Algorithm 1 NorSand model algorithm

Assembling the tangent matrix D_{ijkl}

$\delta\sigma_{ij} = D_{ijkl}\delta\epsilon_{kl}$

$\sigma_{trial} = \sigma + \delta\sigma$

$J_2 = \frac{1}{6}((\sigma_{11} - \sigma_{22})^2 + (\sigma_{11} - \sigma_{33})^2 + (\sigma_{33} - \sigma_{22})^2 + 6(\sigma_{12}^2 + \sigma_{23}^2 + \sigma_{33}^2))$

$p_{trial} = \sigma_i j\delta_{ij}/3,\ q_{trial} = \sqrt{3J_2}$

Then we going to calculating the yield surface

$M_i = M_{tc} \cdot (1 - \frac{\chi N|\psi|}{M_{tc}})$

$p_{max} = p_i \exp\left(\frac{\chi\psi}{M_{tc}}\right)$

$F = q_{trial} - p_{trial}M_i(1 - \ln(\frac{p_{trial}}{p_i})),\ F_2 = p_{trial} - p_{max}$

Algorithm 2 Procesures to generate the random loading path via gaussian process and principal rotation

Require: Given: The ampltitude for principle strains

Algorithm 3 Summary of the procedure used to calculate the total stress with regard to the give strain increment

Require: Given: the total stress σ and the total strain ϵ at current step, $\Delta\epsilon$ is to be applied to the configuration; Yield function F

Ensure: $F(\sigma) \leq 0$

$\Delta\hat{\sigma} = C_E \cdot \Delta\epsilon$ ▷ Calculate the stress increment assuming elastic behavior
$\hat{\sigma} = \sigma + \Delta\hat{\sigma}$ ▷ Calculate the trial stress
$F(\hat{\sigma})$ ▷ Calculate the value of the yield function, with $\hat{\sigma}$ as the state of stress,
if $F(\hat{\sigma}) \leq 0$ **then**
    $\sigma = \hat{\sigma}$ ▷ The strain increment is elastic. In this case, the trial stress is correct; we return.
    return σ
else
    if The previous stress is plastic **then**
        $r = 0$ ▷ r is the portion of incremental strain taken elastically.
    else ▷ There is a transition from elstic to plastic
        Determine r via $F(\sigma + r \cdot \Delta\epsilon) = 0$
        $\sigma = C_E \cdot r\Delta\epsilon$
        $\Delta\epsilon^p = (1 - r) \cdot \Delta\epsilon$ ▷ $\Delta\epsilon^p$ is the elastoplastic strain increment
        Call ELASTOPLASTIC RETURN MAPPING($\Delta\epsilon^{ep}$)
    end if
end if
return σ

Algorithm 4 Elastoplastic return mapping (Convergence of this algorithm needs verification)

Require: Given: dϵ, $H(\bar{\epsilon^p})$

Ensure: $f(\sigma, H) = 0$

$k = 1$, $\sigma^{(n+1,k)} = \sigma^{(n)} + D\mathrm{d}\epsilon$ ▷ Calculate the trial stress as the begining of the return mapping iteration

while $f(\sigma^{(n+1,k)}, H(\bar{\epsilon^p}^{(k)})) > 0$ **do** ▷ $f = g$ in Association Flow assumption

  $h = -(\frac{\partial f}{\partial \int |\mathrm{d}\bar{\epsilon^p}|}\|\frac{\partial f}{\partial \sigma}\|)^{(k)}$

  $H = h + \frac{\partial f}{\partial \sigma} D \frac{\partial g}{\partial \sigma}$

  $\mathrm{d}\lambda = \frac{1}{H}\frac{\partial f}{\partial \sigma} D\mathrm{d}\epsilon$

  $\mathrm{d}\epsilon^p = \mathrm{d}\lambda \frac{\partial g}{\partial \sigma}^{(k)}$

  $\bar{\epsilon^p}^{(n+1,k)} = \bar{\epsilon^p}^{(n+1,k-1)} + \|\mathrm{d}\epsilon^p\|$

  $\sigma^{(n+1,k)} = \sigma^{(n+1,k-1)} - D\mathrm{d}\epsilon^p$

  $k++$

end while

if $f(\sigma^{(n+1,k)}, H(\bar{\epsilon^p}^{(k)})) < -f_{tolerance}$ **then** ▷ Avoiding overestimating plasticity

  Searching for the right scalar of plastic strain via dichotomy(dλ)

end if

$\mathrm{d}\epsilon^p = \mathrm{d}\lambda \cdot \frac{\partial g}{\partial \sigma}^{(k)}$

$\bar{\epsilon^p} = \bar{\epsilon^p} + \|\mathrm{d}\epsilon^p\|$

$\sigma = \sigma + D \cdot (\mathrm{d}\epsilon - \mathrm{d}\epsilon^p)$

return σ

Algorithm 5 Searching for the right scalar of plastic strain via dichotomy

Require: $\mathrm{d}\lambda$
Ensure: $f_{tolerance} < f(\sigma, H) < 0$
$r_{min} = 0,\ r_{max} = 1$
while $f(\sigma, H) < -f_{tolerance}$ **or** $f(\sigma, H) > 0$ **do**
  $r_{mid} = (r_{min} + r_{max})/2$
  $\mathrm{d}\epsilon^p_{mid} = r_{mid}\mathrm{d}\epsilon^p$
  $\bar{\epsilon^p}_{mid} = \bar{\epsilon^p} + \|\mathrm{d}\epsilon^p_{mid}\|$
  $H_{mid} = \hat{H}(\bar{\epsilon^p}_{mid})$
  $\sigma_{mid} = \sigma_{trial} - D\mathrm{d}\epsilon^p_{mid}$
  $f_{mid} = \hat{f}(\sigma_{mid}, H_{mid})$
  if $f_{mid} > 0$ **then**
    $r_{min} = r_{mid}$
  else
    $r_{min} = r_{mid}$
  end if
end while
return $\mathrm{d}\epsilon^p_{mid}$