this loading/unloading decision algorithm can be further understood with the aid of Figure 6, where a nonconvex surface and several initial stress states are shown. The states are on or just outside the yield surface with \( f_z(0) = 0 \). When the secant path is directed to the inside of the surface, as in the cases d, e, and f, the checkpoints determined by Equation (8) will be inside the current yield surface (\( f_{\text{checkpoint}} \leq 0 \)) and thus the elastic modulus (unloading) will be selected. On the other hand, for the cases a, b and c illustrated in Figure 6, the checkpoints are outside the current yield surface and thus the elastoplastic modulus (loading) will be selected. The checkpoints for cases g and h are inside the yield surface and hence the elastic modulus will be selected until the intersections. Finally, the checkpoints for cases i, j, and l are outside the yield surface and thus the elastoplastic modulus (loading) will be selected. Therefore, the loading/unloading algorithm handles the bi-modal characteristic of elastoplasticity for the case of a convex surface without the existence of drift.

An essential requirement to make the loading–unloading algorithm work properly is the usage of a drift correction scheme for every update of the stress state. Any correction scheme will suffice since the loading–unloading decision does not depend on the stress–strain response, but only on the current values of \( \xi_{\text{TOL}} \), i.e. the size of the drift. Thus, the drift correction function must guarantee that the yield surface error is not (locally) larger than \( |g_z\xi_{\text{TOL}}| \). In this paper, the drift correction known as the ‘consistent’ method [17, 35] is employed.

4. STRESS UPDATE

The stress update algorithm is based on an explicit RK method embedded with a local error estimator. This error measure is used to develop an automatic substepping scheme resulting in a very convenient and general numerical integrator for constitutive equations. The scheme is second-order accurate and includes the modified-Euler (ME) scheme as a special case. Owing to the consideration of nonconvex yield surfaces, the intersection-finding algorithm must be called for each substep.
The goal is to integrate the following incremental elastoplastic relationship:

$$\Delta \sigma = D^e : \Delta \epsilon - \Delta \Lambda D^e : \mathbf{r} + W^e \Delta s$$  \hspace{1cm} (10)$$
in which the total strain increment is linearly decomposed into elastic and plastic parts:

$$\Delta \epsilon = \Delta \epsilon^e + \Delta \epsilon^p$$  \hspace{1cm} (11)$$
and the plastic part is given by the flow rule

$$\Delta \epsilon^p = \Delta \Lambda \mathbf{r}$$  \hspace{1cm} (12)$$

In the above equation, \( \mathbf{r} \) is the plastic flow direction which can be determined according to a plastic potential \( P \) by means of

$$\mathbf{r} = \frac{\partial P}{\partial \sigma}$$  \hspace{1cm} (13)$$
or, for an associated flow rule, by means of

$$\mathbf{r} = \frac{\partial f}{\partial \sigma}$$  \hspace{1cm} (14)$$

The Lagrange multiplier \( \Delta \Lambda \) is obtained from the consistency condition \( \Delta f = 0 \), yielding

$$\Delta \Lambda = \frac{V : D^e : \Delta \epsilon + (V : W^e + S) \Delta s}{V : D^e : \mathbf{r} + h^p}$$  \hspace{1cm} (15)$$
in which \( h^p \) is a plastic coefficient calculated from

$$h^p = \sum_k y_k \mathcal{H}_k$$  \hspace{1cm} (16)$$

and \( V, S, \) and \( y_k \) are the derivatives of the yield function with respect to the stress tensor, suction, and internal variables, respectively, according to

$$V = \frac{\partial f}{\partial \sigma}, \quad S = \frac{\partial f}{\partial S} \quad \text{and} \quad y_k = \frac{\partial f}{\partial z_k}$$  \hspace{1cm} (17)$$

\( \mathcal{H}_k \) is a hardening modulus used to define the hardening of an elastoplastic model by means of the evolution equation for the internal variables \( z_k \). This evolution may be represented in a general form according to

$$\Delta z_k = \Delta \Lambda \mathcal{H}_k$$  \hspace{1cm} (18)$$

Again, the term containing \( \Delta s \) in Equation (15) may be dropped for saturated soil models.

Following the midpoint rule, the solution of Equation (10) is rewritten as

$$\Delta \sigma = D^e(\sigma^0, s^0) : \Delta \epsilon - \langle \Delta \Lambda(\sigma^0, s^0, z_k^0) D^e(\sigma^0, s^0) : \mathbf{r}(\sigma^0, s^0) \rangle + W^e(\sigma^0, s^0) \Delta s$$  \hspace{1cm} (19)$$
in which the term inside \( \langle \rangle \) is considered only for elastoplastic behaviour (loading) and \( \beta \) represents the position where the state variables are computed. Equation (19) recovers the forward-Euler
The BE scheme has the advantages of being unconditionally stable and allowing for the use of consistently linearized tangent moduli; however, it is still of first-order accuracy and may not converge for large substeps (see, for example, discussions in [6, 17, 28]).

An explicit second-order accurate algorithm can be devised starting with the first-order accurate FE stress update

$$\sigma^{FE} = \sigma^c + \Delta\sigma^1$$

where

$$\Delta\sigma^1 = D^c(\sigma^c, s^c) : \Delta\varepsilon - \Lambda(\Delta\lambda(\sigma^c, s^c, z_k^c)D^c(\sigma^c, s^c) : \epsilon(\sigma^c, s^c)) + W^c(\sigma^c, s^c)\Delta s$$

and the stiffness $D^c$ and $W^c$ are computed a second time at the state obtained from Equation (20). In the above equations, $\sigma^c$, $s^c$, and $z_k^c$ stand for the current stress, suction, and internal variables, respectively. The FE updates are then used to calculate an intermediate increment, denoted by $\Delta\sigma^2$ and defined according to

$$\Delta\sigma^2 = D^c(\sigma^{FE}, s^{FE}) : \Delta\varepsilon - \Lambda(\Lambda^L(\sigma^{FE}, s^{FE}, z_k^{FE})D^c(\sigma^{FE}, s^{FE}) : \epsilon(\sigma^{FE}, s^{FE})) + W^c(\sigma^{FE}, s^{FE})\Delta s$$

The second-order accurate update (ME) is thus obtained:

$$\sigma^{ME} = \sigma^c + \frac{1}{2}(\Delta\sigma^1 + \Delta\sigma^2)$$

This method has an embedded error estimate

$$\text{error} = \frac{\|\sigma^{ME} - \sigma^{FE}\|_2}{1 + \|\sigma^{ME}\|_2}$$

This local error estimate, well known in RK-embedded methods, is a powerful tool for devising algorithms with automatic subinconrementation. If the local error is smaller than a prescribed tolerance (STOL), the stress update is accepted, otherwise, the strain and suction increments are re-divided into smaller subincrements and the procedure is repeated, disregarding the recent updates.

A rational subinconrementation approach can be constructed by extrapolating the local error, following a method similar to that proposed by Richardson (see, for example, [13, 14]). This extrapolation is given by

$$\text{err}_i \approx C(\delta_i)^{p+1}$$

where $i$ is the current step, $C$ is a constant related to a specific system of equations (DAS), $\delta$ is the step size, and $p+1$ is the order of the higher-order update. If it is assumed that the next step is determined from

$$\delta_{i+1} = m\delta_i$$

where $m$ is a step-size multiplier, then the error in this next step can be approximated by

$$\text{err}_{i+1} \approx C(m\delta_i)^{p+1} = m^{p+1}C(\delta_i)^{p+1} = m^{p+1}\text{err}_i$$
To control the local error, the step-size multiplier can be defined as

\[ m \leq \left( \frac{\text{STOL}}{\text{err}_i} \right)^{1/(p+1)} \]  

(28)

for which the following relation must be satisfied:

\[ \text{err}_{i+1} \approx m^{p+1} \text{err}_i \leq \text{STOL} \]  

(29)

For the second-order accurate algorithm, Equation (28) is

\[ m \leq \left( \frac{\text{STOL}}{\text{err}_i} \right)^{1/2} \]  

(30)

in which the value \( p+1 = 2 \) was substituted as the order of the ME evaluation is two.

Since the hardening law can be non-linear and depend on the current state as well, the internal variables should be updated by the same algorithm and at the same time as the stresses. This suggests that the local error in the update of the internal variables should also be considered. This update must be done only when the elastoplastic behaviour is detected (loading). Therefore, a first estimate can be computed by

\[ z_{\text{FE}}^k = z_c^k + \Delta z_k^1 \]  

(31)

where according to Equation (18),

\[ \Delta z_k^1 = \mathcal{H}_k(\sigma^c, s^c, z_c^k) \Delta \Lambda(\sigma^c, s^c, z_c^k) \]  

(32)

and both \( \mathcal{H}_k \) and \( \Delta \Lambda \) are computed at the current state. The second-order accurate state is obtained as follows:

\[ z_{\text{ME}}^k = z_c^k + \frac{1}{2} (\Delta z_k^1 + \Delta z_k^2) \]  

(33)

in which

\[ \Delta z_k^2 = \mathcal{H}_k(\sigma_{\text{FE}}, s_{\text{FE}}, z_{\text{FE}}^k) \Delta \Lambda(\sigma_{\text{FE}}, s_{\text{FE}}, z_{\text{FE}}^k) \]  

(34)

Therefore, the local error during loading can be determined according to

\[ \text{error} = \max \left( \frac{\| \sigma_{\text{ME}} - \sigma_{\text{FE}} \|_2}{1 + \| \sigma_{\text{ME}} \|_2}, \frac{|z_{\text{ME}}^k - z_{\text{FE}}^k|}{1 + |z_{\text{ME}}^k|} \right) \]  

(35)

After a successful update, the drift correction must ensure that the current state lies inside the external surface tolerance (see Figure 6), according to the tolerance FTOL defined in Equation (9). This correction changes both the stresses and the internal variables. However, the strain and suction increments should remain unchanged. After a partial update (using a substep), if the ‘new’ yield function value is not smaller than FTOL, i.e. if

\[ f^{\text{new}}(\sigma^{\text{new}}, s^{\text{new}}, z_k^{\text{new}}) \geq \text{FTOL} \]  

(36)

then a correction must be done. In this case, the ‘correct’ yield function is obtained by considering a truncated Taylor series extrapolation as follows:

\[ f^{\text{correct}} = f^{\text{new}} + \nabla f \cdot \delta \sigma + S \delta s + y_k \delta z_k \]  

(37)
in which the derivatives $V, S$, and $y_k$ are evaluated at this recent new position given by $\sigma^{\text{new}}, s^{\text{new}},$ and $z_k^{\text{new}}$. Substituting $\delta \varepsilon = 0$ and $\delta s = 0$ in Equation (10)

$$\delta \sigma = - \delta \Lambda \mathbf{D}^e : \mathbf{r}$$

Additionally, considering the hardening law (Equation (18)):

$$\delta z_k = \delta \Lambda \mathbf{H}_k$$

in which the hardening modulus $\mathbf{H}_k$ must be computed for the states $\sigma^{\text{new}}, s^{\text{new}},$ and $z_k^{\text{new}}$. From Equations (37)–(39), the plastic multiplier $\delta \Lambda$ can be deduced by considering the requirement $f^{\text{correct}} = 0$. This gives

$$\delta \Lambda = \frac{f^{\text{new}}}{V : \mathbf{D}^e : \mathbf{r} - y_k \mathbf{H}_k}$$

and, hence, the correction of the stresses and internal variables as

$$\sigma^{\text{correct}} = \sigma^{\text{new}} + \delta \sigma \quad \text{and} \quad z_k^{\text{correct}} = z_k^{\text{new}} + \delta z_k$$

for which some iterations may be needed to ensure that $f^{\text{correct}} < \text{FTOL}$.

Figures 7–9 present in detail all the steps for the algorithm proposed. In these figures, the current state, given by $\sigma, s, z_k,$ and $\varepsilon$, and the increments $\Delta \varepsilon$ and $\Delta s$ are input. The increment $\Delta \sigma$ is output. The algorithm provides second-order accurate results, considers nonconvexity of the yield surface, and uses the loading–unloading scheme presented previously. Comments are included in these figures to explain the algorithm and make it easier to implement. References [17, 23, 28] give additional details such as the numeric constants needed for the substepping strategy.

5. EXAMPLES OF INTERSECTION FINDING

Some examples are provided in order to illustrate the intersection-finding algorithm. The first example is a simple polynomial with a nonconvex shape based on the cardioid curve. Another example is based on the Barcelona basic model (BBM) for unsaturated soils [21]. The last example uses the modified Cam clay model with a nonconvex yield surface in the deviatoric plane as defined by the Argyris et al. failure criterion [26].

5.1. Cardioid yield surface

For this example, a generic plane $x–y$ is considered for the definition of the yield function. $x$ and $y$ can be regarded as effective stress invariants. For given trial increments $\Delta x$ and $\Delta y$, the fictional yield function ($f_2$) is calculated as follows:

$$f_2(x) = f_2(x_2, y_2)$$

in which

$$x_2 = x_{\text{current}} + x \Delta x \quad \text{and} \quad y_2 = y_{\text{current}} + y \Delta y$$
**Figure 7.** Second-order explicit Runge–Kutta-based stress update algorithm with substepping for models with nonconvex yield surfaces.

Therefore, the first and second derivatives are simply

\[
g_x(x) = \frac{\partial f}{\partial x} \bigg|_x \Delta x + \frac{\partial f}{\partial y} \bigg|_x \Delta y
\]

\[
h_x(x) = \frac{\partial^2 f}{\partial x^2} \bigg|_x \Delta x^2 + \frac{\partial^2 f}{\partial x \partial y} \bigg|_x \Delta x \Delta y + \frac{\partial^2 f}{\partial y^2} \bigg|_x \Delta y^2
\]

This example is useful to demonstrate the robustness of the proposed algorithm as the cardioid curve has a singular point located at \(x = 0\) and \(y = 0\) (see Figure 10). In addition, it forms a closed
nonconvex region in the $x-y$ plane, thus many types of intersections can arise (see Figure 11). These examples show that the algorithm for intersection finding is independent of the kind of yield surface; it works for convex or nonconvex yield functions with or without singularities.

5.2. Barcelona basic model

The well-known BBM [21] for unsaturated soils possesses a nonconvex yield surface. Here, we restrict attention to the $s-\bar{p}$ plane, i.e. for isotropic net stresses. Only a brief review of the BBM
Figure 9. Second-order explicit Runge–Kutta-based stress update algorithm with substepping for models with nonconvex yield surfaces (continued).

\[
\sigma^{ME} = \sigma + 0.5(\Delta \sigma^1 + \Delta \sigma^2)
\]
\[
z_k^{ME} = z_k + 0.5(\Delta z_k^1 + \Delta z_k^2)
\]

Local error estimator and next step multiplier
\[
error = \max(||\sigma^{ME} - \sigma^{FE}||/\sqrt{1 + ||\sigma^{ME}||}, ||z_k^{ME} - z_k^{FE}||/\sqrt{1 + ||z_k^{ME}||})
\]
\[
m = 0.9\sqrt{STOL/error}
\]

if (error \leq STOL) then ! Step accepted

! Update
\[
T = T + \Delta T, \quad v = (1 - \Delta \varepsilon_{T1} - \Delta \varepsilon_{T2} - \Delta \varepsilon_{T3})v, \quad \varepsilon = \varepsilon + \Delta \varepsilon_T,
\]
\[
\sigma = \sigma^{ME}, \quad z_k = z_k^{ME}, \quad \text{and} \quad s = s^{FE}
\]

! Correct any drift
\[
FTOL = |g_k(0)ATOL|, \quad f^{new} = f(\sigma, z_k, s), \quad it = 0
\]

while (f^{new} > FTOL) do

r, V, y_k, S, \dot{H}_k, h^F = \text{CalculateGradientsAndHardening}(v, \sigma, z_k, s)
\[
\Delta \Lambda = f^{new} / |V : D^* : r + h^F|
\]
\[
\sigma = \sigma - \Delta \Lambda D^* : r
\]
\[
z_k = z_k + \Delta \Lambda H_k
\]
\[
f^{new} = f(\sigma, z_k, s)
\]
\[
it = it + 1
\]

if (it > MAXITDC) then stop (“Drift-corr. did not conv.”)
endwhile

if (m > m_{MAX}) then m = m_{MAX}! Upper bound stepsize change
else ! Step rejected
| if (m < m_{MIN}) then m = m_{MIN}! Lower bound stepsize change
endif

! Next substep size
\[
\Delta T = m\Delta T
\]

if (\Delta T > 1 - T) then \Delta T = 1 - T
endfor

stop (“Stress-update did not converge”)

The yield function is an extension of the modified Cam clay model [36], given by
\[
f(\tilde{\sigma}, q, \bar{\sigma}, z_0) = q^2 - M^2[\tilde{\sigma} - p_\lambda(s)][p_\lambda(s) - \tilde{\sigma}]
\]
in which the compressibility is related to the suction by means of the following expressions:
\[
\hat{\lambda}(s) = \hat{\lambda}_0[(1 - r)e^{-\beta s} + r]
\]
\[
\Psi(s) = \frac{\hat{\lambda}_0 - \kappa}{\hat{\lambda}(s) - \kappa}
\]

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/nme
Figure 10. Cardioid nonconvex fictional yield surface with one intersection at the singular point.

Figure 11. Cardioid nonconvex fictional yield surface with three intersections.

Table I. Parameters used with the Barcelona basic model (BBM).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_0 = 0.02$</td>
<td>Compression index for stress changes with null suction</td>
</tr>
<tr>
<td>$\kappa = 0.01$</td>
<td>Swelling index for stress changes</td>
</tr>
<tr>
<td>$M = 1.37$</td>
<td>Slope of critical state line</td>
</tr>
<tr>
<td>$r = 0.95$</td>
<td>Maximum soil stiffness parameter</td>
</tr>
<tr>
<td>$\beta = 0.1$</td>
<td>Rate of increase of soil stiffness with suction</td>
</tr>
<tr>
<td>$k = 0.3$</td>
<td>Increase in cohesion with suction</td>
</tr>
<tr>
<td>$p_{r_{ef}} = 10$</td>
<td>Reference pressure</td>
</tr>
</tbody>
</table>
A shift in the yield surface for increasing suction is given by

\[ p_s(s) = \begin{cases} 
-ks & \text{if } s > 0 \\
-s & \text{otherwise}
\end{cases} \quad (49) \]

and the size of the yield surface is related to suction according to

\[ p_y(s) = \begin{cases} 
(p_{\text{ref}}(z_0/p_{\text{ref}})\Psi(s)) & \text{if } s > 0 \\
z_0 - s & \text{otherwise}
\end{cases} \quad (50) \]

in which \( z_0 \) is an internal variable related to the hardening for zero suction values.

For given stress–suction increments, Equations (42)–(45) can be used to compute \( f_3, g_2, \) and \( h_2 \), after the substitution of \( x_z \) and \( y_z \) by \( \bar{p} \) and \( s \), respectively, in Equations (42)–(45), for example. All derivatives necessary for these equations are provided in Appendix A.

Figures 12–14 illustrate three situations where the BBM yield surface is crossed two or three times during drying or wetting. The parameters for the BBM are summarized in Table I and the
initial size of the yield surface was set as \( z_0 = 60 \). It can be concluded that the algorithm works well even though this yield surface has a discontinuity when the suction is zero.

5.3. Cam clay with Argyris et al. failure criterion

The next example uses the modified Cam clay model [36], where the yield surface is defined as

\[
f(\sigma, z_0) = f(p, q, t, z_0) = q^2 - M(t)^2 p(z_0 - p)
\]  

To follow the Argyris et al. failure criterion [26], the slope of the critical state line \( M \) is defined by

\[
M(t) = \frac{2\rho M_{CS}}{1 + \rho - (1 - \rho)t}
\]  

in which \( M_{CS} \) is the slope at the critical state under compression and \( \rho \) is given by

\[
\rho = \frac{3 - \sin \phi_{CS}}{3 + \sin \phi_{CS}}
\]  

In the above equation, \( \phi_{CS} \) is the shear angle at the critical state under compression. The first and second derivatives of \( M(t) \) with respect to the stress invariant \( t \) are necessary for the computation of the number of intersections. These derivatives are easily obtained as follows:

\[
\frac{dM}{dt} = \frac{M(t)(1 - \rho)}{1 + \rho - (1 - \rho)t} \quad \text{and} \quad \frac{d^2M}{dt^2} = \frac{2M(t)(1 - \rho)^2}{[1 + \rho - (1 - \rho)t]^2}
\]  

In Equation (51), \( z_0 \) is the only internal variable related to the size of the yield surface and \( p, q, \) and \( t \) are stress invariants defined by the following expressions:

\[
p = \frac{\text{tr} \sigma}{3}, \quad q = \sqrt{\frac{2}{3}} ||s||_2 \quad \text{and} \quad t = \frac{27 \det s}{2 q^3}
\]  

in which \( s \) is the second-order deviator tensor that can be calculated from

\[
s = P^{ad}: \sigma = \sigma - \frac{\text{tr} \sigma}{3} I
\]
where $\mathbf{P}^{sd}$ is the fourth-order isotropic tensor that converts any second-order tensor into its symmetric-deviator form and is given by

$$
\mathbf{P}^{sd} = \mathbf{I}^{\text{sym}} - \frac{1}{3} \mathbf{I} \otimes \mathbf{I}
$$

where $\otimes$ represents the dyadic or tensor product, which generates a fourth-order tensor from two second-order tensors. In an orthonormal Cartesian system, the components involved in such an operation are expressed as

$$
[I \otimes I]_{ijkl} = \delta_{ij} \delta_{kl}
$$

in which $\delta_{ij}$ is Kronecker’s delta. In Equation (57), $\mathbf{I}^{\text{sym}}$ is the fourth-order identity tensor of second-order tensors, which is fully symmetric and may be conveniently represented in a 6D vector space. In addition, it converts any tensor into its symmetric form and is given by

$$
\mathbf{I}^{\text{sym}} = \frac{1}{2} (\mathbf{I} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{I})
$$

in which the two alternative dyadic products $\overline{\otimes}$ and $\otimes$ allow for the generation of different fourth-order tensors using a different combination of components as indicated by (for an orthonormal Cartesian system)

$$
[a \otimes b]_{ijkl} = a_{ik} b_{jl} \quad \text{and} \quad [a \overline{\otimes} b]_{ijkl} = a_{ij} b_{jk}
$$

Further details regarding tensorial analysis and the notation used here may be found in [37–40].

The stress invariant $t$ is related to the Lode angle by

$$
t = \sin 3\theta
$$

which is used in the formulations instead of the Lode angle directly, because its derivatives are continuous. The derivatives of the stress invariants with respect to the stress tensor are simply

$$
\frac{dp}{d\sigma} = \frac{1}{3}, \quad \frac{dq}{d\sigma} = \frac{3s}{2q} \quad \text{and} \quad \frac{dt}{d\sigma} = \frac{9}{2q^2} \left( \frac{3}{q} \mathbf{P}^{sd} : s^2 - ts \right)
$$

in which $s^2$ is the square of the deviator stress given by $s^2 = \mathbf{s} \cdot \mathbf{s}$. Note that the letter ‘d’ was used instead of ‘∂’ since these equations are the derivatives of one entity, a scalar, with respect to another entity, a tensor, and not its components. The second derivatives of the stress invariants with respect to the stress tensor are given as follows:

$$
\frac{d^2 p}{d\sigma^2} = 0, \quad \frac{d^2 q}{d\sigma^2} = \frac{3}{2q} \mathbf{P}^{sd} - \frac{9}{4q^3} \mathbf{s} \otimes \mathbf{s}
$$

and

$$
\frac{d^2 t}{d\sigma^2} = \frac{27}{2q^3} \mathbf{T} - \frac{81}{2q^4} (\mathbf{P}^{sd} : s^2) \otimes \frac{dq}{d\sigma} - \frac{9t}{2q^2} \mathbf{P}^{sd} + \mathbf{s} \otimes \left( \frac{9t}{q^3} \frac{dq}{d\sigma} - \frac{9}{2q^2} \frac{dt}{d\sigma} \right)
$$

in which $\mathbf{T}$ is a fourth-order tensor defined by

$$
\mathbf{T} = (\mathbf{I} \otimes \mathbf{s} + \mathbf{s} \otimes \mathbf{I}) : \mathbf{P}^{sd} - \frac{2}{3} \mathbf{I} \otimes \mathbf{s}
$$
Now, the first- and second-order derivatives of the yield function can be obtained. These are given by

\[
\frac{\partial f}{\partial \sigma} = 3s + mI + n \frac{dM}{d\sigma} \tag{66}
\]

and

\[
\frac{\partial^2 f}{\partial \sigma^2} = 3p^{sd} + I \otimes \frac{dm}{d\sigma} + n \frac{d^2M}{d\sigma^2} + \frac{dM}{d\sigma} \otimes \frac{dn}{d\sigma} \tag{67}
\]

in which

\[
m = \frac{1}{3}M(t)^2(2p - z_0), \quad n = 2M(t)p(p - z_0) \tag{68}
\]

\[
\frac{dm}{d\sigma} = \frac{2M^2}{9}I + \frac{2M}{M} \frac{dM}{d\sigma}, \quad \frac{dn}{d\sigma} = \frac{2M}{M}I + \frac{n}{M} \frac{dM}{d\sigma} \tag{69}
\]

and the derivative of \(M\) with respect to the stress tensor is

\[
\frac{dM}{d\sigma} = \frac{dM(t)}{dt} \frac{dt}{d\sigma} \tag{70}
\]

The second derivative of \(M\) with respect to the stress tensor is given by

\[
\frac{d^2M}{d\sigma^2} = \frac{dM}{dt} \frac{d^2t}{d\sigma^2} + \frac{d^2M}{dt^2} \frac{dt}{d\sigma} \otimes \frac{dt}{d\sigma} \tag{71}
\]

To simplify the analysis, especially when computing fictional stress paths and plotting the yield surface, a 3D space using the eigendecomposition of the stress tensor can be considered. This space is quite convenient for the computation of the gradients and is well known as Haigh–Westergaard space where the axes correspond to the principal stresses \(\sigma_1, \sigma_2,\) and \(\sigma_3\). For the analyses presented here, this space is used for the definition of the stress invariants, including the Lode angle. In Appendix B, all first- and second-order derivatives are presented, and all equations needed for the stress update of the Cam clay model with the Argyris et al. criterion are given.

Figure 15 illustrates a section of the yield surface (AR) defined by Equation (51) using a shear angle equal to 48°, where nonconvexity is clearly observed. The sections of the Matsuoka–Nakai (MN) [41] and Argyris–Sheng et al. (AS) [25] criteria are also shown. A stress path is defined in such a way that it crosses the yield surfaces twice. The computed values of the yield function, its first derivative, and second derivative along with the stress path are also presented. It is possible to conclude that the algorithm works very well for this case. In addition, two more tests where the yield surface is crossed only once are presented in Figures 16 and 17. In the test illustrated in Figure 16, the stress path passes near the centre of the \(\Pi\) plane where the second derivative has a discontinuity. In Figure 17, the path passes exactly by the centre of the \(\Pi\) plane. For both cases, the performance of the algorithm is good.

6. ACCURACY AND EFFICIENCY ASSESSMENT

In this section, the accuracy and efficiency of the stress update algorithm are investigated by means of isoerror surfaces and finite element analysis.
Figure 15. Cam clay with Argyris et al. criterion and a nonconvex yield surface with two intersections.

Figure 16. Cam clay with Argyris et al. criterion and a nonconvex yield surface with one intersection and path near the centre of the $\Pi$ plane.

6.1. Isoerror surfaces

To assess the overall accuracy of the algorithm, isoerror surfaces are elaborated following the method described in [7, 8]. The stress update algorithm is called for a set of given strain increments, in the sense of a strain-controlled homogeneous problem. Although this technique assesses the overall accuracy of the algorithm, Simo and Hughes [8] advise that it should not be regarded as a replacement for a rigorous accuracy and stability analysis. At this stage, the paper considers a numerical assessment only since a rigorous study is still under development for the complicated problem of nonconvex surfaces.

Here, the model described in Section 5.3 is used for the study of accuracy. This is similar to the modified Cam clay model presented in [36] with an isotropic hardening law, but the nonconvex failure criterion of Argyris et al. [26] is adopted. The constitutive parameters are listed in Table II where a description for each one is also included.
Figure 17. Cam clay with Argyris et al. criterion and a nonconvex yield surface with one intersection and a path passing by the centre of the $\Pi$ plane.

Table II. Parameters for the modified Cam clay model with Argyris et al. failure criterion used in the accuracy assessment.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_0 = 0.09$</td>
<td>Compression index</td>
</tr>
<tr>
<td>$\kappa = 0.002$</td>
<td>Swelling index</td>
</tr>
<tr>
<td>$\phi_{CS} = 40$</td>
<td>Shear-strength angle for compression at the critical state</td>
</tr>
<tr>
<td>$G = 3920$ (kPa)</td>
<td>Shear modulus</td>
</tr>
</tbody>
</table>

Figure 18. 3D initial states on the yield surface of the modified Cam clay model with Argyris et al. nonconvex criterion: (a) point $A$; (b) point $B$; (c) point $C$; and (d) point $D$.

Four initial states on the yield surface are selected representing a range of possible states of stress. These states are indicated by small spheres (points) in Figure 18 and are summarized in Table III. For each point, a set of increments of strain, specified by the triad of principal values $\Delta e_x$, $\Delta e_y$, and $\Delta e_z$, is applied and the resulting stresses are calculated using the stress update algorithm. The set of strain increments is defined according to all combinations of principal values ranging from 0 to 6% in increments of 0.5% as indicated in Figure 19. Using this approach, the maximum volumetric strain increment and deviatoric strain invariant increment are $\Delta V_{V} = 6\%$ and $\Delta D_{D} = 6\%$, respectively. The Lode angle for the strain increments varies from 0 to 360°.
Table III. Initial states on the yield surface used for the accuracy assessment ($z_0 = 2.0$).

<table>
<thead>
<tr>
<th></th>
<th>Point A</th>
<th>Point B</th>
<th>Point C</th>
<th>Point D</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_x$</td>
<td>2.000</td>
<td>1.458</td>
<td>1.051</td>
<td>0.455</td>
</tr>
<tr>
<td>$\sigma_y$</td>
<td>2.000</td>
<td>1.458</td>
<td>1.628</td>
<td>0.455</td>
</tr>
<tr>
<td>$\sigma_z$</td>
<td>2.000</td>
<td>2.458</td>
<td>2.206</td>
<td>2.091</td>
</tr>
<tr>
<td>$p$</td>
<td>2.000</td>
<td>1.791</td>
<td>1.628</td>
<td>1.000</td>
</tr>
<tr>
<td>$q$</td>
<td>0.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.636</td>
</tr>
<tr>
<td>$t$</td>
<td>−1.000</td>
<td>1.000</td>
<td>0.000</td>
<td>1.000</td>
</tr>
<tr>
<td>Lode (deg.)</td>
<td>0.000</td>
<td>0.000</td>
<td>30.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Figure 19. Combination of strain increments applied to the stress update algorithm.

The results are reported as the relative root mean square of the error between the ‘exact’ ($\sigma_i^*$) and computed solution ($\sigma_i$), in which the index $i$ corresponds to each increment of strain defined according to the grid of Figure 19. This error is obtained according to the following expression:

$$\text{error}_i = \frac{\|\sigma_i - \sigma_i^*\|_2}{\|\sigma_i^*\|_2 + 1}$$  \hspace{1cm} (72)

in which the Euclidian norm $\|a\|_2$ of a second-order tensor $a$ is calculated as

$$\|a\|_2 = \sqrt{a : a}$$  \hspace{1cm} (73)

The ‘exact’ solution is computed using the stress update algorithm with a very small tolerance \(STOL = 10^{-9}\). Therefore, a set of isoerror surfaces can be elaborated using each strain increment position and the error computed with the stress update algorithm.
The isoerror surfaces computed with the results from the numerical integration are illustrated in Figures 20, where it is possible to observe that the error increases strongly with increasing volumetric strains. In Figure 21, the variation of the error for a fixed value of the volumetric strain increment ($\Delta \varepsilon_v = 6\%$) is presented. From these figures, it can be observed that the variation of the error for point $A$ is symmetric around the centre, symmetric for the points $B$ and $D$, and nonsymmetric for point $C$. These results are consistent with the initial states on the yield surface.

In Figure 22, the results from all computations are summarized. The minimum, average, and maximum error for each initial point $A$, $B$, $C$, and $D$ are plotted against the tolerances STOL for STOL equal to $10^{-4}$, $10^{-5}$, $10^{-6}$, $10^{-7}$, and $10^{-8}$. The computation time for the integration is also given on the right-hand side of each plot. As expected, it can be seen that for all points $A$, $B$, $C$, and $D$, the error decreases with decreasing STOL values, while the computation time increases accordingly. The increase in time reflects the increasing number of substeps, as determined by the algorithm, leading to a relatively accurate scheme.
Figure 21. Results from the numerical integration for STOL $= 10^{-6}$; octahedral plane at $\Delta \epsilon_1 = \Delta \epsilon_2 = \Delta \epsilon_3 = 2\%$ ($\Delta \epsilon_Y = 6\%$) integration (STOL $= 10^{-6}$): (a) point A; (b) point B; (c) point C; and (d) point D.

6.2. Finite element analysis

Next, attention is focused on verification of the stress update algorithm inside a finite element program. The BVP selected corresponds to a true triaxial experiment on a sample of Fujinomori clay from Japan (Figure 23). The solution of this problem should lead to results similar to the ones computed with local (point) integration of the constitutive relation selected. In addition, some experimental results are provided in order to check, at least qualitatively, the behaviour predicted. The stress path applied during this experiment includes the variation of three principal stresses, yielding a 3D situation of stress components (see the stress path in Figure 24). The boundary conditions are also illustrated in Figure 23. The finite element mesh contains 27 elements as illustrated in Figure 23. The results presented here (stress, strain) are the averages of the values obtained from the eight integration points inside the hexahedron, which is located at the centre of the mesh.

Two models are considered: the modified Cam clay model with the nonconvex Argyris et al. [26] criterion, here denoted by CCAR, and the same Cam clay model, but with the convex (for $\phi_{CS} < 48.6^\circ$) AS et al. criterion [25], denoted by CCAS. For the selected value of friction angle

DOI: 10.1002/nme
\( \phi_{CS} = 31.6 \), it is expected that the simulations using the CCAS model would run faster than those with the CCAR model, because the CCAR model requires the use of the nonconvex update algorithm, while the CCAS model can be solved with ‘conventional’ techniques such as the ones discussed in Sloan et al. [17]. The difference in speed results from the fact that the ‘conventional’ method does not require the computation of second-order gradients.

The parameters for the FEM simulations are summarized in Table IV. The global solution method of the finite element program used is based on the algorithm presented in [42], which adopts an automatic subincrementation of the (global) time step. The simulations with the CCAS model used the stress update explained in [17] which also employs a substepping strategy controlled by the local error STOL; however, for convex surfaces only. Both simulations, with CCAR and CCAS models, were carried out with \( \text{STOL} = 10^{-4} \).

The results of the simulations, together with experimental values, are presented in Figure 24. Although the constitutive models did not provide very good predictions of the clay behaviour, which is complicated in this case by the 3D stress path and cyclic loading, the solution obtained...
Figure 23. Finite element analysis of a laboratory test on a sample of Fujinomori clay: (a) boundary conditions ($t$ stands for tractions) and (b) mesh of 27 hexahedra of 8 nodes.

Figure 24. Results from the FEM simulations and experimental tests.
Table IV. Parameters for the Fujinomori clay of Japan set for the two modified Cam clay models: CCAR and CCAS used in the finite element analysis.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_0 = 0.0891$</td>
<td>Compression index</td>
</tr>
<tr>
<td>$\kappa = 0.0196$</td>
<td>Swelling index</td>
</tr>
<tr>
<td>$\phi_{CS} = 31.6$</td>
<td>Shear-strength angle for compression at the critical state</td>
</tr>
<tr>
<td>$G = 18130$ (kPa)</td>
<td>Shear modulus</td>
</tr>
</tbody>
</table>

for the model with the nonconvex surface (CCAR) was very similar to the one computed with the conventional Cam clay model and a convex surface (CCAS). It can be concluded that the accuracy of the simulations with the algorithm for the nonconvex surface is satisfactory; however, the computer time for the nonconvex case was 14 times greater than that with the CCAS model.

7. CONCLUSIONS

A stress update algorithm for elastoplastic models with nonconvex yield surfaces has been presented. The algorithm is based on an RK-embedded method of second order and features adaptive subincrementation according to a local error estimate.

The difficult task of detecting the loading–unloading transition is solved using a simple algorithm that analyses the signs of the yield function along a secant trial stress path at a ‘checkpoint’ position. As multiple intersections may arise during the stress update, a multiple-root-finding algorithm has been developed. This algorithm is based on the Kronecker–Picard (KP) formula that provides the number of roots inside a closed interval for any function which is continuously differentiable to the second order. The KP formula was incorporated into a recursive algorithm and used to compute all intersections. This algorithm requires a numerical integration (quadrature) and a single-root-finding algorithm. For the latter, the robust method known as Brent’s method was selected.

A simple fictional nonconvex yield surface with the Cardioid shape, a model for unsaturated soils (the BBM), and a more complex 3D nonconvex surface were considered during the accuracy assessment of the intersection-finding algorithm. The 3D surface is based on the modified Cam clay model with the nonconvex Argyris et al. yield criterion in the deviatoric plane and yields a relatively complicated system of equations. For all tests, it has been verified that the algorithm performs quite well.

An assessment of accuracy and efficiency is presented using two different techniques: (a) with the aid of isoerror surfaces as discussed by Simo and Hughes [8] and (b) from finite element analysis of a small BVP. The isoerror surfaces are elaborated after the computation of the stresses for a given set of strain increments. A 3D set of principal values of strain increments are used for this task. The finite element analysis solves a BVP corresponding to a laboratory true triaxial test on a cubic sample of Fujinomori clay. Both techniques suggest that the algorithm proposed is both efficient and accurate, with the degree of the latter being controlled by STOL. However, the solution for nonconvex yield surfaces is much slower than that for similar models with convex yield surfaces, due mainly to the higher-order gradients required for the computation of the number of intersections.
APPENDIX A

This appendix summarizes all derivatives required for the stress update of the BBM using the algorithms proposed. Note that the second derivatives are also necessary due to the KP formula (Equation (4))

\[
\frac{d\lambda}{ds} = -\beta \lambda_0 (1-r)e^{-\beta s} \quad (A1)
\]

\[
\frac{d\Psi}{ds} = \frac{-\Psi(s) \ d\lambda}{\lambda(s) - \kappa} \quad (A2)
\]

\[
\frac{dp_s}{ds} = \begin{cases} 
-k & \text{if } s > 0 \\
-1 & \text{otherwise} 
\end{cases} \quad (A3)
\]

\[
\frac{dp_y}{ds} = \begin{cases} 
 p_y \ln(z_0/p_{ref}) \frac{d\Psi}{ds} & \text{if } s > 0 \\
-1 & \text{otherwise} 
\end{cases} \quad (A4)
\]

\[
\frac{\partial f}{\partial \bar{p}} = M^2(2\bar{p} - p_s - p_y) \quad (A5)
\]

\[
\frac{\partial f}{\partial s} = M^2 \left[ (p_y - \bar{p}) \frac{dp_s}{ds} + (p_s - \bar{p}) \frac{dp_y}{ds} \right] \quad (A6)
\]

\[
\frac{\partial^2 \lambda}{\partial s^2} = \beta^2 \lambda_0 (1-r)e^{-\beta s} \quad (A7)
\]

\[
\frac{\partial^2 \Psi}{\partial s^2} = \frac{-\Psi(s) \ d^2\lambda}{\lambda(s) - \kappa} \frac{1}{ds^2} + \frac{d\lambda}{\lambda(s) - \kappa} \frac{d\Psi(s) \ d\lambda}{\lambda(s) - \kappa} - \frac{d\Psi}{ds} \quad (A8)
\]

\[
\frac{d^2 p_s}{ds^2} = \begin{cases} 
 p_y \ln(z_0/p_{ref}) \frac{d^2\Psi}{ds^2} + \ln(z_0/p_{ref}) \frac{d\Psi}{ds} \frac{dp_y}{ds} & \text{if } s > 0 \\
0 & \text{otherwise} 
\end{cases} \quad (A9)
\]

\[
\frac{\partial^2 f}{\partial \bar{p}^2} = 2M^2 \quad (A10)
\]

\[
\frac{\partial^2 f}{\partial s^2} = M^2 \left[ (p_y - \bar{p}) \frac{d^2 p_s}{ds^2} + 2 \frac{dp_s}{ds} \frac{dp_y}{ds} + (p_s - \bar{p}) \frac{d^2 p_y}{ds^2} \right] \quad (A11)
\]

\[
\frac{\partial^2 f}{\partial \bar{p} \partial s} = -M^2 \left( \frac{dp_s}{ds} + \frac{dp_y}{ds} \right) \quad (A12)
\]

APPENDIX B

In order to facilitate plotting and analyses of problems such as intersection finding, the Haigh–Westergaard principal stress space can be used conveniently. In this space, it is possible to define...
a ‘vector of stress’ $\Sigma$ according to the following notation:

$$\Sigma = \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \end{bmatrix}_{1-2-3}$$  \hspace{1cm} (B1)

in which the components of this stress vector are valid only on the Haigh–Westergaard space as indicated by the subscripts ‘1–2–3’. The correspondent deviator vector $S$ is given by

$$S = \frac{1}{3} \begin{bmatrix} 2\sigma_1 - \sigma_2 - \sigma_3 \\ 2\sigma_2 - \sigma_3 - \sigma_1 \\ 2\sigma_3 - \sigma_1 - \sigma_2 \end{bmatrix}_{1-2-3}$$  \hspace{1cm} (B2)

With this notation, the first derivatives of the stress invariants with respect to the stress vector can be obtained as follows:

$$\frac{dp}{d\Sigma} = I, \quad \frac{dq}{d\Sigma} = \frac{3S}{2q} \quad \text{and} \quad \frac{dt}{d\Sigma} = \frac{27\ell}{4q^5} B$$  \hspace{1cm} (B3)

in which the auxiliary vectors $I$ and $B$ are defined according to

$$I = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}_{1-2-3}, \quad B = \begin{bmatrix} \sigma_3 - \sigma_2 \\ \sigma_1 - \sigma_3 \\ \sigma_2 - \sigma_1 \end{bmatrix}_{1-2-3}$$  \hspace{1cm} (B4)

and the scalar $\ell$ is defined according to

$$\ell = (\sigma_1 - \sigma_2)(\sigma_2 - \sigma_3)(\sigma_3 - \sigma_1)$$  \hspace{1cm} (B5)

The second-order derivatives are then given as follows:

$$\frac{d^2p}{d\Sigma^2} = 0, \quad \frac{d^2q}{d\Sigma^2} = \frac{3p^{sd}}{2q} - \frac{9}{4q^3} S \otimes S$$  \hspace{1cm} (B6)

and

$$\frac{d^2t}{d\Sigma^2} = \frac{27}{4} \left[ \frac{\ell}{q^5 d\Sigma} dB \otimes \left( \frac{1}{q^5 d\Sigma} \frac{d\ell}{d\Sigma} - \frac{5\ell dq}{q^6 d\Sigma} \right) \right]$$  \hspace{1cm} (B7)

in which the additional auxiliary entities are

$$P^{sd} = \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}_{1-2-3}, \quad dB = \begin{bmatrix} 0 & -1 & 1 \\ 1 & 0 & -1 \\ -1 & 1 & 0 \end{bmatrix}_{1-2-3}$$  \hspace{1cm} (B8)
\[ \frac{d\ell}{d\Sigma} = 3 \begin{cases} (\sigma_3 - \sigma_2)S_1 \\ (\sigma_1 - \sigma_3)S_2 \\ (\sigma_2 - \sigma_1)S_3 \end{cases}_{1-2-3} \]  

(B9)

Now, the first and second derivatives of the yield surface in terms of the principal stress values can be obtained as follows:

\[ \frac{\partial f}{\partial \Sigma} = 3S + mI + n \frac{dM}{d\Sigma} \]  

(B10)

and

\[ \frac{\partial^2 f}{\partial \Sigma^2} = 3P' + I \otimes \frac{dm}{d\Sigma} + n \frac{d^2M}{d\Sigma^2} + \frac{dM}{d\Sigma} \otimes \frac{dn}{d\Sigma} \]  

(B11)

in which

\[ \frac{dm}{d\Sigma} = \frac{2M^2}{9}I + \frac{2m}{M} \frac{dM}{d\Sigma}, \quad \frac{dn}{d\Sigma} = \frac{2m}{M}I + \frac{n}{M} \frac{dM}{d\Sigma} \]  

(B12)

and the derivative of \( M \) with respect to the stress tensor is

\[ \frac{dM}{d\Sigma} = \frac{dM}{dt} \frac{dt}{d\Sigma} \]  

(B13)

The second derivative of \( M \) with respect to the stress tensor is given by

\[ \frac{d^2M}{d\Sigma^2} = \frac{dM}{dt} \frac{d^2t}{d\Sigma^2} + \frac{d^2M}{dt^2} \frac{dt}{d\Sigma} \]  

(B14)

ACKNOWLEDGEMENTS

The computer tools freely provided by the MechSys (http://mechsys.nongnu.org) and ParaView (http://www.paraview.org) developer teams are deeply appreciated. The authors are also grateful to the reviewers for discerning comments on this paper.

REFERENCES


