

## Parresia Research Report #PAR-10021867-Manual.v1

# Arena Sand/Soil Model Theory Manual

Author: **Biswajit Banerjee**

Prepared for: **Prof. Rebecca Brannon  
University of Utah**

Parresia Research Limited  
Thursday 1<sup>st</sup> September, 2016

---

Written permission is required if this report is to be reproduced in part or for promotional purposes.

# Contents

<b>1</b>	<b>A typical input file</b>	<b>3</b>
<b>2</b>	<b>The governing equations</b>	<b>6</b>
2.1	Momentum balance	6
2.2	Mass balance	6
2.3	Energy balance	7
2.4	Constitutive model	7
2.4.1	Rate-independent plasticity	8
2.4.2	Rate-dependent plasticity	14
<b>3</b>	<b>Arena Partially Saturated Soil Model: Computing the stress</b>	<b>16</b>
3.1	Initialization of the model	16
3.2	Computing the stress and internal variables	17
3.2.1	Compute elastic properties	18
3.2.2	Rate-independent stress update	21
3.2.3	Computing the trial stress	23
3.2.4	Computing the number of subcycles in the return algorithm	23
3.2.5	Updating the stress for a substep: consistency bisection	24
3.2.6	The nonhardening return algorithm	25
3.2.7	Consistency bisection algorithm	29
3.2.8	Rate-dependent plastic update	34
	<b>Appendices</b>	<b>37</b>
<b>A</b>	<b>Governing equations from mixture theory</b>	<b>37</b>
A.1	Volume fraction, porosity, saturation, density	37
A.2	Material derivatives	37
A.3	Mass balance	38
A.3.1	Solid phase	38
A.3.2	Water phase	40
A.3.3	Air phase	41
A.4	Linear momentum balance	42
A.4.1	Solid phase	42
A.4.2	Water phase	42
A.4.3	Air phase	42
A.4.4	Mixture	43
A.5	Energy balance	43
A.6	Simplified model	44
<b>B</b>	<b>Partially saturated bulk modulus</b>	<b>45</b>
<b>C</b>	<b>Pore pressure equation</b>	<b>46</b>
<b>D</b>	<b>Computing saturation from weight ratio</b>	<b>47</b>
<b>E</b>	<b>The Kayenta damage model</b>	<b>48</b>
<b>F</b>	<b>Implementing parameter variability</b>	<b>49</b>
<b>G</b>	<b>Density-dependence model</b>	<b>52</b>
<b>H</b>	<b>Verifying stress paths for uniaxial strain loading</b>	<b>52</b>

# 1 A typical input file

The inputs for the Arena soil model are typically specified as follows.

```
<?xml version='1.0' encoding='ISO-8859-1' ?>

<Uintah_specification>

  <Meta>
    <title>Uniaxial Strain Compression fully saturated</title>
  </Meta>

  <SimulationComponent type="mpm" />

  <Time>
    <maxTime> 1.0 </maxTime>
    <initTime> 0.0 </initTime>
    <delt_min> 1.0e-6 </delt_min>
    <delt_max> 0.01 </delt_max>
    <timestep_multiplier> 0.3 </timestep_multiplier>
  </Time>

  <DataArchiver>
    <filebase>UniaxialStrainSaturated.uda</filebase>
    <outputInterval>0.04</outputInterval>
    <save label = "p.x"/>
    <save label = "p.color"/>
    <save label = "p.temperature"/>
    <save label = "p.velocity"/>
    <save label = "p.particleID"/>
    <save label = "p.stress"/>
    <save label = "g.mass"/>
    <save label = "p.deformationMeasure"/>
    <save label = "g.acceleration"/>
    <save label = "p.capX"/>
    <save label = "p.plasticStrain"/>
    <save label = "p.plasticCumEqStrain"/>
    <save label = "p.plasticVolStrain"/>
    <save label = "p.p3"/>
    <save label = "p.porePressure"/>
    <save label = "p.ArenaPEAKI1"/>
    <save label = "p.ArenaFSLOPE"/>
    <save label = "p.ArenaSTREN"/>
    <save label = "p.ArenaYSLOPE"/>
    <save label = "p.ArenaBETA"/>
    <save label = "p.ArenaCR"/>
    <save label = "p.ArenaT1"/>
    <save label = "p.ArenaT2"/>
    <save label = "p.porosity"/>
    <save label = "p.saturation"/>
    <save label = "p.elasticVolStrain"/>
    <save label = "p.stressQS"/>
    <save label = "p.COHER"/>
    <save label = "p.TGROW"/>
    <checkpoint cycle = "2" timestepInterval = "4000"/>
  </DataArchiver>

  <MPM>
    <time_integrator> explicit </time_integrator>
    <interpolator> linear </interpolator>
    <use_load_curves> false </use_load_curves>
    <minimum_particle_mass> 1.0e-15 </minimum_particle_mass>
    <minimum_mass_for_acc> 1.0e-15 </minimum_mass_for_acc>
    <maximum_particle_velocity> 1.0e5 </maximum_particle_velocity>
```

```
<artificial_damping_coeff> 0.0 </artificial_damping_coeff>
<artificial_viscosity> true </artificial_viscosity>
<artificial_viscosity_heating> false </artificial_viscosity_heating>
<do_contact_friction_heating> false </do_contact_friction_heating>
<create_new_particles> false </create_new_particles>
<UseMomentumForm> false </UseMomentumForm>
<withColor> true </withColor>
<UsePrescribedDeformation> true </UsePrescribedDeformation>
<PrescribedDeformationFile> UniaxialStrain_PrescribedDeformation.inp </
  PrescribedDeformationFile>
<minimum_subcycles_for_F> -2 </minimum_subcycles_for_F>
<erosion_algorithm> = "none"/>
</MPM>

<PhysicalConstants>
  <gravity>[0,0,0]</gravity>
</PhysicalConstants>

<MaterialProperties>
  <MPM>
    <material_name>"MasonSand">
      <density>1800</density>
      <melt_temp>3695.0</melt_temp>
      <room_temp>294.0</room_temp>
      <thermal_conductivity>174.0e-7</thermal_conductivity>
      <specific_heat>134.0e-8</specific_heat>

    <constitutive_model type="ArenaSoil">

      <reference_porosity> 0.42 </reference_porosity>
      <initial_porosity> 0.40 </initial_porosity>
      <initial_saturation> 0.5 </initial_saturation>
      <initial_fluid_pressure> 0.0 </initial_fluid_pressure>

      <p0> 0.1e4 </p0>
      <p1> 482.68e6 </p1>
      <p1_sat> 1.0 </p1_sat>
      <p1_density_scale_fac> 5.0 </p1_density_scale_fac>
      <p2> 0.719 </p2>
      <p3> 0.448 </p3>

      <elastic_moduli_model type="arena">
        <b0> 0.0029 </b0>
        <b1> 0.4731 </b1>
        <b2> 1.5057 </b2>
        <b3> 2.5728 </b3>
        <b4> 2.0799 </b4>
        <G0> 7.0e8 </G0>
        <nu1> 0.35 </nu1>
        <nu2> -0.05 </nu2>
      </elastic_moduli_model>

      <plastic_yield_condition type="arena">
        <PEAKI1> 1.0e3 </PEAKI1>
        <weibullDist_PEAKI1> weibull, 1.0e3, 4, 0.001, 1 </weibullDist_PEAKI1>
        <FSLOPE> 0.453 </FSLOPE>
        <weibullDist_FSLOPE> weibull, 0.453, 4, 0.001, 1 </weibullDist_FSLOPE>
        <STREN> 1.0e7 </STREN>
        <weibullDist_STREN> weibull, 1.0e7, 4, 0.001, 1 </weibullDist_STREN>
        <YSLOPE> 0.31 </YSLOPE>
        <weibullDist_YSLOPE> weibull, 0.31, 4, 0.001, 1 </weibullDist_YSLOPE>
        <BETA> 1.0 </BETA>
        <CR> 0.5 </CR>
        <T1> 5.0e-5 </T1>
      </plastic_yield_condition>
    </MPM>
  </MaterialProperties>
```

```
<T2> 0.5 </T2>
</plastic_yield_condition>

<use_disaggregation_algorithm> false </use_disaggregation_algorithm>
<subcycling_characteristic_number>256</subcycling_characteristic_number>
<consistency_bisection_tolerance>0.0001</consistency_bisection_tolerance>
<yield_surface_radius_scaling_factor> 1000.0 </yield_surface_radius_scaling_factor>

<do_damage> true </do_damage>
<fspeed> 7 </fspeed>
<time_at_failure> 800.0e-6 </time_at_failure>

</constitutive_model>

<geom_object>
  <box label = "Plate1">
    <min>[0.0,0.0,0.0]</min>
    <max>[1.0,1.0,1.0]</max>
  </box>
  <res>[1,1,1]</res>
  <velocity>[0.0,0.0,0.0]</velocity>
  <temperature>294</temperature>
  <color>0</color>
</geom_object>

</material>

<contact>
  <type>null</type>
  <materials>[0]</materials>
  <mu>0.1</mu>
</contact>

</MPM>
</MaterialProperties>

<Grid>

  <BoundaryConditions>
    .....
  </BoundaryConditions>

  <Level>
    <Box label = "Domain">
      <lower>[-2.0, -2.0, -2.0]</lower>
      <upper>[3.0, 3.0, 3.0]</upper>
      <resolution>[5,5,5]</resolution>
      <extraCells>[0,0,0]</extraCells>
      <patches>[1,1,1]</patches>
    </Box>
  </Level>

</Grid>

</Uintah_specification>
```

## 2 The governing equations

The governing equations are based on the approach used by Uzuoka and Borja (Uzuoka and Borja, 2012).

### 2.1 Momentum balance

The momentum balance equation for the mixture is

$$\rho \mathbf{a}^s = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b}^s \quad \text{where} \quad \rho = \sum_{\alpha} \rho^{\alpha} \quad (1)$$

where  $\rho$  is the mass density of the mixture,  $\mathbf{a}^s$  is the acceleration of the solid skeleton,  $\boldsymbol{\sigma}$  is the total Cauchy stress in the mixture,  $\mathbf{b}^s$  is the body force experienced by the mixture, and  $\alpha = \{s, w, a\}$  are the three phases (solid, water, air). The effective stress is defined as

$$\boldsymbol{\sigma}_{\text{eff}} = \boldsymbol{\sigma} + B \bar{p}^w \mathbf{I} = \boldsymbol{\sigma} - \boldsymbol{\alpha} \quad (2)$$

where  $\bar{p}^w = -p^w$  is the pore pressure,  $\boldsymbol{\alpha} := -B \bar{p}^w \mathbf{I}$  can be treated as a backstress, and  $B$  is the Biot coefficient which is defined as

$$B := 1 - \frac{K_d(\bar{p}^s)}{K_s(\bar{p}^s)}. \quad (3)$$

Here  $p^s$  is the intrinsic pressure in the solid skeleton,  $K_d$  is the bulk modulus of the drained solid skeleton,  $K_s$  is the bulk modulus of the solid grains, and we assume that  $\bar{p}^s = \bar{p}^w$ . All unbarred quantities are positive in tension. For example,  $\boldsymbol{\sigma}$  and  $p^w$  are positive in tension while  $\bar{p}^w$  is positive in compression.

### 2.2 Mass balance

As shown in the previous report, with the modifications shown in Appendix C, the mass balance equations can be expressed as a rate equation for the pore pressure:

$$\frac{d\bar{p}^w}{d\bar{\epsilon}_v} = \frac{1}{B} \quad (4)$$

where  $\bar{\epsilon}_v$  is the total volumetric strain, and

$$B := \frac{1}{(1 - S_0) \exp(\bar{\epsilon}_v - \bar{\epsilon}_v^a) + S_0 \exp(\bar{\epsilon}_v - \bar{\epsilon}_v^w)} \left[ -\frac{(B - \phi) \phi}{B \phi_0} \left( \frac{S_w}{K_w} + \frac{1 - S_w}{K_a} \right) + \frac{1 - S_0}{K_a} \exp(\bar{\epsilon}_v - \bar{\epsilon}_v^a) + \frac{S_0}{K_w} \exp(\bar{\epsilon}_v - \bar{\epsilon}_v^w) \right]. \quad (5)$$

In the above expression,  $B$  is the Biot coefficient,  $\phi$  is the current total porosity,  $\phi_0$  is the initial total porosity,  $S_w$  is the total saturation,  $S_0$  is the initial total saturation,  $K_w$  is the pressure-dependent bulk modulus of water,  $K_a$  is the pressure-dependent bulk modulus of air,  $\bar{\epsilon}_v^a$  is the total strain in the air phase, and  $\bar{\epsilon}_v^w$  is the total strain in the water phase.

## 2.3 Energy balance

The energy balance equation of the mixture is

$$\rho \frac{D^s e}{Dt} - \sigma_{\text{eff}} : \mathbf{d}^s + \bar{p}^w \left[ \frac{S_w \phi}{K_w(\bar{p}^w)} - \frac{(1 - S_w) \phi}{K_a(\bar{p}^w)} \right] \frac{D^s \bar{p}^w}{Dt} + \nabla \cdot \mathbf{q} - \rho h = 0 \quad (6)$$

where  $e$  is the internal energy density,  $\mathbf{d}^s$  is the symmetric part of the velocity gradient in the solid skeleton,  $\mathbf{q}$  is the heat flux, and  $h$  is a heat source. This equation may also be written in the alternative form (Hassanizadeh and Gray, 1990)

$$\rho \frac{D^s e}{Dt} - \sigma : \mathbf{d}^s + \nabla \cdot \mathbf{q} - \rho h = 0. \quad (7)$$

## 2.4 Constitutive model

We follow the approach used by Homel, Guilkey, and Brannon (2015) where the total stress is used in the constitutive model instead of the effective stress. In rate form, and after rigid body rotations have been eliminated, the constitutive model has the form

$$\dot{\sigma} = \dot{\sigma}_{\text{eff}} + \dot{\alpha} = \dot{\sigma}_{\text{eff}} - \dot{B} \bar{p}^w \mathbf{I} - B \dot{\bar{p}^w} \mathbf{I} = g_1(\sigma, \eta, \mathbf{d}^s) - g_2(\sigma, \eta, \mathbf{d}^s) \bar{p}^w - g_3(\sigma, \eta, \mathbf{d}^s) B \quad (8)$$

where  $g_1$ ,  $g_2$ , and  $g_3$  are assumed constitutive relations that can be expressed as

$$\begin{aligned} g_1(\sigma, \eta, \mathbf{d}^s) &= \dot{\sigma}_{\text{eff}} = \mathbf{C}_{\text{eff}}^{\text{pe}} : \mathbf{d}^s \\ g_2(\sigma, \eta, \mathbf{d}^s) &= \dot{B} = -\frac{1}{K_s} \dot{\bar{K}}_s + \frac{K_d}{K_s^2} \dot{K}_s = -\frac{1}{K_s B} \left[ \frac{\partial K_d}{\partial \bar{p}^w} - \frac{K_d}{K_s} \frac{\partial K_s}{\partial \bar{p}^w} \right] \text{tr}(\mathbf{d}^s) \\ g_3(\sigma, \eta, \mathbf{d}^s) &= \dot{\bar{p}^w} = \frac{1}{B} \text{tr}(\mathbf{d}^s) \end{aligned} \quad (9)$$

where  $\mathbf{C}_{\text{eff}}^{\text{pe}}(\sigma, \eta)$  is an effective elastic-plastic tangent modulus,  $K_d(\sigma, \eta)$  and  $K_s(\sigma, \eta)$  are the bulk moduli that contribute to the Biot parameter, and  $B(\sigma, \eta)$  is defined in equation (5). Equation (8) suggests that the bulk modulus may also have to be treated as an internal variable if the backstress is treated as an internal variable.

As discussed in the previous report, we assume that

$$\mathbf{d}^s = \mathbf{d}^e + \mathbf{d}^p = \dot{\epsilon}^e + \dot{\epsilon}^p \quad (10)$$

where  $\epsilon^e$  is an elastic strain and  $\epsilon^p$  is an plastic strain. In addition to the plastic strain, we consider the following internal variables ( $\eta$ ) for the partially saturated soil model:

$$\eta = \{\bar{p}_e^w, \bar{p}_p^w, B^e, B^p, \phi^e, \phi^p, S_w^e, S_w^p, \bar{X}^e, \bar{X}^p\} \quad (11)$$

where  $\bar{p}_e^w$  is the part of the pore pressure that goes to zero after a load that leads to plastic deformation is removed,  $\bar{p}_p^w = \bar{p}^w - \bar{p}_e^w$  is the unrecoverable pore pressure which remains nonzero after removal of the load,  $B^e, B^p$  are the the Biot parameter values associated with elastic and inelastic deformation,  $\phi^e$  is the recoverable porosity when the load is removed,  $\phi^p$  is the unrecoverable or “unloaded” porosity,  $S_w^e$  is the recoverable saturation,  $S_w^p$  is the unrecoverable saturation, and  $\bar{X}^e, \bar{X}^p$  are the hydrostatic compressive strengths associated with elastic and inelastic deformation, respectively.

### 2.4.1 Rate-independent plasticity

**Coupled elasticity** We assume that during elastic deformation the elastic behavior of the soil can be coupled to the plastic behavior. The rate equation for elastic behavior can then be written as (see Brannon (2007))

$$\dot{\sigma} = \mathbf{C}^e : \mathbf{d}^e - \dot{\lambda} \mathbf{Z} \quad (12)$$

where the purely elastic stiffness tensor,  $\mathbf{C}^e(\sigma, \bar{p}_e^w, B^e, \phi^e, S_w^e, \bar{X}^e)$ , is defined using

$$\begin{aligned} \mathbf{C}^e : \mathbf{d}^e &= \frac{\partial \sigma}{\partial \epsilon^e} : \mathbf{d}^e + \frac{\partial \sigma}{\partial \bar{p}_e^w} \dot{\bar{p}}_e^w + \frac{\partial \sigma}{\partial B^e} \dot{B}^e + \frac{\partial \sigma}{\partial \phi^e} \dot{\phi}^e + \frac{\partial \sigma}{\partial S_w^e} \dot{S}_w^e + \frac{\partial \sigma}{\partial \bar{X}^e} \dot{\bar{X}}^e \\ &= \left[ \frac{\partial \sigma_{\text{eff}}}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial \bar{p}_e^w} \otimes \frac{\partial \bar{p}_e^w}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial B^e} \otimes \frac{\partial B^e}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial \phi^e} \otimes \frac{\partial \phi^e}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial S_w^e} \otimes \frac{\partial S_w^e}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial \bar{X}^e} \otimes \frac{\partial \bar{X}^e}{\partial \epsilon^e} \right] : \mathbf{d}^e \end{aligned} \quad (13)$$

or

$$\mathbf{C}^e = \frac{\partial \sigma}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial \bar{p}_e^w} \otimes \frac{\partial \bar{p}_e^w}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial B^e} \otimes \frac{\partial B^e}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial \phi^e} \otimes \frac{\partial \phi^e}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial S_w^e} \otimes \frac{\partial S_w^e}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial \bar{X}^e} \otimes \frac{\partial \bar{X}^e}{\partial \epsilon^e}. \quad (14)$$

where  $\epsilon^e$  is the elastic part of the strain that is energy conjugate to the *unrotated* Cauchy stress (see Norris (2008) for possible strain measures). In the plastic coupling term in (12),  $\dot{\lambda}$  is the plastic flow rate parameter, and  $\mathbf{Z}(\sigma, \bar{p}_p^w, B^p, \phi^p, S_w^p, \bar{X}^p; \epsilon^p)$  is a rank-2 elastic-plastic coupling tensor given by

$$\mathbf{Z} = h_p \frac{\partial \sigma}{\partial \bar{p}_p^w} + h_B \frac{\partial \sigma}{\partial B^p} + h_\phi \frac{\partial \sigma}{\partial \phi^p} + h_{S_w} \frac{\partial \sigma}{\partial S_w^p} + h_X \frac{\partial \sigma}{\partial \bar{X}^p} \quad (15)$$

where

$$\dot{\bar{p}}_p^w = \dot{\lambda} h_p, \quad \dot{B}^p = \dot{\lambda} h_B, \quad \dot{\phi}^p = \dot{\lambda} h_\phi, \quad \dot{S}_w^p = \dot{\lambda} h_{S_w}, \quad \dot{\bar{X}}^p = \dot{\lambda} h_X. \quad (16)$$

The “hardening” functions  $h_p, h_B, h_\phi, h_{S_w}$ , and  $h_X$  require additional constitutive assumptions. During purely elastic loading, the coupling term is zero because  $\dot{\lambda}$  is zero.

**Flow rule and consistency** The plastic flow rule is

$$\dot{\epsilon}^p = \mathbf{d}^p = \dot{\lambda} \mathbf{M} \quad (17)$$

where  $\mathbf{M}$  is the unit tensor in the direction of the plastic rate of deformation. The consistency condition can be written as

$$\hat{\mathbf{N}} : \dot{\sigma} = \dot{\lambda} H \quad (18)$$

where  $\hat{\mathbf{N}}$  is the unit normal to the yield surface, and  $H$  is the ensemble hardening modulus.

**The plastic flow rate parameter** To find the consistency parameter, we write the equation for coupled elasticity as

$$\dot{\sigma} = \mathbf{C}^e : (\mathbf{d}^s - \mathbf{d}^p) - \dot{\lambda} \mathbf{Z}. \quad (19)$$

After substituting the flow rule into the above relation and defining

$$\dot{\sigma}_{\text{trial}} := \mathbf{C}^e : \mathbf{d}^s \quad \text{and} \quad \mathbf{P} := \mathbf{C}^e : \mathbf{M} + \mathbf{Z}. \quad (20)$$



we get

$$\dot{\lambda} = \frac{\widehat{\mathbf{N}} : \dot{\boldsymbol{\sigma}}_{\text{trial}}}{\widehat{\mathbf{N}} : \mathbf{P} + H}. \quad (21)$$

Therefore,

$$\dot{\boldsymbol{\sigma}} = \left[ \mathbf{I}^{(s)} - \frac{\mathbf{P} \otimes \widehat{\mathbf{N}}}{\widehat{\mathbf{N}} : \mathbf{P} + H} \right] : \dot{\boldsymbol{\sigma}}_{\text{trial}} =: (\mathbf{C}^p : \mathbf{C}^e) : \mathbf{d}^s = \mathbf{C}^{pe} : \mathbf{d}^s. \quad (22)$$

**Yield condition** For continued plastic loading on the yield surface, the yield condition is

$$f(\boldsymbol{\sigma}, \overline{p}_e^w, \overline{p}_p^w, B^e, B^p, \phi^e, \phi^p, S_w^e, S_w^p, \overline{X}^e, \overline{X}^p; \boldsymbol{\epsilon}^p) = 0 \quad (23)$$

where  $f(\dots)$  is the yield function.

**Consistency condition** The consistency condition implies that for continued plastic loading

$$\dot{f}(\dots) = 0. \quad (24)$$

Assuming appropriate smoothness of  $f(\dots)$ , we have

$$\begin{aligned} \frac{\partial f}{\partial \boldsymbol{\sigma}} : \dot{\boldsymbol{\sigma}} + \frac{\partial f}{\partial \overline{p}_e^w} \dot{\overline{p}}_e^w + \frac{\partial f}{\partial B^e} \dot{B}^e + \frac{\partial f}{\partial \phi^e} \dot{\phi}^e + \frac{\partial f}{\partial S_w^e} \dot{S}_w^e + \frac{\partial f}{\partial \overline{X}^e} \dot{\overline{X}}^e + \\ \dot{\lambda} \left[ h_p \frac{\partial f}{\partial \overline{p}_p^w} + h_B \frac{\partial f}{\partial B^p} + h_\phi \frac{\partial f}{\partial \phi^p} + h_{S_w} \frac{\partial f}{\partial S_w^p} + h_X \frac{\partial f}{\partial \overline{X}^p} \right] = 0. \end{aligned} \quad (25)$$

The normal to the yield surface is

$$\mathbf{N} = \frac{\partial f}{\partial \boldsymbol{\sigma}} + \frac{\partial f}{\partial \overline{p}_e^w} \frac{\partial \overline{p}_e^w}{\partial \boldsymbol{\sigma}} + \frac{\partial f}{\partial B^e} \frac{\partial B^e}{\partial \boldsymbol{\sigma}} + \frac{\partial f}{\partial \phi^e} \frac{\partial \phi^e}{\partial \boldsymbol{\sigma}} + \frac{\partial f}{\partial S_w^e} \frac{\partial S_w^e}{\partial \boldsymbol{\sigma}} + \frac{\partial f}{\partial \overline{X}^e} \frac{\partial \overline{X}^e}{\partial \boldsymbol{\sigma}} \quad (26)$$

and the ensemble hardening modulus is

$$H = - \left[ h_p \frac{\partial f}{\partial \overline{p}_p^w} + h_B \frac{\partial f}{\partial B^p} + h_\phi \frac{\partial f}{\partial \phi^p} + h_{S_w} \frac{\partial f}{\partial S_w^p} + h_X \frac{\partial f}{\partial \overline{X}^p} \right] / \|\mathbf{N}\|. \quad (27)$$

Therefore equation (25) can be written in the consistency condition form

$$\widehat{\mathbf{N}} : \dot{\boldsymbol{\sigma}} = \dot{\lambda} H \quad \text{where} \quad \widehat{\mathbf{N}} = \mathbf{N} / \|\mathbf{N}\|. \quad (28)$$

**Hardening functions** If we assume that

$$\overline{p}_p^w \equiv \overline{p}_p^w(\boldsymbol{\epsilon}^p), \quad B^p \equiv B^p(\boldsymbol{\epsilon}^p), \quad \phi^p \equiv \phi^p(\boldsymbol{\epsilon}^p), \quad S_w^p \equiv S_w^p(\boldsymbol{\epsilon}^p), \quad \overline{X}^p \equiv \overline{X}^p(\boldsymbol{\epsilon}^p) \quad (29)$$

we can write

$$\dot{\overline{p}}_p^w = \frac{d\overline{p}_p^w}{d\boldsymbol{\epsilon}^p} : \mathbf{d}^p, \quad \dot{B}^p = \frac{dB^p}{d\boldsymbol{\epsilon}^p} : \mathbf{d}^p, \quad \dot{\phi}^p = \frac{d\phi^p}{d\boldsymbol{\epsilon}^p} : \mathbf{d}^p, \quad \dot{S}_w^p = \frac{dS_w^p}{d\boldsymbol{\epsilon}^p} : \mathbf{d}^p, \quad \dot{\overline{X}}^p = \frac{d\overline{X}^p}{d\boldsymbol{\epsilon}^p} : \mathbf{d}^p. \quad (30)$$

Using the flow rule (17), we have

$$\dot{\overline{p}}_p^w = \dot{\lambda} \frac{d\overline{p}_p^w}{d\boldsymbol{\epsilon}^p} : \mathbf{M}, \quad \dot{B}^p = \dot{\lambda} \frac{dB^p}{d\boldsymbol{\epsilon}^p} : \mathbf{M}, \quad \dot{\phi}^p = \dot{\lambda} \frac{d\phi^p}{d\boldsymbol{\epsilon}^p} : \mathbf{M}, \quad \dot{S}_w^p = \dot{\lambda} \frac{dS_w^p}{d\boldsymbol{\epsilon}^p} : \mathbf{M}, \quad \dot{\overline{X}}^p = \dot{\lambda} \frac{d\overline{X}^p}{d\boldsymbol{\epsilon}^p} : \mathbf{M}. \quad (31)$$

Therefore, comparing the above with (16), we have

$$h_p = \frac{d\bar{p}_p^w}{d\epsilon^p} : \mathbf{M}, \quad h_B = \frac{dB^p}{d\epsilon^p} : \mathbf{M}, \quad h_\phi = \frac{d\phi^p}{d\epsilon^p} : \mathbf{M}, \quad h_{S_w} = \frac{dS_w^p}{d\epsilon^p} : \mathbf{M}, \quad h_X = \frac{d\bar{X}^p}{d\epsilon^p} : \mathbf{M}. \quad (32)$$

Noting that the derivative of the trace of a rank-2 tensor with respect to the tensor is equal to the rank-2 identity tensor, we have

$$h_p = \frac{d\bar{p}_p^w}{d\epsilon_v^p} \text{tr}(\mathbf{M}), \quad h_B = \frac{dB^p}{d\epsilon_v^p} \text{tr}(\mathbf{M}), \quad h_\phi = \frac{d\phi^p}{d\epsilon_v^p} \text{tr}(\mathbf{M}), \quad h_{S_w} = \frac{dS_w^p}{d\epsilon_v^p} \text{tr}(\mathbf{M}), \quad h_X = \frac{d\bar{X}^p}{d\epsilon_v^p} \text{tr}(\mathbf{M}). \quad (33)$$

Expressions that have this form are derived later in this report.

**Yield function** If the volumetric and deviatoric components of the total stress are

$$\bar{p} := -\frac{1}{3}\text{tr}(\boldsymbol{\sigma}) \quad \text{and} \quad \mathbf{s} := \boldsymbol{\sigma} + \bar{p}\mathbf{I} \quad (34)$$

we can define

$$\bar{p}_{\text{eff}} := -\frac{1}{3}\text{tr}(\boldsymbol{\sigma}_{\text{eff}}) = \bar{p} - B\bar{p}^w, \quad \mathbf{s}_{\text{eff}} := \boldsymbol{\sigma}_{\text{eff}} + \bar{p}_{\text{eff}}\mathbf{I} = \boldsymbol{\sigma} + \bar{p}\mathbf{I} = \mathbf{s} \quad \text{and} \quad J_2^{\text{eff}} := \frac{1}{2}\mathbf{s}_{\text{eff}} : \mathbf{s}_{\text{eff}}. \quad (35)$$

Then the Arena yield function is

$$f(\boldsymbol{\sigma}, B, \bar{p}^w, \bar{X}) = \beta \sqrt{J_2^{\text{eff}}} - F_f(\bar{p}_{\text{eff}}) F_c(\bar{p}_{\text{eff}}, \bar{X}) \quad (36)$$

where

$$F_f(\bar{p}_{\text{eff}}) = a_1 - a_3 \exp[-3a_2\bar{p}_{\text{eff}}] + 3a_4\bar{p}_{\text{eff}} \quad (37)$$

and

$$F_c(\bar{p}_{\text{eff}}, \bar{X}) = \begin{cases} 1 & \text{for } 3\bar{p}_{\text{eff}} \leq \bar{\kappa} \\ \sqrt{1 - \left(\frac{3\bar{p}_{\text{eff}} - \bar{\kappa}}{\bar{X}_{\text{eff}} - \bar{\kappa}}\right)^2} & \text{for } 3\bar{p}_{\text{eff}} > \bar{\kappa}. \end{cases} \quad (38)$$

Here  $a_i$  are material parameters,  $\bar{X}_{\text{eff}}(\epsilon^p, B, \bar{p}^w) = \bar{X} - 3B\bar{p}^w$  is the shifted form of the apparent hydrostatic compressive strength ( $\bar{X}/3$ ) of the partially saturated material, and  $\bar{\kappa}$  is the branch point at which the cap function  $F_c$  starts decreasing until it reaches the hydrostatic strength point ( $\bar{X}$ ):

$$\bar{\kappa} = 3\bar{p}_{\text{eff}}^{\text{peak}} - (3\bar{p}_{\text{eff}}^{\text{peak}} - \bar{X}_{\text{eff}})R_c \quad (39)$$

where  $\bar{p}_{\text{eff}}^{\text{peak}}$  is the maximum hydrostatic tensile stress that the material can support and  $R_c$  is a cap ratio. Non-associativity is modeled using the parameter  $\beta$  that modifies  $\sqrt{J_2^{\text{eff}}}$ .

**Hydrostatic compressive strength** The drained hydrostatic compressive strength ( $\bar{X}_d/3$ ) is found from the empirical drained material crush curve using

$$\bar{X}_d(\bar{\epsilon}_v^p, \phi_0) - p_0 = p_1 \left[ \frac{1 - \exp(-p_3)}{1 - \exp(-p_3 + \bar{\epsilon}_v^p)} - 1 \right]^{1/p_2}, \quad p_3 := -\ln(1 - \phi_0) \quad (40)$$

where  $\bar{\epsilon}_v^p = -\text{tr}(\epsilon^p)$  is the volumetric plastic strain,  $\bar{X}_d/3$  is the drained hydrostatic compressive strength,  $p_0, p_1, p_2, p_3$  are model parameters, and  $\phi_0$  is the initial porosity. The derivative of  $\bar{X}_d$  is

$$\frac{d\bar{X}_d}{d\epsilon_v^p} = \frac{p_1[1 - \exp(-p_3)] \exp(-p_3 + \bar{\epsilon}_v^p)}{p_2[1 - \exp(-p_3 + \bar{\epsilon}_v^p)]^2} \left[ \frac{1 - \exp(-p_3)}{1 - \exp(-p_3 + \bar{\epsilon}_v^p)} - 1 \right]^{1/p_2 - 1}. \quad (41)$$

The effective hydrostatic compressive strength for a partially saturated material is expected to be different from the drained value (we do not have any direct experimental data which supports that conjecture). We follow the approach used by Grujicic et al. (2009) and assume a model of the form

$$\bar{X}_{\text{eff}}(\bar{\varepsilon}_v^p, \phi_0, S_w) - p_0 = [(1 - S_w) + p_1^{\text{sat}} S_w] (\bar{X}_d - p_0) ; \quad \bar{X} = \bar{X}_{\text{eff}} + 3B\bar{p}^w \quad (42)$$

where  $p_1 \times p_1^{\text{sat}}$  is the value of  $p_1$  in a fully saturated material. **Note that the Jan 2016 monthly report has results that suggest that  $p_1^{\text{sat}} < 1$ .**

**Porosity, saturation, and volumetric strain** Recall that the saturation is defined as

$$S_w = \frac{v^w}{v^a + v^w} = 1 - S_a \quad \implies \quad \frac{v^a}{v^w} = \frac{1 - S_w}{S_w} \quad (43)$$

where  $v^\alpha$  is the volume occupied by phase  $\alpha$  in the pore volume. Also recall that the porosity is defined as

$$\phi = \frac{v^a + v^w}{v^s + v^a + v^w} \quad \implies \quad 1 - \phi = \frac{v^s}{v^s + v^a + v^w} . \quad (44)$$

The volumetric strain in each phase is defined as

$$\varepsilon_v^\alpha = \ln \left( \frac{v^\alpha}{v_0^\alpha} \right) \quad \text{where} \quad \alpha = \{s, w, a\} \quad (45)$$

where  $v_0^\alpha$  is the initial volume of phase  $\alpha$ . The total volumetric strain of the mixture is

$$\exp(\varepsilon_v) = (1 - S_0)\phi_0 \exp(\varepsilon_v^a) + S_0\phi_0 \exp(\varepsilon_v^w) + (1 - \phi_0) \exp(\varepsilon_v^s) \quad (46)$$

where  $S_0$  is the initial saturation and  $\phi_0$  is the initial porosity. With the assumption that the total volumetric strain can be additively decomposed into elastic and plastic parts, we have

$$\varepsilon_v = \varepsilon_v^e + \varepsilon_v^p . \quad (47)$$

Therefore,

$$\exp(\varepsilon_v^p) = (1 - S_0)\phi_0 \frac{\exp(\varepsilon_v^a)}{\exp(\varepsilon_v^e)} + S_0\phi_0 \frac{\exp(\varepsilon_v^w)}{\exp(\varepsilon_v^e)} + (1 - \phi_0) \frac{\exp(\varepsilon_v^s)}{\exp(\varepsilon_v^e)} . \quad (48)$$

**Saturation equation** The saturation equation when  $\bar{p}^w = \bar{p}^a$  is

$$S_w(\bar{p}^w) = \frac{C}{1 - S_0 + C} , \quad C := S_0 \exp [\bar{\varepsilon}_v^a(\bar{p}^w) - \bar{\varepsilon}_v^w(\bar{p}^w)] . \quad (49)$$

For a soil that has undergone compressive loading leading to plastic volumetric deformation, the continuity of the phases implies that, after unloading, the saturation is different from the initial value of saturation even when the hydraulic conductivity of the material is zero. Equation (49) then implies that the residual saturation is a function non-zero strains in the water and air phases. Because the air and the water are assumed to be perfectly elastic, the saturation is a function of a residual pressure which can be thought of as a “plastic” pore-pressure which is a function of the plastic volumetric strain.

One way of defining the “plastic” pore pressure can be that the unrecoverable part of the saturation is

$$S_w^p(\bar{p}_p^w) = \frac{C^p}{1 - S_0 + C^p} , \quad C^p := S_0 \exp [\bar{\varepsilon}_v^a(\bar{p}_p^w) - \bar{\varepsilon}_v^w(\bar{p}_p^w)] . \quad (50)$$

**Porosity equation** Recall that the porosity is given by

$$1 - \phi = \frac{v^s}{v} \implies \frac{1 - \phi}{1 - \phi_0} = \frac{v^s}{v_0^s} \frac{v_0}{v} = \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^s) = \frac{\exp(\varepsilon_v^s)}{\exp(\varepsilon_v)}. \quad (51)$$

If we know the total volumetric strain and the volumetric strains in the fluids, we can use equation (46) to write the above as

$$1 - \phi = \frac{\exp(\varepsilon_v) - (1 - S_0)\phi_0 \exp(\varepsilon_v^a) - S_0\phi_0 \exp(\varepsilon_v^w)}{\exp(\varepsilon_v)} \quad (52)$$

When the pore water and air pressure are equal to each other, and also equal to the intrinsic pressure in the solid grains), the porosity equation (52) can be expressed as

$$\phi(\bar{p}^w) = (1 - S_0)\phi_0 \exp[\bar{\varepsilon}_v(\bar{p}^w) - \bar{\varepsilon}_v^a(\bar{p}^w)] + S_0\phi_0 \exp[\bar{\varepsilon}_v(\bar{p}^w) - \bar{\varepsilon}_v^w(\bar{p}^w)] \quad (53)$$

The primary cause of the residual pressure in a soil that has undergone volumetric plastic deformation is the irreversible change in porosity. Let the unrecoverable part of the porosity (called the “unloaded” porosity by Brannon, 2007) be  $\phi^p$ . We could define the “unloaded” or “plastic” porosity as

$$\phi^p(\bar{p}_p^w) = (1 - S_0)\phi_0 \exp[\bar{\varepsilon}_v(\bar{p}_p^w) - \bar{\varepsilon}_v^a(\bar{p}_p^w)] + S_0\phi_0 \exp[\bar{\varepsilon}_v(\bar{p}_p^w) - \bar{\varepsilon}_v^w(\bar{p}_p^w)] \quad (54)$$

**Pore pressure evolution** As discussed in the March-April 2016 report, an expression for the evolution of the pore pressure can be found from the mass balance equations:

$$\dot{\bar{p}}^w = \frac{1}{\mathcal{B}} \text{tr}(\mathbf{d}^s) \quad (55)$$

where  $\mathcal{B} \equiv \mathcal{B}(\bar{p}^w, B, \phi, S_w)$  is given in equation (5). If we express the above equation in terms of elastic and plastic components, we have

$$\dot{\bar{p}}_e^w + \dot{\bar{p}}_p^w = \frac{1}{\mathcal{B}} [\text{tr}(\mathbf{d}^e) + \text{tr}(\mathbf{d}^p)]. \quad (56)$$

We define the “elastic” and “plastic” pore pressure evolution as

$$\dot{\bar{p}}_e^w := \frac{1}{\mathcal{B}} \text{tr}(\mathbf{d}^e) \quad \text{and} \quad \dot{\bar{p}}_p^w := \frac{1}{\mathcal{B}} \text{tr}(\mathbf{d}^p) = \frac{\dot{\lambda}}{\mathcal{B}} \text{tr}(\mathbf{M}). \quad (57)$$

**Biot ratio evolution** If we assume that  $\bar{p}^s = \bar{p}^w = \bar{p}^a$ , the material time derivative of the Biot ratio (3) is

$$\dot{B} := -\frac{1}{K_s} \left( \dot{\bar{K}}_s - \frac{K_d}{K_s} \dot{K}_s \right) = -\frac{1}{K_s} \left( \frac{dK_d}{d\bar{p}^w} - \frac{K_d}{K_s} \frac{dK_s}{d\bar{p}^w} \right) \dot{\bar{p}}^w. \quad (58)$$

We can define “elastic” and “plastic” parts of the rate of the Biot ratio as  $\dot{B} = \dot{B}^e + \dot{B}^p$  where

$$\dot{B}^e := \frac{1}{BK_s} \left( \frac{K_d}{K_s} \frac{dK_d}{d\bar{p}^w} - \frac{dK_s}{d\bar{p}^w} \right) \text{tr}(\mathbf{d}^e) \quad \text{and} \quad \dot{B}^p := \dot{\lambda} \left[ \frac{1}{BK_s} \left( \frac{K_d}{K_s} \frac{dK_d}{d\bar{p}^w} - \frac{dK_s}{d\bar{p}^w} \right) \right] \text{tr}(\mathbf{M}). \quad (59)$$

**Saturation evolution** From equation (49) we have

$$\dot{S}_w = \frac{(1-S_0)\dot{C}}{(1-S_0+C)^2} = \frac{1-S_0}{(1-S_0+C)^2} \frac{dC}{d\bar{p}^w} \dot{\bar{p}}^w = \frac{1-S_0}{(1-S_0+C)^2} \frac{dC}{d\bar{p}^w} \left( \dot{\bar{p}}^w + \frac{\lambda}{B} \text{tr}(\mathbf{M}) \right). \quad (60)$$

We can express the above as  $\dot{S}_w = \dot{S}_w^e + \dot{S}_w^p$ , where the evolution equations for the “elastic” and “plastic” parts of the saturation are

$$\dot{S}_w^e := \left[ \frac{1-S_0}{B(1-S_0+C)^2} \frac{dC}{d\bar{p}^w} \right] \text{tr}(\mathbf{d}^e) \quad \text{and} \quad \dot{S}_w^p := \lambda \left[ \frac{1-S_0}{B(1-S_0+C)^2} \frac{dC}{d\bar{p}^w} \right] \text{tr}(\mathbf{M}). \quad (61)$$

**Porosity evolution** If we take the material time derivative of equation (53), we get

$$\dot{\phi} = (1-S_0)\phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) \left( \dot{\bar{\varepsilon}}_v - \frac{d\bar{\varepsilon}_v^a}{d\bar{p}^w} \dot{\bar{p}}^w \right) + S_0\phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \left( \dot{\bar{\varepsilon}}_v - \frac{d\bar{\varepsilon}_v^w}{d\bar{p}^w} \dot{\bar{p}}^w \right) \quad (62)$$

or

$$\dot{\phi} = - \left[ (1-S_0)\phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) \left( 1 + \frac{1}{K_a B} \right) + S_0\phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \left( 1 + \frac{1}{K_w B} \right) \right] \text{tr}(\mathbf{d}^s). \quad (63)$$

We can express the above as  $\dot{\phi} = \dot{\phi}^e + \dot{\phi}^p$ , where

$$\begin{aligned} \dot{\phi}^e &= - \left[ (1-S_0)\phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) \left( 1 + \frac{1}{K_a B} \right) + S_0\phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \left( 1 + \frac{1}{K_w B} \right) \right] \text{tr}(\mathbf{d}^e) \\ \dot{\phi}^p &= - \lambda \left[ (1-S_0)\phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) \left( 1 + \frac{1}{K_a B} \right) + S_0\phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \left( 1 + \frac{1}{K_w B} \right) \right] \text{tr}(\mathbf{M}). \end{aligned} \quad (64)$$

**Hydrostatic compressive strength evolution** The rate form of the hydrostatic compressive strength equation (42) is

$$\begin{aligned} \dot{\bar{X}} &= \dot{\bar{X}}_{\text{eff}} + 3(\dot{B}\bar{p}^w + B\dot{\bar{p}}^w) \\ &= [(1-S_w) + p_1^{\text{sat}} S_w] \dot{\bar{X}}_d + (p_1^{\text{sat}} - 1) \dot{S}_w \bar{X}_d + 3(\dot{B}\bar{p}^w + B\dot{\bar{p}}^w) \\ &= [(1-S_w) + p_1^{\text{sat}} S_w] \frac{d\bar{X}_d}{d\bar{\varepsilon}_v^p} \text{tr}(\mathbf{d}^p) + (p_1^{\text{sat}} - 1) (\dot{S}_w^e + \dot{S}_w^p) \bar{X}_d + 3[\dot{B}\bar{p}^w + B(\dot{\bar{p}}_p^w + \dot{\bar{p}}_e^w)] \\ &= [(1-S_w) + p_1^{\text{sat}} S_w] \frac{d\bar{X}_d}{d\bar{\varepsilon}_v^p} [\lambda \text{tr}(\mathbf{M})] + \left[ \frac{\bar{X}_d(p_1^{\text{sat}} - 1)(1-S_0)}{B(1-S_0+C)^2} \frac{dC}{d\bar{p}^w} \right] [\text{tr}(\mathbf{d}^e) + \lambda \text{tr}(\mathbf{M})] + \\ &\quad 3 \left[ \frac{\bar{p}^w}{BK_s} \left( \frac{K_d}{K_s} \frac{dK_d}{d\bar{p}^w} - \frac{dK_s}{d\bar{p}^w} \right) [\text{tr}(\mathbf{d}^e) + \lambda \text{tr}(\mathbf{M})] + \frac{B}{B} \text{tr}(\mathbf{d}^e) + \frac{\lambda B}{B} \text{tr}(\mathbf{M}) \right] \\ &= \left[ \frac{\bar{X}_d(p_1^{\text{sat}} - 1)(1-S_0)}{B(1-S_0+C)^2} \frac{dC}{d\bar{p}^w} + \frac{3\bar{p}^w}{BK_s} \left( \frac{K_d}{K_s} \frac{dK_d}{d\bar{p}^w} - \frac{dK_s}{d\bar{p}^w} \right) + \frac{3B}{B} \right] \text{tr}(\mathbf{d}^e) + \\ &\quad \lambda \left[ [(1-S_w) + p_1^{\text{sat}} S_w] \frac{d\bar{X}_d}{d\bar{\varepsilon}_v^p} + \frac{\bar{X}_d(p_1^{\text{sat}} - 1)(1-S_0)}{B(1-S_0+C)^2} \frac{dC}{d\bar{p}^w} + \frac{3\bar{p}^w}{BK_s} \left( \frac{K_d}{K_s} \frac{dK_d}{d\bar{p}^w} - \frac{dK_s}{d\bar{p}^w} \right) + \frac{3B}{B} \right] \text{tr}(\mathbf{M}). \end{aligned} \quad (65)$$

As before, we could define “elastic” and “plastic” parts such that  $\dot{\bar{X}} = \dot{\bar{X}}^e + \dot{\bar{X}}^p$ , where

$$\begin{aligned}\dot{\bar{X}}^e &:= \left[ \frac{\bar{X}_d(p_1^{\text{sat}} - 1)(1 - S_0)}{\mathcal{B}(1 - S_0 + C)^2} \frac{dC}{d\bar{p}^w} + \frac{3\bar{p}^w}{\mathcal{B}K_s} \left( \frac{K_d}{K_s} \frac{dK_d}{d\bar{p}^w} - \frac{dK_s}{d\bar{p}^w} \right) + \frac{3B}{\mathcal{B}} \right] \text{tr}(\mathbf{d}^e) \\ \dot{\bar{X}}^p &:= \dot{\lambda} \left[ \left[ (1 - S_w) + p_1^{\text{sat}} S_w \right] \frac{d\bar{X}_d}{d\bar{\varepsilon}_v^p} + \right. \\ &\quad \left. \frac{\bar{X}_d(p_1^{\text{sat}} - 1)(1 - S_0)}{\mathcal{B}(1 - S_0 + C)^2} \frac{dC}{d\bar{p}^w} + \frac{3\bar{p}^w}{\mathcal{B}K_s} \left( \frac{K_d}{K_s} \frac{dK_d}{d\bar{p}^w} - \frac{dK_s}{d\bar{p}^w} \right) + \frac{3B}{\mathcal{B}} \right] \text{tr}(\mathbf{M}).\end{aligned}\quad (66)$$

## 2.4.2 Rate-dependent plasticity

Rate-dependent plastic effects are determined after the rate-independent solution has been found. An additive decomposition of the (unrotated) rate of deformation into elastic and viscoplastic parts is assumed, i.e.,

$$\mathbf{d}^s = \mathbf{d}^e + \mathbf{d}^{\text{vp}}. \quad (67)$$

The Perzyna formulation for the viscoplastic rate of deformation is used (Brannon, 2007):

$$\mathbf{d}^{\text{vp}} = \left\langle \frac{f(\boldsymbol{\sigma}, \boldsymbol{\eta})}{\tau} \right\rangle = \begin{cases} \frac{f(\boldsymbol{\sigma}, \boldsymbol{\eta})}{\tau} \mathbf{M} & \text{if } f(\boldsymbol{\sigma}, \boldsymbol{\eta}) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (68)$$

where  $f(\dots)$  is the yield function and  $\tau$  is a relaxation time. Note that the above relation can be expressed in Duvaut-Lion form as

$$\mathbf{d}^{\text{vp}} = \begin{cases} (\mathbf{C}^e)^{-1} : \left( \frac{\boldsymbol{\sigma} - \mathcal{P}\boldsymbol{\sigma}}{\tau} \right) & \text{if } f(\boldsymbol{\sigma}, \boldsymbol{\eta}) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (69)$$

where  $\mathbf{C}^e$  is the elastic stiffness tensor and  $\mathcal{P}\boldsymbol{\sigma}$  is the closest point projection of the stress state on to the boundary of the region that bounds all possible elastic stress states. Clearly,  $\mathcal{P}\boldsymbol{\sigma} = \boldsymbol{\sigma}_{qs}$ , and for stress states outside the yield surface we have

$$\mathbf{d}^{\text{vp}} = \frac{1}{\tau} (\mathbf{C}^e)^{-1} : (\boldsymbol{\sigma} - \boldsymbol{\sigma}_{qs}). \quad (70)$$

Recall from equation (12) that

$$\dot{\boldsymbol{\sigma}} = \mathbf{C}^e : \mathbf{d}^e - \dot{\lambda} \mathbf{Z}. \quad (71)$$

If we ignore the  $\mathbf{Z}$  term and use equations (20) and (70), we have

$$\dot{\boldsymbol{\sigma}} = \mathbf{C}^e : (\mathbf{d}^s - \mathbf{d}^{\text{vp}}) = \mathbf{C}^e : \mathbf{d}^s - \frac{1}{\tau} (\boldsymbol{\sigma} - \boldsymbol{\sigma}_{qs}) = \dot{\boldsymbol{\sigma}}^{\text{trial}} - \frac{1}{\tau} (\boldsymbol{\sigma} - \boldsymbol{\sigma}_{qs}). \quad (72)$$

We can subtract the rate of the quasistatic stress from both sides to get an equation for the viscoplastic overstress contribution ( $\boldsymbol{\sigma}_{\text{over}} := \boldsymbol{\sigma} - \boldsymbol{\sigma}_{qs}$ ):

$$\dot{\boldsymbol{\sigma}} - \dot{\boldsymbol{\sigma}}_{qs} = \dot{\boldsymbol{\sigma}}^{\text{trial}} - \dot{\boldsymbol{\sigma}}_{qs} - \frac{1}{\tau} (\boldsymbol{\sigma} - \boldsymbol{\sigma}_{qs}) \quad (73)$$

or

$$\dot{\boldsymbol{\sigma}}_{\text{over}} = (\dot{\boldsymbol{\sigma}}^{\text{trial}} - \dot{\boldsymbol{\sigma}}_{qs}) - \frac{1}{\tau} \boldsymbol{\sigma}_{\text{over}}. \quad (74)$$

From equation (22) we have

$$\dot{\boldsymbol{\sigma}}_{qs} = \dot{\boldsymbol{\sigma}}^{\text{trial}} - \frac{\mathbf{P} \otimes \widehat{\mathbf{N}}}{\widehat{\mathbf{N}} : \mathbf{P} + H} : \dot{\boldsymbol{\sigma}}^{\text{trial}}. \quad (75)$$

Therefore, (74) can be written as

$$\dot{\sigma}_{\text{over}} = \frac{\mathbf{P} \otimes \hat{\mathbf{N}}}{\hat{\mathbf{N}} : \mathbf{P} + H} : \dot{\sigma}_{\text{trial}} - \frac{1}{\tau} \sigma_{\text{over}} =: \dot{\sigma}_{\text{proj}} - \frac{1}{\tau} \sigma_{\text{over}} \quad (76)$$

where

$$\dot{\sigma}_{\text{proj}} := \dot{\sigma}^{\text{trial}} - \dot{\sigma}_{qs} = \frac{\mathbf{P} \otimes \hat{\mathbf{N}}}{\hat{\mathbf{N}} : \mathbf{P} + H} : \dot{\sigma}^{\text{trial}}. \quad (77)$$

The above ordinary differential equation has the solution

$$\sigma_{\text{over}}(t) = \frac{\int \exp\left[\int^t \frac{1}{\tau(t')} dt'\right] \dot{\sigma}_{\text{proj}}(t) dt + C}{\exp\left[\int^t \frac{1}{\tau(t')} dt'\right]} \quad (78)$$

where  $C$  is a constant. If we assume that  $\tau$  is constant over a timestep, we have

$$\sigma_{\text{over}}(t) = \frac{\int \exp\left[\frac{t}{\tau}\right] \dot{\sigma}_{\text{proj}}(t) dt + C}{\exp\left[\frac{t}{\tau}\right]} \quad (79)$$

In addition, if we assume that  $\dot{\sigma}_{\text{proj}}$  is also constant over a timestep, we have

$$\sigma_{\text{over}}(t) = \frac{\tau \exp\left[\frac{t}{\tau}\right] \dot{\sigma}_{\text{proj}} + C}{\exp\left[\frac{t}{\tau}\right]}. \quad (80)$$

If the initial overstress at  $t = t_0$  is  $\sigma_{\text{over}}(t_0)$ , we have

$$C = \exp\left[\frac{t_0}{\tau}\right] \sigma_{\text{over}}(t_0) - \tau \exp\left[\frac{t_0}{\tau}\right] \dot{\sigma}_{\text{proj}}. \quad (81)$$

Therefore

$$\sigma_{\text{over}}(t) = \tau \left[ 1 - \exp\left(\frac{t_0 - t}{\tau}\right) \right] \dot{\sigma}_{\text{proj}} + \exp\left(\frac{t_0 - t}{\tau}\right) \sigma_{\text{over}}(t_0). \quad (82)$$

The model used for the material relaxation time in Arensica is

$$\tau = T_1 \dot{\epsilon}^{-T_2} \quad \text{where} \quad \dot{\epsilon} = \|\mathbf{d}^s\| \quad (83)$$

and  $T_1, T_2$  are material parameters.

### 3 Arena Partially Saturated Soil Model: Computing the stress

The stress tensor computation procedure calls the COMPUTESTRESSTensor routine that is specific to each constitutive model.

---

#### Algorithm 1 Computing the stress tensor

---

**Require:**  $\Delta t, \mathbf{x}_p^n, m_p, V_p^{n+1}, \mathbf{h}_p^n, \mathbf{l}_p^{n+1}, \mathbf{F}_p^{n+1}, \boldsymbol{\sigma}_p^n, \boldsymbol{\eta}_p^n, \rho_0, \text{materialList}, \text{mpmFlags}, \text{constitutiveModel}$

```

1: procedure COMPUTESTRESSTensor
2:   for  $\text{matl}$  in  $\text{materialList}$  do
3:      $\boldsymbol{\sigma}_p^{n+1}, \boldsymbol{\eta}_p^{n+1} \leftarrow \text{constitutiveModel}[\text{matl}].\text{COMPUTESTRESSTensor}(\Delta t, \mathbf{x}_p^n, m_p, V_p^{n+1}, \mathbf{h}_p^n,$ 
        $\hookrightarrow \mathbf{l}_p^{n+1}, \mathbf{F}_p^{n+1}, \boldsymbol{\sigma}_p^n, \boldsymbol{\eta}_p^n, \rho_0, \text{mpmFlags})$   $\triangleright$  Update the stress and any internal variables
        $\hookrightarrow$  needed by the constitutive model
4:   end for
5:   return  $\boldsymbol{\sigma}_p^{n+1}, \boldsymbol{\eta}_p^{n+1}$ 
6: end procedure

```

---

#### 3.1 Initialization of the model

The model is initialized in two steps. In the first step, the constitutive model object is created followed by initialization of the stress (and the deformation gradient if needed). Here

- $\phi_0$  : The initial porosity.
- $S_0$  : The initial porosity.
- $n_{\max}$  : The maximum number of subcycles in the plastic return algorithm.
- $\varepsilon_{v,p}^{e,n}$  : The elastic volumetric strain at a particle at  $t = t_n$ .
- $\boldsymbol{\sigma}_p^n$  : The dynamic Cauchy stress at a particle at  $t = t_n$ .
- $\boldsymbol{\sigma}_{qs,p}^n$  : The quasistatic Cauchy stress at a particle at  $t = t_n$ .
- $\boldsymbol{\sigma}^0$  : The initial Cauchy stress at a particle.

---

#### Algorithm 2 Creating the ArenaPartiallySaturated constitutive model object

---

**Require:**  $\text{mpmFlags}, \text{xmlProblemSpec}$

```

1: procedure CREATECONSTITUTIVEMODEL
2:    $\text{elasticityModel} \leftarrow \text{ElasticModuliModelFactory}.\text{CREATE}(\text{xmlProblemSpec})$ 
3:    $\text{yieldCondition} \leftarrow \text{YieldConditionFactory}.\text{CREATE}(\text{xmlProblemSpec})$ 
4:    $p_0, p_1, \bar{p}_1^{\text{sat}}, p_2 \leftarrow \text{READCRUSHCURVEPARAMETERS}(\text{xmlProblemSpec})$ 
5:    $\phi_0, S_0, \bar{p}_0^w \leftarrow \text{READINITIALPOROSITYANDSATURATION}(\text{xmlProblemSpec})$ 
6:    $n_{\max} \leftarrow \text{READSUBCYCLINGCHARACTERISTICNUMBER}(\text{xmlProblemSpec})$ 
7:   return  $\text{elasticityModel}, \text{yieldCondition}, \phi_0, S_0, \bar{p}_0^w, n_{\max}, p_0, p_1, \bar{p}_1^{\text{sat}}, p_2$ 
8: end procedure

```

---



---

#### Algorithm 3 Initializing the ArenaPartiallySaturated particle variables

---

**Require:**  $\phi_0, S_0, \bar{p}_0^w, \text{particleList}, V_p^0, m_p^0, \mathbf{v}_p^0, \text{fluidParams}, \text{elasticityModel}, \text{yieldCondition}$

```

1: procedure INITIALIZE
2:    $\text{yieldCondition}.\text{INITIALIZE}(\text{particleList}, V_p^0)$ 
3:    $\text{yieldParams} \leftarrow \text{yieldCondition}.\text{GETPARAMETERS}()$ 
4:   for  $\text{part}$  in  $\text{particleList}$  do
5:      $\phi_p^0[\text{part}] \leftarrow \phi_0$ 
6:      $S_{w,p}^0[\text{part}] \leftarrow S_0$ 
7:      $\chi_p^0[\text{part}] \leftarrow 0$ 
8:      $\varepsilon_{v,p}^{e,0}[\text{part}] \leftarrow 0$ 

```

---



```

9:       $\sigma_p^0[\text{part}] \leftarrow (\text{fluidParams}.\overline{p}_0^w) I$ 
10:      $\sigma_{qs,p}^0[\text{part}] \leftarrow \sigma^0$ 
11:   end for
12:    $\Delta t \leftarrow \text{COMPUTESTABLETIMESTEP}(V_p^0, m_p, \mathbf{v}_p^0, \text{elasticityModel})$ 
13:   return  $\phi_p^0, S_{w,p}^0, \chi_p^0, \varepsilon_{v,p}^0, \sigma_{qs,p}^0, \sigma_p^0, \Delta t$ 
14: end procedure

```

### 3.2 Computing the stress and internal variables

The COMPUTESTRESS TENSOR routine in the partially saturated Arena model assumes that the Biot coefficient is  $B = 1$ , and has the following form. Here we introduce the new variables

- $\phi_p^n, \phi_p^{n+1}$  : The porosity at  $t = t_n$  and  $t = t_{n+1}$ .
- $S_{w,p}^n, S_{w,p}^{n+1}$  : The saturation at  $t = t_n$  and  $t = t_{n+1}$ .
- $a_{1,p}, a_{2,p}, a_{3,p}, a_{4,p}$  : Yield condition parameters at each particle.
- $p_{3,p}^n$  : The particle crush curve parameter  $p_3$  at  $t = t_n$ .
- $X_p^n$  : The particle hydrostatic compressive strength at  $t = t_n$ .
- $\kappa_p^n$  : The yield function branch point at  $t = t_n$ .
- $\varepsilon_p^{p,n}$  : The particle plastic strain tensor at  $t = t_n$ .
- $\alpha_p^n$  : The particle backstress tensor at  $t = t_n$ .
- $\mathbf{d}^n$  : The particle rate of deformation tensor at  $t = t_n$ .
- $\mathbf{R}^n, \mathbf{U}^n$  : The particle rotation and stretch tensors at  $t = t_n$ .
- $K^n, G^n$  : The particle bulk and shear modulus at  $t = t_n$ .

#### Algorithm 4 Computing the Arena partially saturated stress tensor

**Require:**  $\Delta t, \mathbf{x}_p^n, m_p, V_p^{n+1}, \mathbf{h}_p^n, \mathbf{l}_p^{n+1}, \mathbf{F}_p^n, \mathbf{F}_p^{n+1}, X_p^n, \kappa_p^n, \varepsilon_{v,p}^n, p_{3,p}^n, \varepsilon_p^{p,n}, \alpha_p^n, \phi_p^n, S_{w,p}^n, \chi_p^n, \varepsilon_{v,p}^n, \sigma_{qs,p}^n, \sigma_p^n, \rho_0, \text{particleList}, \text{mpmFlags}, \text{elasticityModel}, \text{yieldCondition}$

```

1: procedure COMPUTESTRESS TENSOR
2:   yieldParams  $\leftarrow$  yieldCondition.GETPARAMETERS()
3:    $a_{1,p}, a_{2,p}, a_{3,p}, a_{4,p}, I_{1,p}^{\text{peak}}, R_{c,p} \leftarrow$  yieldCondition.GETLOCALVARIABLES()
    $\hookrightarrow$   $\triangleright$ Yield condition parameters vary at each particle.
    $\hookrightarrow$  Get the per-particle values of these parameters.
4:   for part in particleList do
5:      $\chi_p^{n+1}[\text{part}] \leftarrow \chi_p^n[\text{part}]$   $\triangleright$ Copy over failure indicator variable
6:      $\mathbf{d}^{n+1} \leftarrow [\mathbf{l}_p^{n+1}[\text{part}] + (\mathbf{l}_p^{n+1}[\text{part}])^T]/2$   $\triangleright$ Compute rate of deformation
7:      $\mathbf{R}^n, \mathbf{U}^n \leftarrow \text{POLARDECOMPOSITION}(\mathbf{F}_p^n[\text{part}])$   $\triangleright$ Compute rotation and stretch tensors
8:      $\mathbf{d}_{\text{unrot}}^{n+1} \leftarrow (\mathbf{R}^n)^T \cdot \mathbf{d}^{n+1} \cdot \mathbf{R}^n$   $\triangleright$ Unrotate the rate of deformation tensor
9:      $\sigma_{qs,\text{unrot}}^n \leftarrow (\mathbf{R}^n)^T \cdot \sigma_{qs,p}^n[\text{part}] \cdot \mathbf{R}^n$   $\triangleright$ Unrotate the quasistatic stress
10:     $\sigma_{\text{unrot}}^n \leftarrow (\mathbf{R}^n)^T \cdot \sigma_p^n[\text{part}] \cdot \mathbf{R}^n$   $\triangleright$ Unrotate the total stress
11:     $K^n, G^n, \mathbf{s}^n, (\overline{p}^w)^n, I_1^{\text{eff},n}, \sqrt{J_2^n}, r^n, z_{\text{eff}}^n, \varepsilon_v^{p,n} \leftarrow$ 
       $\hookrightarrow$  COMPUTEELASTICPROPERTIES( $\sigma_{qs,\text{unrot}}^n[\text{part}]$ ,
       $\hookrightarrow$   $\phi_p^n[\text{part}], S_{w,p}^n[\text{part}], \varepsilon_p^{p,n}[\text{part}], \alpha_p^n[\text{part}], p_{3,p}^n[\text{part}]$ )
       $\hookrightarrow$   $\triangleright$ Compute elastic properties and stress invariants
12:    isSuccess,  $\sigma_{qs,\text{unrot}}^{n+1}, \phi_{qs}^{n+1}, S_{w,qs}^{n+1}, X_p^{n+1}[\text{part}], \alpha_p^{n+1}[\text{part}], \varepsilon_p^{p,n+1}[\text{part}] \leftarrow$ 
       $\hookrightarrow$  RATEINDEPENDENTPLASTICUPDATE( $\Delta t, \mathbf{d}_{\text{unrot}}^{n+1}, K^n, G^n, \mathbf{s}^n, (\overline{p}^w)^n, I_1^{\text{eff},n}, \sqrt{J_2^n}, r^n, z_{\text{eff}}^n,$ 
       $\hookrightarrow$   $\varepsilon_v^{p,n}, \sigma_{qs,\text{unrot}}^n, \phi_p^n[\text{part}], S_{w,p}^n[\text{part}], X_p^n[\text{part}], \alpha_p^n[\text{part}], \varepsilon_p^{p,n}[\text{part}], p_{3,p}^n[\text{part}],$ 
       $\hookrightarrow$   $a_{1,p}[\text{part}], a_{2,p}[\text{part}], a_{3,p}[\text{part}], a_{4,p}[\text{part}]$ )
       $\hookrightarrow$   $\triangleright$ Compute updated quasistatic state using the consistency bisection algorithm
13:   if isSuccess = FALSE then

```

```

14:     FLAGPARTICLEFORDELETION(part)
15:   end if
16:    $\sigma_{unrot}^{n+1} \leftarrow \text{RATEDEPENDENTPLASTICUPDATE}(\Delta t, d_{unrot}^{n+1}, \sigma_{qs,unrot}^n, \sigma_{qs,unrot}^{n+1}, \phi_p^n[\text{part}], \phi_{qs}^{n+1},$ 
     $\hookrightarrow S_{w,p}^n[\text{part}], S_{w,qs}^{n+1}, X_p^n[\text{part}], X_p^{n+1}[\text{part}], \alpha_p^n[\text{part}], \alpha_p^{n+1}[\text{part}],$ 
     $\hookrightarrow \epsilon_p^{p,n}[\text{part}], \epsilon_p^{p,n+1}[\text{part}], p_{3,p}^n[\text{part}],$ 
     $\hookrightarrow a_{1,p}[\text{part}], a_{2,p}[\text{part}], a_{3,p}[\text{part}], a_{4,p}[\text{part}])$ 
17:    $R^{n+1}, U^{n+1} \leftarrow \text{POLARDECOMPOSITION}(F_p^{n+1}[\text{part}])$   $\triangleright$  Compute rotation and stretch tensors
18:    $\sigma_{p,qs}^{n+1}[\text{part}] \leftarrow R^{n+1} \cdot \sigma_{qs,unrot}^{n+1} \cdot (R^{n+1})^T$   $\triangleright$  Rotate the quasistatic stress
19:    $\sigma_p^{n+1}[\text{part}] \leftarrow R^{n+1} \cdot \sigma_{unrot}^{n+1} \cdot (R^{n+1})^T$   $\triangleright$  Rotate the dynamic stress
20:   end for
21:    $\Delta t^{n+1} \leftarrow \text{COMPUTESTABLETimestep}(V_p^{n+1}, m_p, \sigma_p^{n+1}, \sigma_p^n, I_p^{n+1}, \text{elasticityModel})$ 
22:   return  $\sigma_p^{n+1}, \sigma_{p,qs}^{n+1}, \phi_p^{n+1}, S_{w,p}^{n+1}, \alpha_p^{n+1}, X_p^{n+1}, \epsilon_p^{p,n+1}, \Delta t^{n+1}$ .
23: end procedure

```

### 3.2.1 Compute elastic properties

The pseudocode for the generic COMPUTEELASTICPROPERTIES is listed below. The function has side effects beyond computing the elastic properties and should be used carefully. Note that the subscript  $p$  has been dropped for simplicity because all quantities are particle-based.

#### Algorithm 5 Computing the elastic properties

**Require:**  $\sigma, \phi, S_w, \epsilon^p, \alpha, p_3, \text{elasticityModel}$

```

1: procedure COMPUTEELASTICPROPERTIES
2:    $s, \bar{p}^w, I_1^{\text{eff}}, \sqrt{J_2}, r, z_{\text{eff}} \leftarrow \text{UPDATESTRESSINVARIANTS}(\sigma, \alpha)$   $\triangleright$  Compute the deviatoric stress and
    the invariants of the input stress tensor.
3:    $\epsilon_v^p \leftarrow \text{UPDATEVOLUMETRICPLASTICSTRAIN}(\epsilon^p)$   $\triangleright$  Compute the volumetric plastic strain from the
    input plastic strain tensor.
4:    $K, G \leftarrow \text{elasticityModel.GETCURRENTELASTICMODULI}(I_1^{\text{eff}}, \bar{p}^w, \epsilon_v^p, \phi, S_w)$   $\triangleright$  Compute the
    elastic moduli corresponding to the input state.
5:   if useDisaggregationAlgorithm = TRUE then
6:     scale = MAX(exp[-(p3 +  $\epsilon_v^p$ )], 10-5)
7:      $K \leftarrow K \cdot \text{scale}$ ,  $G \leftarrow G \cdot \text{scale}$ 
8:   end if
9:   return  $K, G, s, \bar{p}^w, I_1^{\text{eff}}, \sqrt{J_2}, r, z_{\text{eff}}, \epsilon_v^p$ 
10: end procedure

```

#### Algorithm 6 Updating the stress invariants

```

1: procedure UPDATESTRESSINVARIANTS( $\sigma, \alpha$ )
2:    $I_1 \leftarrow \text{tr}(\sigma)$ 
3:    $s \leftarrow \sigma - (I_1/3)\mathbf{I}$   $\triangleright$  Compute deviatoric stress
4:    $\bar{p}^w \leftarrow -\text{tr}(\alpha)/3$   $\triangleright$  Compute pore pressure
5:    $I_1^{\text{eff}} \leftarrow I_1 + 3\bar{p}^w$ ,  $\sqrt{J_2} \leftarrow \sqrt{(1/2)s : s}$   $\triangleright$  Compute invariants of the effective stress
6:    $r \leftarrow \sqrt{2J_2}$ ,  $z_{\text{eff}} \leftarrow I_1^{\text{eff}}/\sqrt{3}$   $\triangleright$  Compute Lode coordinates of the effective stress
7:   return  $s, \bar{p}^w, I_1^{\text{eff}}, \sqrt{J_2}, r, z_{\text{eff}}$ 
8: end procedure

```

**Bulk modulus - Dry soil:** Fits to experimental data on sands suggest that the pressure model for dry soils has the form

$$\frac{\bar{p}^{\text{eff}}}{K_s(\bar{p}^{\text{eff}})} = b_0 \bar{\epsilon}_v^e + \frac{b_1(\bar{\epsilon}_v^e)^{b_4}}{b_2(\bar{\epsilon}_v^e)^{b_4} + b_3} \quad (84)$$

where the material parameters are  $b_0 > 0, b_1 > 0, b_2 > 0, b_3 > 0, b_4 > 1$ ,  $\bar{\epsilon}_v^e$  is the volumetric elastic strain in the matrix,  $K_s$  is the bulk modulus of the solid grain material, and

$$\bar{p}_{\text{eff}} := -\frac{1}{3}\text{tr}(\boldsymbol{\sigma}_{\text{eff}}) = -\frac{1}{3}\text{tr}(\boldsymbol{\sigma} - \boldsymbol{\alpha}). \quad (85)$$

The bulk modulus of the solid grains is assumed to be given by

$$K_s(\bar{p}^s) = K_{s0} + n_s (\bar{p}^s - \bar{p}_0^s) \quad (86)$$

where  $K_{s0}$  and  $n_s$  are material properties, and  $\bar{p}_0^s$  is a reference pressure.

The tangent bulk modulus of the dry soil is defined as

$$K_d(\bar{p}_{\text{eff}}) := \frac{d\bar{p}_{\text{eff}}}{d\bar{\epsilon}_v^e}. \quad (87)$$

Then, using (84),

$$K_d(\bar{p}_{\text{eff}}) = \frac{[K_s(\bar{p}_{\text{eff}})]^2}{[K_s(\bar{p}_{\text{eff}}) - n_s \bar{p}_{\text{eff}}]} \left[ b_0 + \frac{b_1 b_3 b_4 (\bar{\epsilon}_v^e)^{b_4-1}}{[b_2 (\bar{\epsilon}_v^e)^{b_4} + b_3]^2} \right]. \quad (88)$$

To express (88) in closed-form in terms of  $\bar{p}$  we have to eliminate  $\bar{\epsilon}_v^e$ . But a closed form expression for the volumetric elastic strain cannot be derived from the pressure model. So we find an approximate form of (84) by assuming  $b_0 \rightarrow 0$ . This approximation is valid at moderate to large strains. Then, from (84) with  $b_0 = 0$ , we have

$$\bar{\epsilon}_v^e \approx \left[ \frac{b_3 \bar{p}_{\text{eff}}}{b_1 K_s(\bar{p}_{\text{eff}}) - b_2 \bar{p}_{\text{eff}}} \right]^{1/b_4}. \quad (89)$$

**Shear modulus - Dry soil:** The shear modulus is typically assumed to be constant. However, a variable shear modulus may be needed to fit experimental data and to prevent negative values of Poisson's ratio in the simulations. In those situations a variable Poisson's ratio ( $\nu$ ) is defined as

$$\nu(\bar{p}_{\text{eff}}) = \nu_1 + \nu_2 \exp \left[ -\frac{K_d(\bar{p}_{\text{eff}})}{K_s(\bar{p}_{\text{eff}})} \right] \quad (90)$$

where  $\nu_1$  and  $\nu_2$  are material parameters. The shear modulus is computed using the Poisson's ratio and the dry bulk modulus:

$$G_d(\bar{p}_{\text{eff}}) = \frac{3K_d(\bar{p}_{\text{eff}})(1 - 2\nu)}{2(1 + \nu)}. \quad (91)$$

**Bulk modulus - Partially saturated soil:** The tangent bulk modulus of the partially saturated soil is found using a variation on the Gassman model for fully saturated rocks (Dvorkin et al., 1999):

$$\widehat{K}(\bar{p}_{\text{eff}}, \bar{p}^w, \bar{\epsilon}_v^p, \phi, S_w) = K_d(\bar{p}_{\text{eff}}) + \frac{\left(1 - \frac{K_d(\bar{p}_{\text{eff}})}{K_s(\bar{p}_{\text{eff}})}\right)^2}{\frac{1}{K_s(\bar{p}_{\text{eff}})} \left(1 - \frac{K_d(\bar{p}_{\text{eff}})}{K_s(\bar{p}_{\text{eff}})}\right) + \phi \left(\frac{1}{K_f(\bar{p}^w)} - \frac{1}{K_s(\bar{p}_{\text{eff}})}\right)} \quad (92)$$

where  $\widehat{K}$  is the effective bulk modulus of the partially saturated soil,  $K_d$  is the bulk modulus of the drained soil,  $K_f$  is the bulk modulus of the pore fluid, and  $K_s$  is the bulk modulus of the solid grains. At partial saturation, we compute the pore fluid bulk modulus using a harmonic mean (lower bound) on the air and water bulk moduli ( $K_a, K_w$ ):

$$\frac{1}{K_f(\bar{p}^w)} = \frac{S_w}{K_w(\bar{p}^w)} + \frac{1 - S_w}{K_a(\bar{p}^w)}. \quad (93)$$

**Shear modulus - Partially saturated soil:** The shear modulus of the saturated soil is assumed to be the same as that of the dry soil.

$$\widehat{G}(\bar{p}_{\text{eff}}, \bar{p}^w, \bar{\epsilon}_v^p, \phi, S_w) = G_d(\bar{p}_{\text{eff}}). \quad (94)$$

**The elastic modulus computation procedures:** The functions used to compute the moduli are listed in the pseudocode below.

---

**Algorithm 7** Computing the current elastic moduli

---

**Require:**  $I_1^{\text{eff}}, \bar{p}^w, \bar{\epsilon}_v^p, \phi, S_w$

- 1: **procedure** GETCURRENTELASTICMODULI
- 2:    $\bar{I}_1^{\text{eff}} \leftarrow -I_1^{\text{eff}}, \bar{\epsilon}_v^p \leftarrow -\epsilon_v^p,$
- 3:    $K \leftarrow 0, G \leftarrow 0$
- 4:   **if**  $S_w > 0$  **then**
- 5:      $K, G \leftarrow \text{COMPUTEPARTIALSATURATEDMODULI}(\bar{I}_1^{\text{eff}}, \bar{p}^w, \bar{\epsilon}_v^p, \phi, S_w)$
- 6:   **else**
- 7:      $K, G \leftarrow \text{COMPUTEDRAINEDMODULI}(\bar{I}_1^{\text{eff}}, \bar{\epsilon}_v^p)$
- 8:   **end if**
- 9:   **return**  $K, G$
- 10: **end procedure**

---



---

**Algorithm 8** Computing the partially saturated elastic moduli

---

**Require:**  $K_{s0}, n_s, \bar{p}_{s0}, K_{w0}, n_w, \bar{p}_{w0}, \gamma, \bar{p}_r$

- 1: **procedure** COMPUTEPARTIALSATURATEDMODULI( $\bar{I}_1^{\text{eff}}, \bar{p}^w, \bar{\epsilon}_v^p, \phi, S_w$ )
- 2:   **if**  $\bar{I}_1^{\text{eff}} > 0$  **then**
- 3:      $\bar{p}^{\text{eff}} \leftarrow \bar{I}_1^{\text{eff}}/3$
- 4:      $K_s \leftarrow K_{s0} + n_s(\bar{p}^{\text{eff}} - \bar{p}_{s0})$
- 5:      $K_w \leftarrow K_{w0} + n_w(\bar{p}^w - \bar{p}_{w0})$
- 6:      $K_a \leftarrow \gamma(\bar{p}^w + \bar{p}_r)$
- 7:      $K_d, G \leftarrow \text{COMPUTEDRAINEDMODULI}(\bar{I}_1^{\text{eff}}, \bar{\epsilon}_v^p)$
- 8:      $K_f \leftarrow 1/[S_w/K_w + (1 - S_w)/K_a] \triangleright \text{Bulk modulus of air + water mixture}$
- 9:      $\text{numer} \leftarrow (1 - K_d/K_s)^2$
- 10:     $\text{denom} \leftarrow (1/K_s)(1 - K_d/K_s) + \phi(1/K_f - 1/K_s)$
- 11:     $K \leftarrow K_d + \text{numer}/\text{denom} \triangleright \text{Bulk modulus of partially saturated material}$   
        $\hookrightarrow \text{(Biot-Gassman model)}$
- 12:   **else**
- 13:      $K, G \leftarrow \text{COMPUTEDRAINEDMODULI}(\bar{I}_1, \bar{\epsilon}_v^p)$
- 14:   **end if**
- 15:   **return**  $K, G$
- 16: **end procedure**

---

---

### Algorithm 9 Computing the drained elastic moduli

---

**Require:**  $K_{s0}, n_s, \bar{p}_{s0}, b_0, b_1, b_2, b_3, b_4, G_0, \nu_1, \nu_2$

```

1: procedure COMPUTEDRAINEDMODULI( $\bar{I}_1^{\text{eff}}, \bar{\epsilon}_v^p$ )
2:   if  $\bar{I}_1^{\text{eff}} > 0$  then
3:      $\bar{p}^{\text{eff}} \leftarrow \bar{I}_1^{\text{eff}}/3$ 
4:      $K_s \leftarrow K_{s0} + n_s(\bar{p}^{\text{eff}} - \bar{p}_{s0})$ 
5:      $K_s^{\text{ratio}} \leftarrow K_s/(1 - n_s \bar{p}^{\text{eff}}/K_s)$ 
6:      $\epsilon_v^e \leftarrow \text{POW}((b_3 \bar{p}^{\text{eff}})/(b_1 K_s - b_2 \bar{p}^{\text{eff}}), (1/b_4));$ 
7:      $y \leftarrow \text{POW}(\epsilon_v^e, b_4)$ 
8:      $z \leftarrow b_2 y + b_3$ 
9:      $K \leftarrow K_s^{\text{ratio}}[b_0 + (1/\epsilon_v^e)b_1 b_3 b_4 y/z^2];$   $\triangleright$  Compute compressive bulk modulus
10:     $\nu = \nu_1 + \nu_2 \exp(-K/K_s)$ 
11:     $G \leftarrow G_0$ 
12:    if  $\nu > 0$  then
13:       $G \leftarrow 1.5K(1 - 2\nu)/(1 + \nu)$   $\triangleright$  Update the shear modulus (if  $\nu_1, \nu_2 > 0$ )
14:    end if
15:  else
16:     $K \leftarrow b_0 K_{s0}$   $\triangleright$  Tensile bulk modulus = Bulk modulus at  $p = 0$ 
17:     $G \leftarrow G_0$   $\triangleright$  Tensile shear modulus
18:  end if
19:  return  $K, G$ 
20: end procedure
```

---

### 3.2.2 Rate-independent stress update

The partially saturated soil model uses Michael Homel’s “consistency bisection” algorithm (Homel, Guilkey, and Brannon, 2015) to find the plastic strain direction and to update the internal state variables. A closest-point return algorithm in transformed stress space is used to project the trial stress state on to the yield surface. Because of the nonlinearities in the material models, it is easier to solve the problem by dividing the strain increment into substeps.

Recall from equation (20) and (14) that

$$\dot{\sigma}_{\text{trial}} := \mathbf{C}^e : \mathbf{d}^s \quad (95)$$

where

$$\mathbf{C}^e = \frac{\partial \sigma}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial \bar{p}_e^w} \otimes \frac{\partial \bar{p}_e^w}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial B^e} \otimes \frac{\partial B^e}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial \phi^e} \otimes \frac{\partial \phi^e}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial S_w^e} \otimes \frac{\partial S_w^e}{\partial \epsilon^e} + \frac{\partial \sigma}{\partial \bar{X}^e} \otimes \frac{\partial \bar{X}^e}{\partial \epsilon^e}. \quad (96)$$

We assume that

$$\frac{\partial \sigma}{\partial \epsilon^e} = \mathbf{C}_\sigma^e = \left( \widehat{K} - \frac{2}{3} \widehat{G} \right) \mathbf{I} \otimes \mathbf{I} + 2 \widehat{G} \mathbf{I}^{(s)} \quad (97)$$

where  $\widehat{K}(\sigma, \bar{p}^w, B, \phi, S_w)$  is the tangent bulk modulus of the mixture,  $\widehat{G}(\sigma, \bar{p}^w, B, \phi, S_w)$  is the tangent shear modulus of the mixture,  $\mathbf{I}$  is rank-2 identity tensor, and  $\mathbf{I}^{(s)}$  is the symmetric part of the rank-4 identity tensor. All other quantities in the expression for  $\mathbf{C}^e$  are assumed to be negligible, i.e.,  $\mathbf{C}^e = \mathbf{C}_\sigma^e$ .

The update algorithm uses the standard predictor-corrector approach of hypoelastic-plasticity where a trial predictor stress is computed first and then a corrector return algorithm is used to locate the

position of the correct stress on the yield surface. This approach requires that the trial stress ( $\sigma_{\text{trial}}$ ) is computed using the relation

$$\sigma_{\text{trial}} = \sigma^n + \mathbf{C}^e : (d^s \Delta t). \quad (98)$$

After the trial stress is computed, the timestep is subdivided into substeps based on the characteristic dimension of the yield surface relative to the magnitude of the trial stress increment ( $\sigma_{\text{trial}} - \sigma^n$ ). The substep size is then recomputed by comparing the elastic bulk modulus  $\widehat{K}(\sigma_{\text{trial}}, \dots)$  with that at  $\sigma^n$  to make sure that the nonlinear elastic solution is accurate.

Recall that the parameters of the yield function, not all of which are independent, are  $a_1, a_2, a_3, a_4, R_c, \beta, I_1^{\text{peak}} = 3p_{\text{eff}}^{\text{peak}}$ . The code uses the parameters  $\alpha_1 := \text{PEAKI1}$ ,  $\alpha_2 := \text{FSLOPE}$ ,  $\alpha_3 := \text{YSLOPE}$ , and  $\alpha_4 := \text{STREN}$  in some places. The relationship between these sets of parameters is:

$$\begin{aligned} I_1^{\text{peak}} &\leftarrow \alpha_1 = \text{PEAKI1} \\ a_1 &\leftarrow \alpha_4 = \text{STREN} \\ a_2 &\leftarrow \frac{\alpha_2 - \alpha_3}{\alpha_4 - \alpha_1 \alpha_3} \\ a_3 &\leftarrow (\alpha_4 - \alpha_1 \alpha_3) \exp(-a_2 \alpha_1) \\ a_4 &\leftarrow \alpha_3 = \text{YSLOPE}. \end{aligned} \quad (99)$$

The pseudocode for the algorithm is given below. All quantities are particle-based and the subscript  $p$  has been dropped for convenience.

---

**Algorithm 10** The rate-independent stress and internal variable update algorithm

---

**Require:**  $\Delta t, d^{n+1}, K^n, G^n, s^n, (\overline{p^w})^n, I_1^{\text{eff},n}, \sqrt{J_2^n}, r^n, z_{\text{eff}}^n, \epsilon_v^{\text{p},n}, a_1, a_2, a_3, a_4, p_3^n, \sigma_{qs}^n, \phi^n, S_w^n, X^n, \alpha^n, \epsilon^{\text{p},n}, n_{\text{max}}, \epsilon_{\text{sub}}, \chi_{\text{max}}$

```

1: procedure RATEINDEPENDENTPLASTICUPDATE
2:    $\sigma_{\text{trial}} \leftarrow \text{COMPUTETRIALSTRESS}(\sigma_{qs}^n, K^n, G^n, d^{n+1}, \Delta t)$  ▷ Compute trial stress
3:    $\alpha^{\text{trial}} \leftarrow \alpha^n, p_3^{\text{trial}} \leftarrow p_3^n, \phi^{\text{trial}} \leftarrow \phi^n, S_w^{\text{trial}} \leftarrow S_w^n,$ 
      $\rightarrow X^{\text{trial}} \leftarrow X^n, \epsilon^{\text{p},\text{trial}} \leftarrow \epsilon^{\text{p},n}$  ▷ Set all other trial quantities to the values
     → at the beginning of the timestep
4:    $K^{\text{trial}}, G^{\text{trial}}, s^{\text{trial}}, (\overline{p^w})^{\text{trial}}, I_1^{\text{eff},\text{trial}}, \sqrt{J_2^{\text{trial}}}, r^{\text{trial}}, z_{\text{eff}}^{\text{trial}}, \epsilon_v^{\text{p},\text{trial}} \leftarrow$ 
      $\rightarrow \text{COMPUTEELASTICPROPERTIES}(\sigma_{\text{trial}}, \phi^{\text{trial}}, S_w^{\text{trial}}, \epsilon^{\text{p},\text{trial}}, \alpha^{\text{trial}}, p_3^{\text{trial}})$  ▷ Update the trial
     → values of the moduli and compute the invariants of the trial stress
5:    $n_{\text{sub}} \leftarrow \text{COMPUTESTEPDIVISIONS}(n_{\text{max}}, \epsilon_{\text{sub}}, K^n, K^{\text{trial}}, I_1^{\text{peak}}, a_1, X^n, \sigma_{qs}^n, \sigma_{\text{trial}})$ 
     → ▷ Compute number of substeps used by the return algorithm
6:   if  $n_{\text{sub}} < 0$  then
7:     return isSuccess = FALSE
8:   end if
9:    $\delta t \leftarrow \frac{\Delta t}{n_{\text{sub}}}$  ▷ Substep timestep
10:   $\chi \leftarrow 1, t_{\text{local}} \leftarrow 0$  ▷ Initialize substep multiplier and accumulated time increment
11:   $\sigma^k \leftarrow \sigma_{qs}^n, \epsilon^{\text{p},k} \leftarrow \epsilon^{\text{p},n}, \phi^k \leftarrow \phi^n, S_w^k \leftarrow S_w^n, X^k \leftarrow X^n, \alpha^k \leftarrow \alpha^n, K^k \leftarrow K^n, G^k \leftarrow G^n, p_3^k \leftarrow p_3^n,$ 
      $\rightarrow s^k \leftarrow s^n, (\overline{p^w})^k \leftarrow (\overline{p^w})^n, I_1^{\text{eff},k} \leftarrow I_1^{\text{eff},n}, \sqrt{J_2^k} \leftarrow \sqrt{J_2^n}, r^k \leftarrow r^n, z_{\text{eff}}^k \leftarrow z_{\text{eff}}^n, \epsilon_v^{\text{p},k} \leftarrow \epsilon_v^{\text{p},n}$ 
12:  repeat
13:    isSuccess,  $\sigma^{k+1}, \epsilon^{\text{p},k+1}, \phi^{k+1}, S_w^{k+1}, X^{k+1}, \alpha^{k+1}, K^{k+1}, G^{k+1}, p_3^{k+1} \leftarrow$ 
       $\rightarrow \text{COMPUTESUBSTEP}(\sigma^k, \epsilon^{\text{p},k}, \phi^k, S_w^k, X^k, \alpha^k, K^k, G^k, s^k, (\overline{p^w})^k, I_1^{\text{eff},k}, \sqrt{J_2^k}, r^k, z_{\text{eff}}^k,$ 
       $\rightarrow \epsilon_v^{\text{p},k}, p_3^k, d^{n+1}, \delta t)$ 
      → ▷ Compute updated stress and internal variables for the current substep

```

```

14:   if isSuccess = TRUE then
15:        $t_{\text{local}} \leftarrow t_{\text{local}} + \delta t$ 
16:        $\sigma^k \leftarrow \sigma^{k+1}, \epsilon^{p,k} \leftarrow \epsilon^{p,k+1}, \phi^k \leftarrow \phi^{k+1}, S_w^k \leftarrow S_w^{k+1}, X^k \leftarrow X^{k+1}, \alpha^k \leftarrow \alpha^{k+1}$ 
17:        $K^k \leftarrow K^{k+1}, G^k \leftarrow G^{k+1}, p_3^k \leftarrow p_3^{k+1}$ 
18:   else
19:        $\delta t \leftarrow \delta t/2$  ▷Halve the timestep
20:        $\chi \leftarrow 2\chi$  ▷Keep a count of how many times the timestep has been halved.
21:       if  $\chi > \chi_{\text{max}}$  then
22:           return isSuccess = FALSE,  $\sigma^k, \phi^k, S_w^k, X^k, \alpha^k, \epsilon^{p,k}, K^k, G^k, p_3^k$ 
23:           ↪ ▷Algorithm has failed to converge
24:       end if
25:   until  $t_{\text{local}} \geq \Delta t$ 
26:        $\sigma_{qs}^{n+1} \leftarrow \sigma^{k+1}, \alpha^{n+1} \leftarrow \alpha^{k+1}, \epsilon^{p,n+1} \leftarrow \epsilon^{p,k+1}, \phi^{n+1} \leftarrow \phi^{k+1}, S_w^{n+1} \leftarrow S_w^{k+1}, X^{n+1} \leftarrow X^{k+1}$ 
27:        $K^{n+1} \leftarrow K^{k+1}, G^{n+1} \leftarrow G^{k+1}, p_3^{n+1} \leftarrow p_3^{k+1}$ 
28:   return isSuccess = TRUE,  $\sigma_{qs}^{n+1}, \phi^{n+1}, S_w^{n+1}, X^{n+1}, \alpha^{n+1}, \epsilon^{p,n+1}, K^{n+1}, G^{n+1}, p_3^{n+1}$ 
29:   ↪ ▷Algorithm has converged
29: end procedure

```

### 3.2.3 Computing the trial stress

The trial stress is computed on the basis of the total stress rather than the effective stress. Recall that the trial stress ( $\sigma_{\text{trial}}$ ) is computed using the relation

$$\sigma_{\text{trial}} = \sigma^n + \mathbf{C}^e : (\mathbf{d}^s \Delta t) =: \sigma^n + \mathbf{C}^e : \Delta \epsilon \quad (100)$$

where

$$\mathbf{C}^e = \left( \widehat{K} - \frac{2}{3} \widehat{G} \right) \mathbf{I} \otimes \mathbf{I} + 2\widehat{G} \mathbf{I}^{(s)}. \quad (101)$$

Therefore

$$\sigma_{\text{trial}} = \sigma^n + K \text{tr}(\Delta \epsilon) \mathbf{I} + 2G [\Delta \epsilon - \frac{1}{3} \text{tr}(\Delta \epsilon) \mathbf{I}]. \quad (102)$$

The pseudocode of the trial stress algorithm is given below.

#### Algorithm 11 Computing the trial stress

```

1: procedure COMPUTETRIALSTRESS( $\sigma_{qs}^n, K^n, G^n, \mathbf{d}^{n+1}, \Delta t$ )
2:    $\Delta \epsilon \leftarrow \mathbf{d}^{n+1} \Delta t$  ▷Total strain increment
3:    $\Delta \epsilon^{\text{iso}} \leftarrow \frac{1}{3} \text{tr}(\Delta \epsilon) \mathbf{I}$ 
4:    $\Delta \epsilon^{\text{dev}} \leftarrow \Delta \epsilon - \Delta \epsilon^{\text{iso}}$ 
5:    $\sigma_{\text{trial}} \leftarrow \sigma_{qs}^n + 3K^n \Delta \epsilon^{\text{iso}} + 2G^n \Delta \epsilon^{\text{dev}}$ 
6:   return  $\sigma_{\text{trial}}$ 
7: end procedure

```

### 3.2.4 Computing the number of subcycles in the return algorithm

To allow for nonlinear parameter variations, the algorithm breaks a trial loading step into subcycles. The algorithm below, determines the number of substeps based on the magnitude of the trial stress increment relative to the characteristic dimensions of the yield surface. Another comparison uses the value of the pressure dependent elastic properties at  $\sigma_{qs}^n$  and  $\sigma_{\text{trial}}$  and adjusts the number of substeps if there is a large change in elastic moduli. This ensures an accurate solution for nonlinear elasticity even with fully elastic loading.

The number of substeps based on the bulk modulus is given by

$$n_{\text{bulk}} = \left\lceil \frac{\|K^n - K^{\text{trial}}\|}{K^n} \right\rceil. \quad (103)$$

The number of substeps based on the size of the yield surface is computed as

$$n_{\text{yield}} = \left\lceil \frac{\epsilon \|\sigma_{\text{trial}} - \sigma_{qs}^n\|}{\ell_{\text{yield}}} \right\rceil \quad (104)$$

where  $\epsilon$  is a constant, and the characteristic length of the yield surface is

$$\ell_{\text{yield}} = \text{MIN} \left( a_1, \frac{I_1^{\text{peak}} - X^n}{2} \right). \quad (105)$$

The number of substeps used in the actual calculation is

$$n_{\text{sub}} = \text{MAX}(n_{\text{bulk}}, n_{\text{yield}}). \quad (106)$$

---

#### Algorithm 12 Computing the number of subcycles

---

**Require:**  $n_{\text{max}}, \epsilon_{\text{sub}} \leftarrow 10^{-4}, K^n, K^{\text{trial}}, I_1^{\text{peak}}, a_1, X^n, \sigma_{qs}^n, \sigma_{\text{trial}}$

```

1: procedure COMPUTESTEPDIVISIONS
2:    $n_{\text{bulk}} \leftarrow \lceil \|K^n - K^{\text{trial}}\| / K^n \rceil$  ▷ Compute change in bulk modulus
3:    $\Delta\sigma \leftarrow \sigma_{\text{trial}} - \sigma_{qs}^n$ 
4:    $L \leftarrow \frac{1}{2}(I_1^{\text{peak}} - X^n)$ 
5:   if  $a_1 > 0$  then
6:      $L \leftarrow \text{MIN}(L, a_1)$ 
7:   end if
8:    $n_{\text{yield}} \leftarrow \lceil \epsilon_{\text{sub}} \times \|\Delta\sigma\| / L \rceil$  ▷ Compute trial stress increment relative to yield surface size
9:    $n_{\text{sub}} \leftarrow \text{MAX}(n_{\text{bulk}}, n_{\text{yield}})$  ▷  $n_{\text{sub}}$  is the maximum of the two values
10:  if  $n_{\text{sub}} > n_{\text{max}}$  then
11:     $n_{\text{sub}} \leftarrow -1$ 
12:  end if
13:  return  $n_{\text{sub}}$ 
14: end procedure
```

---

### 3.2.5 Updating the stress for a substep: consistency bisection

This procedure computes the updated stress state for a substep that may be either elastic, plastic, or partially elastic. It uses Homel's consistency bisection and non-hardening return concepts (Homel, Guilkey, and Brannon, 2015).

---

#### Algorithm 13 Computing the stress and internal variable update for a substep

---

**Require:**  $d^{n+1}, \delta t, \sigma^k, \epsilon^{p,k}, \phi^k, S_w^k, X^k, \alpha^k, K^k, G^k, s^k, (\bar{p}^w)^k, I_1^{\text{eff},k}, \sqrt{J_2^k}, r^k, z_{\text{eff}}^k, \epsilon_v^{p,k}, p_3^k, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta, \text{yieldCondition}$

```

1: procedure COMPUTESUBSTEP
2:    $\delta\epsilon \leftarrow d^{n+1} \delta t$  ▷ Compute strain increment
3:    $\sigma^{\text{trial}} \leftarrow \text{COMPUTETRIALSTRESS}(\sigma^k, K^k, G^k, d^{n+1}, \Delta t)$  ▷ Compute substep trial stress
4:    $\alpha^{\text{trial}} \leftarrow \alpha^k, K^{\text{trial}} \leftarrow K^k, G^{\text{trial}} \leftarrow G^k, p_3^{\text{trial}} \leftarrow p_3^k, \phi^{\text{trial}} \leftarrow \phi^k, S_w^{\text{trial}} \leftarrow S_w^k,$   

    $\hookrightarrow X^{\text{trial}} \leftarrow X^k, \epsilon^{p,\text{trial}} \leftarrow \epsilon^{p,k}$  ▷ Set all other trial quantities to the values  

    $\hookrightarrow$  at the beginning of the substep
```



```

5:    $K^{\text{trial}}, G^{\text{trial}}, \mathbf{s}^{\text{trial}}, (\bar{p}^w)^{\text{trial}}, I_1^{\text{eff,trial}}, \sqrt{J_2^{\text{trial}}}, r^{\text{trial}}, z_{\text{eff}}^{\text{trial}}, \epsilon_v^{\text{p,trial}} \leftarrow$ 
       $\hookrightarrow \text{COMPUTEELASTICPROPERTIES}(\sigma_{\text{trial}}, \phi^{\text{trial}}, S_w^{\text{trial}}, \epsilon^{\text{p,trial}}, \alpha^{\text{trial}}, p_3^{\text{trial}})$ 
       $\hookrightarrow \triangleright \text{Compute elastic moduli and stress invariants for the trial state}$ 
6:    $\text{isElastic} \leftarrow \text{yieldCondition.EVALYIELDCONDITION}(I_1^{\text{eff,trial}}, \sqrt{J_2^{\text{trial}}}, X^{\text{trial}}, (\bar{p}^w)^{\text{trial}}, \phi^{\text{trial}}, S_w^{\text{trial}},$ 
       $\hookrightarrow a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta)$ 
       $\hookrightarrow \triangleright \text{Determine whether the trial stress is elastic or not}$ 
7:   if  $\text{isElastic} = \text{TRUE}$  then
8:      $\sigma^{k+1} \leftarrow \sigma_{\text{trial}}, \epsilon^{\text{p},k+1} \leftarrow \epsilon^{\text{p,trial}}, \phi^{k+1} \leftarrow \phi^{\text{trial}}, S_w^{k+1} \leftarrow S_w^{\text{trial}}, X^{k+1} \leftarrow X^{\text{trial}}, \alpha^{k+1} \leftarrow \alpha^{\text{trial}}$ 
9:      $K^{k+1} \leftarrow K^{\text{trial}}, G^{k+1} \leftarrow G^{\text{trial}}, p_3^{k+1} \leftarrow p_3^{\text{trial}}$ 
       $\hookrightarrow \triangleright \text{This is an elastic substep. Update the state to the trial value.}$ 
10:     $\text{isSuccess} = \text{TRUE}$ 
11:    return  $\text{isSuccess}, \sigma^{k+1}, \epsilon^{\text{p},k+1}, \phi^{k+1}, S_w^{k+1}, X^{k+1}, \alpha^{k+1}, K^{k+1}, G^{k+1}, p_3^{k+1}$ 
12:  end if
13:   $\sigma_{\text{fixed}}, \delta \epsilon_{\text{fixed}}^{\text{p}} \leftarrow \text{NONHARDENINGRETURN}(\sigma^k, \delta \epsilon, X^k, K^k, G^k, (\bar{p}^w)^k,$ 
       $\hookrightarrow \mathbf{s}^{\text{trial}}, \sqrt{J_2^{\text{trial}}}, r^{\text{trial}}, z_{\text{eff}}^{\text{trial}}, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta)$ 
       $\hookrightarrow \triangleright \text{Compute return to updated yield surface (no hardening)}$ 
14:   $\text{isSuccess}, \sigma^{k+1}, \epsilon^{\text{p},k+1}, \alpha^{k+1}, (\bar{p}^w)^{k+1}, \phi^{k+1}, S_w^{k+1}, X^{k+1}, K^{k+1}, G^{k+1}, \mathbf{s}^{k+1}, (\bar{p}^w)^{k+1}, I_1^{\text{eff},k+1}, \sqrt{J_2^{k+1}},$ 
       $\hookrightarrow r^{k+1}, z_{\text{eff}}^{k+1}, \epsilon_v^{\text{p},k+1} \leftarrow \text{CONSISTENCYBISECTION}(\delta \epsilon, \epsilon^{\text{p},k}, \sigma^k, K^k, G^k, (\bar{p}^w)^k, \phi^k, S_w^k, X^k,$ 
       $\hookrightarrow \mathbf{s}^{\text{trial}}, I_1^{\text{eff,trial}}, \sqrt{J_2^{\text{trial}}}, r^{\text{trial}}, z_{\text{eff}}^{\text{trial}}, \epsilon_v^{\text{p,trial}}, p_3^{\text{trial}}, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta, i_{\text{max}}, j_{\text{max}},$ 
       $\hookrightarrow \sigma_{\text{fixed}}, \delta \epsilon_{\text{fixed}}^{\text{p}}) \triangleright \text{The bisection return algorithm to take care of yield surface hardening.}$ 
15:  if  $\text{isSuccess} = \text{FALSE}$  then
16:    return  $\text{isSuccess}, \sigma^k, \epsilon^{\text{p},k}, \phi^k, S_w^k, X^k, \alpha^k, K^k, G^k, p_3^k$ 
17:  end if
18:  return  $\text{isSuccess}, \sigma^{k+1}, \epsilon^{\text{p},k+1}, \phi^{k+1}, S_w^{k+1}, X^{k+1}, \alpha^{k+1}, K^{k+1}, G^{k+1}, p_3^{k+1}$ 
19: end procedure

```

### 3.2.6 The nonhardening return algorithm

Let the plastic flow direction be  $\mathbf{M}$ . Then

$$\dot{\epsilon}^{\text{p}} = \dot{\lambda} \mathbf{M}. \quad (107)$$

The nonhardening return algorithm uses a transformed space (see Homel, Guilkey, and Brannon (2015)) where the computation is carried out in special Lode coordinates  $(z_{\text{eff}}, r')$  where

$$z_{\text{eff}} := \frac{\text{tr}(\sigma - \alpha)}{\sqrt{3}} \quad \text{and} \quad r' = \beta r \sqrt{\frac{3K}{2G}}, \quad r := \sqrt{2J_2}. \quad (108)$$

If the flow rule is non-associative, the yield surface parameter  $\beta \neq 1$ .

The closest point from the trial stress state to the yield surface in this transformed space,  $(z_{\text{eff}}^{\text{close}}, r'_{\text{close}})$ , is computed using a geometric algorithm discussed later. After the closest point has been found, the coordinates of the closest point are transformed back to the  $(z, r)$  coordinates and the stress state is reconstructed using

$$\sigma^{\text{fixed}} = z^{\text{close}} \frac{\mathbf{I}}{\|\mathbf{I}\|} + r^{\text{close}} \frac{\mathbf{s}^{\text{trial}}}{\|\mathbf{s}^{\text{trial}}\|}. \quad (109)$$

Let the stress at the beginning of the load step be  $\sigma^k$  and let the trial stress be  $\sigma_{\text{trial}}$ . Assume the yield surface is fixed and let the correct projection of the trial stress on to the fixed yield surface be  $\sigma_{\text{fixed}}^{k+1}$ .

The increment of stress for the load step ( $\Delta\sigma_{\text{fixed}}$ ) is related to the elastic strain increment ( $\Delta\epsilon_{\text{fixed}}^e$ ) by

$$\Delta\sigma_{\text{fixed}} = \sigma_{\text{fixed}}^{k+1} - \sigma^k = \mathbf{C}_{\text{fixed}}^e : \Delta\epsilon_{\text{fixed}}^e \quad (110)$$

where  $\mathbf{C}_{\text{fixed}}^e$  is a constant elastic modulus tensor. The elastic modulus tensor can be assumed to be an average value of the nonlinear tangent modulus for the load step. If we know  $\mathbf{C}_{\text{fixed}}^e$ , we can compute the elastic strain increment using

$$\Delta\epsilon_{\text{fixed}}^e = (\mathbf{C}_{\text{fixed}}^e)^{-1} : \Delta\sigma_{\text{fixed}}. \quad (111)$$

For the situation where  $\mathbf{C}^e$  is given by equation (101), we have

$$(\mathbf{C}^e)^{-1} = \left( \frac{1}{9K^k} - \frac{1}{6G^k} \right) \mathbf{I} \otimes \mathbf{I} + \frac{1}{2G^k} \mathbf{I}^{(s)}. \quad (112)$$

Therefore,

$$\begin{aligned} \Delta\epsilon_{\text{fixed}}^e &= \left( \frac{1}{9K^k} - \frac{1}{6G^k} \right) \text{tr}(\Delta\sigma_{\text{fixed}}) \mathbf{I} + \frac{1}{2G^k} \Delta\sigma_{\text{fixed}} \\ &= \frac{1}{3K^k} \left[ \frac{1}{3} \text{tr}(\Delta\sigma_{\text{fixed}}) \mathbf{I} \right] + \frac{1}{2G^k} \left[ \Delta\sigma_{\text{fixed}} - \frac{1}{3} \text{tr}(\Delta\sigma_{\text{fixed}}) \mathbf{I} \right] \end{aligned} \quad (113)$$

or

$$\Delta\epsilon_{\text{fixed}}^e = \frac{1}{3K^k} \Delta\sigma_{\text{fixed}}^{\text{iso}} + \frac{1}{2G^k} \Delta\sigma_{\text{fixed}}^{\text{dev}}. \quad (114)$$

For a strain driven update algorithm, the total strain increment  $\Delta\epsilon$  is known. Assuming that the total strain increment can be additively decomposed into an elastic and a plastic part, we can find the plastic strain increment ( $\Delta\epsilon_{\text{fixed}}^p$ ) using

$$\Delta\epsilon_{\text{fixed}}^p = \Delta\epsilon - \Delta\epsilon_{\text{fixed}}^e. \quad (115)$$

The nonhardening return algorithm pseudocode is listed below:

---

#### Algorithm 14 Non-hardening return algorithm

---

**Require:**  $\sigma^k, \delta\epsilon, X^k, K^k, G^k, (\bar{p}^w)^k, s^{\text{trial}}, \sqrt{J_2^{\text{trial}}}, r^{\text{trial}}, z_{\text{eff}}^{\text{trial}}, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta, \text{yieldCondition}$

1: **procedure** NONHARDENINGRETURN

2:  $r'_{\text{trial}} \leftarrow \beta r^{\text{trial}} \sqrt{\frac{3K^k}{2G^k}}$  *▷ Transform the trial r coordinate*

3:  $X_{\text{eff}}^k \leftarrow X^k + 3(\bar{p}^w)^k$

4:  $z_{\text{eff}}^{\text{close}}, r'_{\text{close}} \leftarrow \text{yieldCondition.GETCLOSESTPOINT}(K^k, G^k, X_{\text{eff}}^k, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta,$   
 $\quad \hookrightarrow z_{\text{eff}}^{\text{trial}}, r'_{\text{trial}})$

5:  $I_1^{\text{close}} \leftarrow \sqrt{3} z_{\text{eff}}^{\text{close}} - 3(\bar{p}^w)^k, \sqrt{J_2^{\text{close}}} \leftarrow \frac{1}{\beta} \sqrt{\frac{G^k}{3K^k}} r'_{\text{close}}$

6: **if**  $\sqrt{J_2^{\text{trial}}} > 0$  **then**

7:  $\sigma^{\text{fixed}} = \frac{1}{3} I_1^{\text{close}} \mathbf{I} + \frac{\sqrt{J_2^{\text{close}}}}{\sqrt{J_2^{\text{trial}}}} s^{\text{trial}}$  *▷ Compute updated total stress*

8: **else**

9:  $\sigma^{\text{fixed}} = \frac{1}{3} I_1^{\text{close}} \mathbf{I} + s^{\text{trial}}$  *▷ Compute updated total stress when the trial stress is hydrostatic*

10: **end if**

11:  $\delta\sigma_{\text{fixed}} \leftarrow \sigma^{\text{fixed}} - \sigma^k$  *▷ Compute stress increment*

12:  $\delta\sigma_{\text{fixed}}^{\text{iso}} \leftarrow \frac{1}{3} \text{tr}(\delta\sigma_{\text{fixed}}) \mathbf{I}, \quad \delta\sigma_{\text{fixed}}^{\text{dev}} \leftarrow \delta\sigma_{\text{fixed}} - \delta\sigma_{\text{fixed}}^{\text{iso}}$

13:  $\delta\epsilon^{\text{p, fixed}} = \delta\epsilon - \frac{1}{3K^k} \delta\sigma_{\text{fixed}}^{\text{iso}} - \frac{1}{2G^k} \delta\sigma_{\text{fixed}}^{\text{dev}}$  *▷ Compute plastic strain increment*

14: **return**  $\sigma^{\text{fixed}}, \delta\epsilon^{\text{p, fixed}}$

15: **end procedure**

---

## Finding the closest point in transformed space

**Algorithm 15** Compute the closest point from the trial state to transformed non-hardening yield surface

**Require:**  $K^k, G^k, X_{\text{eff}}^k, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta, z_{\text{eff}}^{\text{trial}}, r'_{\text{trial}}$

- 1: **procedure** GETCLOSESTPOINT
- 2:    $n_{\text{poly}} \leftarrow 1000$
- 3:    $\mathbf{x}_{\text{poly}} \leftarrow \text{GETYIELDSURFACEPOINTSALL\_RPRIMEZ}(n_{\text{poly}}, K^k, G^k, X_{\text{eff}}^k,$   
      $\hookrightarrow a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta)$   
      $\hookrightarrow \triangleright \text{Get the polygon that represents the yield surface in } z_{\text{eff}}\text{-}r' \text{ space.}$
- 4:    $\mathbf{x}_{\text{seg}} \leftarrow \text{GETCLOSESTSEGMENTS}(z_{\text{eff}}^{\text{trial}}, r'_{\text{trial}}, \mathbf{x}_{\text{poly}})$   
      $\hookrightarrow \triangleright \text{Find two yield surface segments that are closest to the trial stress state.}$
- 5:    $\mathbf{x}_{\text{segpoly}} \leftarrow \text{GETYIELDSURFACEPOINTSSEGMENT\_RPRIMEZ}(n_{\text{poly}}, K^k, G^k, X_{\text{eff}}^k,$   
      $\hookrightarrow a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta, \mathbf{x}_{\text{seg}}[1], \mathbf{x}_{\text{seg}}[3])$   $\triangleright \text{Discretize the two closest segments.}$
- 6:    $\mathbf{x}_{\text{close}} \leftarrow \text{FINDCLOSESTPOINT}(z_{\text{eff}}^{\text{trial}}, r'_{\text{trial}}, \mathbf{x}_{\text{segpoly}})$   
      $\hookrightarrow \triangleright \text{Find the closest point in the discretized segments to the trial stress state.}$
- 7:   **return** isSuccess = TRUE,  $\mathbf{x}_{\text{close}}.z_{\text{eff}}, \mathbf{x}_{\text{close}}.r'$
- 8: **end procedure**

## Finding the yield surface polygon in $z_{\text{eff}}\text{-}r'$ space

**Algorithm 16** Find points in a closed polygon that describes the yield surface in  $z_{\text{eff}}\text{-}r'$  space

**Require:**  $n_{\text{poly}}, K^k, G^k, X_{\text{eff}}^k, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta$

- 1: **procedure** GETYIELDSURFACEPOINTSALL\_RPRIMEZ
- 2:    $\kappa \leftarrow I_1^{\text{peak}} - R_c(I_1^{\text{peak}} - X_{\text{eff}}^k)$   $\triangleright \text{Compute } \kappa.$
- 3:    $I_1^{\text{eff}} \leftarrow \text{Linspace}(\text{from} = X_{\text{eff}}^k, \text{to} = I_1^{\text{peak}}, \text{points} = n_{\text{poly}})$   
      $\hookrightarrow \triangleright \text{Create an equally spaced set of } I_1^{\text{eff}} \text{ values.}$
- 4:   **for**  $I_1$  **in**  $I_1^{\text{eff}}$  **do**
- 5:      $F_f = a_1 - a_3 \exp(a_2 I_1) - a_4 I_1$ ;  $\triangleright \text{Compute } F_f.$
- 6:      $F_c^2 \leftarrow 1$
- 7:     **if**  $I_1 < \kappa$  **and**  $X_{\text{eff}}^k < I_1$  **then**
- 8:        $F_c^2 = 1 - \left[ \frac{\kappa - I_1}{\kappa - X_{\text{eff}}^k} \right]^2$ ;  $\triangleright \text{Compute } F_c.$
- 9:     **end if**
- 10:     $J_2 = F_f^2 F_c^2$   $\triangleright \text{Compute } J_2 \text{ and push into a vector}$
- 11:   **end for**
- 12:    $z_{\text{eff}} \leftarrow I_1^{\text{eff}} / \sqrt{3}$ ,  $r' \leftarrow \beta \sqrt{\frac{3K^k}{2G^k}} \sqrt{2J_2}$
- 13:    $\mathbf{x}_{\text{poly}}.z_{\text{eff}} \leftarrow z_{\text{eff}} \cup \text{REVERSE}(z_{\text{eff}})$ ,  $\mathbf{x}_{\text{poly}}.r' \leftarrow r' \cup \text{REVERSE}(-r')$   
      $\hookrightarrow \triangleright \text{Add the points on the negative } r' \text{ side of the polygon}$
- 14:    $\mathbf{x}_{\text{poly}}[2n_{\text{poly}} + 1] \leftarrow \mathbf{x}_{\text{poly}}[1]$   $\triangleright \text{Add the first point to close the polygon}$
- 15:   **return**  $\mathbf{x}_{\text{poly}}$
- 16: **end procedure**

## Locating the closest segments of the yield surface polygon in $z_{\text{eff}}\text{-}r'$ space

**Algorithm 17** Locate two closest segments of the yield surface polygon to the trial stress state

**Require:**  $z_{\text{eff}}^{\text{trial}}, r'_{\text{trial}}, \mathbf{x}_{\text{poly}}$

- 1: **procedure** GETCLOSESTSEGMENTS
- 2:    $n_{\text{poly}} \leftarrow \text{LENGTH}(\mathbf{x}_{\text{poly}})$   $\triangleright \text{Get the number of points in the closed polygon}$

```

3:    $\mathbf{x}_{\text{prev}} \leftarrow \mathbf{x}_{\text{poly}}[n_{\text{poly}} - 1]$  ▷Get the second to last point in the closed polygon
4:    $i_{\text{next}} \leftarrow 2$ ,  $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x}_{\text{poly}}[i_{\text{next}}]$  ▷Get the second point in the closed polygon
5:    $d_{\text{min}}^2 \leftarrow \text{DOUBLE\_MAX}$ 
6:    $\mathbf{x}_{\text{min}} \leftarrow \mathbf{0}$ ,  $\mathbf{x}_{\text{prev}}^{\text{min}} \leftarrow \mathbf{0}$ ,  $\mathbf{x}_{\text{next}}^{\text{min}} \leftarrow \mathbf{0}$ 
7:   for  $\mathbf{x}_{\text{cur}}$  in  $\mathbf{x}_{\text{poly}}$  do
8:      $d^2 \leftarrow \text{DISTANCESQ}(z_{\text{eff}}^{\text{trial}}, r'_{\text{trial}}, \mathbf{x}_{\text{cur}}.z_{\text{eff}}, \mathbf{x}_{\text{cur}}.r')$ 
9:     if  $d^2 < d_{\text{min}}^2$  then
10:       $d_{\text{min}}^2 \leftarrow d^2$ 
11:       $\mathbf{x}_{\text{min}} \leftarrow \mathbf{x}_{\text{cur}}$ ,  $\mathbf{x}_{\text{prev}}^{\text{min}} \leftarrow \mathbf{x}_{\text{prev}}$ ,  $\mathbf{x}_{\text{next}}^{\text{min}} \leftarrow \mathbf{x}_{\text{next}}$ 
12:    end if
13:     $i_{\text{next}} \leftarrow i_{\text{next}} + 1$ 
14:    if  $i_{\text{next}} = n_{\text{poly}}$  then
15:      break ▷Since the polygon is closed, ignore the last point
16:    end if
17:     $\mathbf{x}_{\text{prev}} \leftarrow \mathbf{x}_{\text{cur}}$ ,  $\mathbf{x}_{\text{next}} \leftarrow \mathbf{x}_{\text{poly}}[i_{\text{next}}]$ 
18:  end for
19:   $\mathbf{x}_{\text{seg}} \leftarrow \{\mathbf{x}_{\text{prev}}^{\text{min}}, \mathbf{x}_{\text{min}}, \mathbf{x}_{\text{next}}^{\text{min}}\}$ 
20:  return  $\mathbf{x}_{\text{seg}}$ 
21: end procedure

```

### Finding the segments of the surface polygon in $z_{\text{eff}}-r'$ space

**Algorithm 18** Find points in an open segment that describes the yield surface in  $z_{\text{eff}}-r'$  space

**Require:**  $n_{\text{poly}}$ ,  $K^k$ ,  $G^k$ ,  $X_{\text{eff}}^k$ ,  $a_1$ ,  $a_2$ ,  $a_3$ ,  $a_4$ ,  $I_1^{\text{peak}}$ ,  $R_c$ ,  $\beta$ ,  $\mathbf{x}_{\text{start}}$ ,  $\mathbf{x}_{\text{end}}$

```

1: procedure GETYIELDSURFACEPOINTSSEGMENT_RPRIMEZ
2:    $\kappa \leftarrow I_1^{\text{peak}} - R_c(I_1^{\text{peak}} - X_{\text{eff}}^k)$  ▷Compute  $\kappa$ .
3:    $I_1^{\text{start}} \leftarrow \sqrt{3}\mathbf{x}_{\text{start}}.z_{\text{eff}}$ ,  $I_1^{\text{end}} \leftarrow \sqrt{3}\mathbf{x}_{\text{end}}.z_{\text{eff}}$ 
4:    $I_1^{\text{eff}} \leftarrow \text{Linspace}(\text{from} = I_1^{\text{start}}, \text{to} = I_1^{\text{end}}, \text{points} = n_{\text{poly}})$ 
   ↪ ▷Create an equally spaced set of  $I_1^{\text{eff}}$  values.
5:   for  $I_1$  in  $I_1^{\text{eff}}$  do
6:      $F_f = a_1 - a_3 \exp(a_2 I_1) - a_4 I_1$ ; ▷Compute  $F_f$ .
7:      $F_c^2 \leftarrow 1$ 
8:     if  $I_1 < \kappa$  and  $X_{\text{eff}}^k < I_1$  then
9:        $F_c^2 = 1 - \left[ \frac{\kappa - I_1}{\kappa - X_{\text{eff}}^k} \right]^2$ ; ▷Compute  $F_c$ .
10:    end if
11:     $J_2 = F_f^2 F_c^2$  ▷Compute  $J_2$  and push into a vector
12:  end for
13:   $z_{\text{eff}} \leftarrow I_1^{\text{eff}}/\sqrt{3}$ ,  $r' \leftarrow \beta \sqrt{\frac{3K^k}{2G^k}} \sqrt{2J_2}$ 
14:   $\mathbf{x}_{\text{segpoly}}.z_{\text{eff}} \leftarrow z_{\text{eff}} \cup \text{REVERSE}(z_{\text{eff}})$ ,  $\mathbf{x}_{\text{segpoly}}.r' \leftarrow r' \cup \text{REVERSE}(-r')$ 
   ↪ ▷Add the points on the negative  $r'$  side of the polygon
15:  return  $\mathbf{x}_{\text{segpoly}}$ 
16: end procedure

```

### Finding the closest point on yield surface segments in $z_{\text{eff}}-r'$ space

**Algorithm 19** Find the closest point from the trial stress state on the polyline describing the yield surface

**Require:**  $\mathbf{x}_{\text{trial}}.z_{\text{eff}}$ ,  $\mathbf{x}_{\text{trial}}.r'$ ,  $\mathbf{x}_{\text{segpoly}}$

```

1: procedure FINDCLOSESTPOINT
2:    $i \leftarrow 0$ 

```

```

3:   for { $\mathbf{x}_{\text{segstart}}, \mathbf{x}_{\text{segend}}$ } in  $\mathbf{x}_{\text{segpoly}}$  do
4:      $\mathbf{x}_{\text{seg}} \leftarrow \mathbf{x}_{\text{segend}} - \mathbf{x}_{\text{segstart}}$ 
5:      $\mathbf{x}_{\text{proj}} \leftarrow \mathbf{x}_{\text{trial}} - \mathbf{x}_{\text{segstart}}$ 
6:      $t \leftarrow \frac{\mathbf{x}_{\text{proj}} \cdot \mathbf{x}_{\text{seg}}}{\|\mathbf{x}_{\text{seg}}\|}$ 
7:      $i \leftarrow i + 1$ 
8:     if  $t < 0$  then
9:        $\mathbf{x}[i] \leftarrow \mathbf{x}_{\text{segstart}}$ 
10:    else if  $t > 1$  then
11:       $\mathbf{x}[i] \leftarrow \mathbf{x}_{\text{segend}}$ 
12:    else
13:       $\mathbf{x}[i] \leftarrow \mathbf{x}_{\text{segstart}} + t \mathbf{x}_{\text{seg}}$ 
14:    end if
15:  end for
16:   $d_{\min}^2 \leftarrow \text{DOUBLE\_MAX}$ 
17:   $\mathbf{x}_{\text{close}} \leftarrow \mathbf{0}$ 
18:  for  $\mathbf{x}_i$  in  $\mathbf{x}$  do
19:     $d^2 \leftarrow \text{DISTANCESQ}(\mathbf{x}_i, \mathbf{x}_{\text{trial}})$ 
20:    if  $d^2 < d_{\min}^2$  then
21:       $d_{\min}^2 \leftarrow d^2$ 
22:       $\mathbf{x}_{\text{close}} \leftarrow \mathbf{x}_i$ 
23:    end if
24:  end for
25:  return  $\mathbf{x}_{\text{close}}$ 
26: end procedure

```

### 3.2.7 Consistency bisection algorithm

Now, if we allow the yield surface to harden, the distance between the trial stress point and its projection on to the yield surface decreases compared to that for a fixed yield surface. If  $\Delta \epsilon^P$  is the plastic strain increment for a hardening yield surface, we have

$$\Delta \epsilon^P < \Delta \epsilon_{\text{fixed}}^P \quad (116)$$

where the inequality can be evaluated using an appropriate Euclidean norm. Note that this distance is proportional to the consistency parameter  $\lambda$ .

**Fully saturated model** In Homel's *fully saturated* version of the Arenisca model (Homel, Guilkey, and Brannon, 2015), the internal variables are the hydrostatic compressive strength ( $X$ ) and the scalar isotropic backstress ( $\alpha$ ). These depend only on the *volumetric* plastic strain increment

$$\Delta \epsilon_v^P = \text{tr}(\Delta \epsilon^P). \quad (117)$$

Because  $\Delta \epsilon_v^P < \Delta \epsilon_{v,\text{fixed}}^P$  we can define a parameter,  $\eta \in (0, 1)$ , such that

$$\eta := \frac{\Delta \epsilon_v^P}{\Delta \epsilon_{v,\text{fixed}}^P}. \quad (118)$$

Because the solution is bounded by the fixed yield surface, a **bisection algorithm can be used to find the parameter  $\eta$** .

**Partially saturated model** To use the same approach for our *partially saturated model*, the internal variables have to be of the form

$$\overline{p}_p^w \equiv \overline{p}_p^w(\epsilon_v^P), \quad B^P \equiv B^P(\epsilon_v^P), \quad \phi^P \equiv \phi^P(\epsilon_v^P), \quad S_w^P \equiv S_w^P(\epsilon_v^P), \quad \overline{X}^P \equiv \overline{X}^P(\epsilon_v^P) \quad (119)$$

so that

$$\dot{\bar{p}}_p^w = \lambda \frac{d\bar{p}_p^w}{d\varepsilon_v^p} \text{tr}(\mathbf{M}), \quad \dot{B}^p = \lambda \frac{dB^p}{d\varepsilon_v^p} \text{tr}(\mathbf{M}), \quad \dot{\phi}^p = \lambda \frac{d\phi^p}{d\varepsilon_v^p} \text{tr}(\mathbf{M}), \quad \dot{S}_w^p = \lambda \frac{dS_w^p}{d\varepsilon_v^p} \text{tr}(\mathbf{M}), \quad \dot{\bar{X}}^p = \lambda \frac{d\bar{X}^p}{d\varepsilon_v^p} \text{tr}(\mathbf{M}). \quad (120)$$

Recall that

$$\dot{\bar{p}}_p^w = \frac{1}{B} \text{tr}(\mathbf{d}^p) = \frac{\lambda}{B} \text{tr}(\mathbf{M}) \quad (121)$$

$$\dot{B}^p = \lambda \left[ \frac{1}{BK_s} \left( \frac{K_d}{K_s} \frac{dK_d}{d\bar{p}^w} - \frac{dK_s}{d\bar{p}^w} \right) \right] \text{tr}(\mathbf{M}) \quad (122)$$

$$\dot{S}_w^p = \lambda \left[ \frac{1 - S_0}{B(1 - S_0 + C)^2} \frac{dC}{d\bar{p}^w} \right] \text{tr}(\mathbf{M}) \quad (123)$$

$$\dot{\phi}^p = -\lambda \left[ (1 - S_0) \phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) \left( 1 + \frac{1}{K_a B} \right) + S_0 \phi_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \left( 1 + \frac{1}{K_w B} \right) \right] \text{tr}(\mathbf{M}) \quad (124)$$

$$\dot{\bar{X}}^p := \lambda \left[ \left[ (1 - S_w) + p_1^{\text{sat}} S_w \right] \frac{d\bar{X}_d}{d\varepsilon_v^p} + \frac{\bar{X}_d (p_1^{\text{sat}} - 1)(1 - S_0)}{B(1 - S_0 + C)^2} \frac{dC}{d\bar{p}^w} + \frac{3\bar{p}^w}{BK_s} \left( \frac{K_d}{K_s} \frac{dK_d}{d\bar{p}^w} - \frac{dK_s}{d\bar{p}^w} \right) + \frac{3B}{B} \right] \text{tr}(\mathbf{M}) \quad (125)$$

where

$$B := \frac{1}{(1 - S_0) \exp[\bar{\varepsilon}_v - \bar{\varepsilon}_v^a(\bar{p}^w)] + S_0 \exp[\bar{\varepsilon}_v - \bar{\varepsilon}_v^w(\bar{p}^w)]} \left[ -\frac{(B - \phi) \phi}{B \phi_0} \left( \frac{S_w}{K_w} + \frac{1 - S_w}{K_a} \right) + \frac{1 - S_0}{K_a} \exp[\bar{\varepsilon}_v - \bar{\varepsilon}_v^a(\bar{p}^w)] + \frac{S_0}{K_w} \exp[\bar{\varepsilon}_v - \bar{\varepsilon}_v^w(\bar{p}^w)] \right] \quad (126)$$

$$C := S_0 \exp[\bar{\varepsilon}_v^a(\bar{p}^w) - \bar{\varepsilon}_v^w(\bar{p}^w)].$$

These equation are consistent with Homel's assumptions (allowing for the use of a bisection algorithm) only if

$$\bar{p}^w = \bar{p}_p^w, \quad B = B^p, \quad \phi = \phi^p, \quad S_w = S_w^p, \quad \bar{X} = \bar{X}^p. \quad (127)$$

This is a strong assumption and may violate the balance of mass. However, to keep the algorithm efficient, we make the assumptions in (127) and hence can use the bisection algorithm discussed later.

Now we can compare equations (120) and (121), and use the assumption that  $B \approx 1$ , to get the following set of coupled ordinary differential equations for the internal variables.

$$\begin{aligned} \frac{d\bar{p}_p^w}{d\varepsilon_v^p} &= \frac{1}{B^p} \\ \frac{d\phi^p}{d\varepsilon_v^p} &= (S_0 - 1) \phi_0 \exp[\bar{\varepsilon}_v^p - \bar{\varepsilon}_v^a(\bar{p}_p^w)] \left( 1 + \frac{1}{K_a(\bar{p}_p^w)B^p} \right) - S_0 \phi_0 \exp[\bar{\varepsilon}_v^p - \bar{\varepsilon}_v^w(\bar{p}_p^w)] \left( 1 + \frac{1}{K_w(\bar{p}_p^w)B^p} \right) \\ \frac{dS_w^p}{d\varepsilon_v^p} &= \frac{1 - S_0}{B^p(1 - S_0 + C^p)^2} \frac{dC^p}{d\bar{p}_p^w} \\ \frac{d\bar{X}^p}{d\varepsilon_v^p} &= \left[ (1 - S_w^p) + p_1^{\text{sat}} S_w^p \right] \frac{d\bar{X}_d}{d\varepsilon_v^p} + \frac{\bar{X}_d (p_1^{\text{sat}} - 1)(1 - S_0)}{B^p(1 - S_0 + C^p)^2} \frac{dC^p}{d\bar{p}_p^w} + \frac{3}{B^p} \end{aligned} \quad (128)$$

where

$$\begin{aligned} \mathcal{B}^P &:= \frac{1}{(1 - S_0) \exp[\bar{\varepsilon}_v^P - \bar{\varepsilon}_v^a(\bar{p}_p^w)] + S_0 \exp[\bar{\varepsilon}_v^P - \bar{\varepsilon}_v^w(\bar{p}_p^w)]} \left[ -\frac{(1 - \phi^P) \phi^P}{\phi_0} \left( \frac{S_w^P}{K_w(\bar{p}_p^w)} + \frac{1 - S_w^P}{K_a(\bar{p}_p^w)} \right) + \right. \\ &\quad \left. \frac{1 - S_0}{K_a(\bar{p}_p^w)} \exp[\bar{\varepsilon}_v^P - \bar{\varepsilon}_v^a(\bar{p}_p^w)] + \frac{S_0}{K_w(\bar{p}_p^w)} \exp[\bar{\varepsilon}_v^P - \bar{\varepsilon}_v^w(\bar{p}_p^w)] \right] \\ \mathcal{C}^P &:= S_0 \exp[\bar{\varepsilon}_v^a(\bar{p}_p^w) - \bar{\varepsilon}_v^w(\bar{p}_p^w)], \quad \frac{d\mathcal{C}^P}{d\bar{\varepsilon}_v^P} = \mathcal{C}^P \left[ \frac{1}{K_a(\bar{p}_p^w)} - \frac{1}{K_w(\bar{p}_p^w)} \right]. \end{aligned} \quad (129)$$

If the vector of internal variables is  $\boldsymbol{\eta} := \{\bar{p}_p^w, \phi^P, S_w^P, \bar{X}^P\}$ , we can express the above system as

$$\frac{d\boldsymbol{\eta}}{d\bar{\varepsilon}_v^P} = \mathbf{F}(\boldsymbol{\eta}, \bar{\varepsilon}_v^P). \quad (130)$$

We will use a forward Euler integration approach to update the internal variables:

$$\boldsymbol{\eta}^{n+1} = \boldsymbol{\eta}^n + \mathbf{F}[\boldsymbol{\eta}^n, (\bar{\varepsilon}_v^P)^n] \Delta \bar{\varepsilon}_v^P. \quad (131)$$

Since the porosity and saturation can be computed in closed form, we use the following in our implementation

$$\begin{aligned} \phi^P(\bar{p}_p^w) &= (1 - S_0) \phi_0 \exp[\bar{\varepsilon}_v^P - \bar{\varepsilon}_v^a(\bar{p}_p^w)] + S_0 \phi_0 \exp[\bar{\varepsilon}_v^P - \bar{\varepsilon}_v^w(\bar{p}_p^w)] \\ S_w^P(\bar{p}_p^w) &= \frac{\mathcal{C}^P}{1 - S_0 + \mathcal{C}^P}, \quad \mathcal{C}^P := S_0 \exp[\bar{\varepsilon}_v^a(\bar{p}_p^w) - \bar{\varepsilon}_v^w(\bar{p}_p^w)]. \end{aligned} \quad (132)$$

## The hardening consistency bisection algorithm

**Algorithm 20** The consistency bisection algorithm for partially saturated materials

**Require:**  $\delta\boldsymbol{\varepsilon}, \boldsymbol{\varepsilon}^{P,k}, \boldsymbol{\sigma}^k, K^k, G^k, (\bar{p}^w)^k, \phi^k, S_w^k, X^k, \mathbf{s}^{\text{trial}}, I_1^{\text{eff,trial}}, \sqrt{J_2^{\text{trial}}}, r^{\text{trial}}, z_{\text{eff}}^{\text{trial}}, \varepsilon_v^{P,\text{trial}}, p_3^{\text{trial}}, a_1, a_2,$

$a_3, a_4, I_1^{\text{peak}}, R_c, \beta, i_{\text{max}}, j_{\text{max}}, \boldsymbol{\sigma}_{\text{fixed}}, \delta\boldsymbol{\varepsilon}_{\text{fixed}}^P, \text{yieldCondition}$

1: **procedure** CONSISTENCYBISECTION

2:  $\delta\varepsilon_v^{P,\text{fixed}} \leftarrow \text{tr}(\delta\boldsymbol{\varepsilon}_{\text{fixed}}^P)$

3:  $i \leftarrow 1$

4:  $\eta^{\text{in}} \leftarrow 0, \eta^{\text{out}} \leftarrow 1$

5: **while**  $\text{ABS}(\eta^{\text{out}} - \eta^{\text{in}}) > \text{TOLERANCE}$  **do**

6:  $j \leftarrow 1$

7:  $\text{isElastic} \leftarrow \text{TRUE}$

8: **while**  $\text{isElastic} = \text{TRUE}$  **do**

9:  $\eta^{\text{mid}} \leftarrow \frac{1}{2}(\eta^{\text{in}} + \eta^{\text{out}})$

10:  $\delta\varepsilon_v^{P,\text{mid}} \leftarrow \eta^{\text{mid}} \delta\varepsilon_v^{P,\text{fixed}}$

11:  $(\bar{p}^w)^{\text{mid}}, \phi^{\text{mid}}, S_w^{\text{mid}}, X^{\text{mid}} \leftarrow \text{COMPUTEINTERNALVARIABLES}(K^k, G^k, (\bar{p}^w)^k, \phi^k, S_w^k,$

$\hookrightarrow X^k, \delta\varepsilon_v^{P,\text{mid}})$   $\triangleright$  Update the internal variables using the bisected increment

$\hookrightarrow$  of the volumetric plastic strain

12:  $\text{isElastic} \leftarrow \text{yieldCondition.EVALYIELDCONDITION}(I_1^{\text{eff,trial}}, \sqrt{J_2^{\text{trial}}}, X^{\text{mid}}, (\bar{p}^w)^{\text{mid}},$

$\hookrightarrow \phi^{\text{mid}}, S_w^{\text{mid}}, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta)$

$\hookrightarrow \triangleright$  Determine whether the trial stress is elastic or not

```

13:         if isElastic = TRUE then
14:              $\eta^{\text{out}} \leftarrow \eta^{\text{mid}}$  ▷ If the local trial state is inside the updated yield surface, the yield
               ↪ condition evaluates to “elastic”. We need to reduce the size of the
               ↪ yield surface by decreasing the plastic strain increment.
15:              $j \leftarrow j + 1$ 
16:             if  $j \geq j_{\text{max}}$  then
17:                 return isSuccess  $\leftarrow$  FALSE ▷ The bisection algorithm failed because of
                   ↪ too many iterations.
18:             end if
19:         end if
20:     end while
21:      $\sigma_{\text{fixed}}^{\text{new}}, \delta \epsilon_{\text{fixed}}^{\text{p,new}} \leftarrow \text{NONHARDENINGRETURN}(\sigma^k, \delta \epsilon, X^{\text{mid}}, K^k, G^k, (\bar{p}^w)^{\text{mid}},$ 
       ↪  $s^{\text{trial}}, \sqrt{J_2^{\text{trial}}}, r^{\text{trial}}, z_{\text{eff}}^{\text{trial}}, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \beta)$ 
       ↪ ▷ Compute return to updated yield surface (no hardening)
22:     if  $\text{SIGN}(\text{tr}(\sigma^{\text{trial}} - \sigma_{\text{fixed}}^{\text{new}})) \neq \text{SIGN}(\text{tr}(\sigma^{\text{trial}} - \sigma_{\text{fixed}}))$  then
23:          $\eta^{\text{out}} \leftarrow \eta^{\text{mid}}$  ▷ Too much plastic strain
24:         continue
25:     end if
26:     if  $\|\delta \epsilon_{\text{fixed}}^{\text{p,new}}\| > \eta^{\text{mid}} \|\delta \epsilon_{\text{fixed}}^{\text{p}}\|$  then
27:          $\eta^{\text{in}} \leftarrow \eta^{\text{mid}}$  ▷ Too little plastic strain
28:     else
29:          $\eta^{\text{out}} \leftarrow \eta^{\text{mid}}$  ▷ Too much plastic strain
30:     end if
31:      $i \leftarrow i + 1$ 
32:     if  $i \geq i_{\text{max}}$  then
33:         return isSuccess  $\leftarrow$  FALSE ▷ Too many iterations
34:     end if
35: end while
36:  $\delta \epsilon_{v,\text{fixed}}^{\text{p},k+1} \leftarrow \text{tr}(\delta \epsilon_{\text{fixed}}^{\text{p},k+1})$ 
37:  $(\bar{p}^w)^{k+1}, \phi^{k+1}, S_w^{k+1}, X^{k+1} \leftarrow \text{COMPUTEINTERNALVARIABLES}(K^k, G^k, (\bar{p}^w)^k, \phi^k, S_w^k,$ 
       ↪  $X^k, \delta \epsilon_{v,\text{fixed}}^{\text{p},k+1}$  ▷ Update the internal variables using the bisected increment
       ↪ of the volumetric plastic strain
38:  $\sigma^{k+1} \leftarrow \sigma_{\text{fixed}}^{\text{new}}, \alpha^{k+1} \leftarrow -(\bar{p}^w)^{k+1} \mathbf{I}, p_3^{k+1} \leftarrow p_3^{\text{trial}}$ 
39:  $\epsilon^{\text{p},k+1} = \epsilon^{\text{p},k} + \delta \epsilon^{\text{p},k+1}$  ▷ Update the plastic strain
40:  $K^{k+1}, G^{k+1}, s^{k+1}, (\bar{p}^w)^{k+1}, I_1^{\text{eff},k+1}, \sqrt{J_2^{k+1}}, r^{k+1}, z_{\text{eff}}^{k+1}, \epsilon_v^{\text{p},k+1} \leftarrow$ 
       ↪  $\text{COMPUTEELASTICPROPERTIES}(\sigma^{k+1}, \phi^{k+1}, S_w^{k+1}, \epsilon^{\text{p},k+1}, \alpha^{k+1}, p_3^{k+1})$ 
       ↪ ▷ Compute elastic moduli and stress invariants for the new state
41: return isSuccess  $\leftarrow$  TRUE,  $\sigma^{k+1}, \epsilon^{\text{p},k+1}, \alpha^{k+1}, (\bar{p}^w)^{k+1}, \phi^{k+1}, S_w^{k+1}, X^{k+1}, K^{k+1}, G^{k+1},$ 
       ↪  $s^{k+1}, (\bar{p}^w)^{k+1}, I_1^{\text{eff},k+1}, \sqrt{J_2^{k+1}}, r^{k+1}, z_{\text{eff}}^{k+1}, \epsilon_v^{\text{p},k+1}, p_3^{k+1}$ 
42: end procedure

```

## Updating the internal variables

### Algorithm 21 Updating the internal variables for partially saturated materials

**Require:**  $\sigma^k, \epsilon_v^{\text{p},k}, K^k, G^k, (\bar{p}^w)^k, \phi^k, S_w^k, X^k, \delta \epsilon_v^{\text{p}}, \text{fluidParams}, \text{crushParams}, \text{airModel}, \text{waterModel}$

```

1: procedure COMPUTEINTERNALVARIABLES
2:      $\bar{\epsilon}_v^{\text{p},k} \leftarrow -\epsilon_v^{\text{p},k}, \delta \bar{\epsilon}_v^{\text{p}} \leftarrow -\delta \epsilon_v^{\text{p}}$ 
3:      $\bar{p}_0^w \leftarrow \text{fluidParams}.\bar{p}_0^w, S_0 \leftarrow \text{fluidParams}.S_0, \phi_0 \leftarrow \text{fluidParams}.\phi_0, p_1^{\text{sat}} \leftarrow \text{crushParams}.p_1^{\text{sat}}$ 
4:      $K_a \leftarrow \text{airModel}.\text{COMPUTE BULK MODULUS}((\bar{p}^w)^k)$ 
5:      $K_w \leftarrow \text{waterModel}.\text{COMPUTE BULK MODULUS}((\bar{p}^w)^k)$ 

```



```

6:    $\varepsilon_v^{a,0} \leftarrow \text{airModel.COMPUTEELASTICVOLUMETRICSTRAIN}(\overline{p_0^w})$ 
7:    $\varepsilon_v^a \leftarrow \text{airModel.COMPUTEELASTICVOLUMETRICSTRAIN}((\overline{p^w})^k)$ 
8:    $\varepsilon_v^w \leftarrow \text{waterModel.COMPUTEELASTICVOLUMETRICSTRAIN}((\overline{p^w})^k, \overline{p_0^w})$ 
9:    $\overline{\varepsilon_v^a} \leftarrow -(\varepsilon_v^a - \varepsilon_v^{a,0}), \overline{\varepsilon_v^w} \leftarrow -\varepsilon_v^w$ 
10:   $C_p \leftarrow S_0 \exp(\overline{\varepsilon_v^a} - \overline{\varepsilon_v^w})$ 
11:   $D_p \leftarrow \frac{1-S_0}{(1-S_0+C_p)^2}$ 
12:   $\frac{dC_p}{d\overline{p^w}} \leftarrow C_p \left[ \frac{1}{K_a} - \frac{1}{K_w} \right]$ 
13:   $G_a \leftarrow \exp(\overline{\varepsilon_v^{p,k}} - \overline{\varepsilon_v^a}), G_w \leftarrow \exp(\overline{\varepsilon_v^{p,k}} - \overline{\varepsilon_v^w})$ 
14:   $B_p \leftarrow \frac{1}{(1-S_0)G_a + S_0G_w} \left[ -\frac{(1-\phi^k)\phi^k}{\phi_0} \left( \frac{S_w^k}{K_w} + \frac{1-S_w^k}{K_a} \right) + \frac{1-S_0}{K_a} G_a + \frac{S_0}{K_w} G_w \right]$ 
15:   $(\overline{p^w})^{k+1} \leftarrow \text{MAX} \left[ (\overline{p^w})^k + \frac{1}{B_p} \delta \varepsilon_v^p, 0 \right] \triangleright \text{Update the pore pressure making sure that pressure does}$ 
    $\hookrightarrow$  not become negative during dilatative plastic deformations.
16:   $\overline{X}_d, \frac{d\overline{X}_d}{d\varepsilon_v^p} \leftarrow \text{COMPUTEDRAINEDHYDROSTATICSTRENGTHANDDERIV}(\overline{\varepsilon_v^{p,k}})$ 
    $\hookrightarrow$   $\triangleright$  Compute the drained hydrostatic compressive strength and its derivative
17:   $\overline{X}^{k+1} = -X^k + \left[ (1 - S_w^k + p_1^{\text{sat}} S_w^k) \frac{d\overline{X}_d}{d\varepsilon_v^p} + \overline{X}_d (p_1^{\text{sat}} - 1) \frac{D_p}{B_p} \frac{dC_p}{d\overline{p^w}} + \frac{3}{B_p} \right] \delta \varepsilon_v^p$ 
    $\hookrightarrow$   $\triangleright$  Update the hydrostatic compressive strength
18:   $\overline{X}^{k+1} \leftarrow -\overline{X}^{k+1}$ 
19:   $\varepsilon_v^{p,k+1} \leftarrow \varepsilon_v^{p,k} + \delta \varepsilon_v^p \triangleright \text{Compute the updated volumetric plastic strain.}$ 
20:   $\varepsilon_v^{a,k+1} \leftarrow \text{airModel.COMPUTEELASTICVOLUMETRICSTRAIN}((\overline{p^w})^{k+1})$ 
21:   $\varepsilon_v^{w,k+1} \leftarrow \text{waterModel.COMPUTEELASTICVOLUMETRICSTRAIN}((\overline{p^w})^{k+1}, \overline{p_0^w})$ 
22:   $\overline{\varepsilon_v^{a,k+1}} \leftarrow -(\varepsilon_v^{a,k+1} - \varepsilon_v^{a,0}), \overline{\varepsilon_v^{w,k+1}} \leftarrow -\varepsilon_v^{w,k+1} \triangleright \text{The updated strains in the fluid phases.}$ 
23:   $C_p^{k+1} \leftarrow S_0 \exp(\overline{\varepsilon_v^{a,k+1}} - \overline{\varepsilon_v^{w,k+1}})$ 
24:   $S_w^{k+1} \leftarrow \frac{C_p^{k+1}}{1-S_0+C_p^{k+1}} \triangleright \text{Update the saturation}$ 
25:   $G_a^{k+1} \leftarrow \exp(\overline{\varepsilon_v^{p,k+1}} - \overline{\varepsilon_v^{a,k+1}}), G_w^{k+1} \leftarrow \exp(\overline{\varepsilon_v^{p,k+1}} - \overline{\varepsilon_v^{w,k+1}})$ 
26:   $\phi^{k+1} \leftarrow (1 - S_0)\phi_0 G_a^{k+1} + S_0\phi_0 G_w^{k+1} \triangleright \text{Update the porosity}$ 
27:  return  $(\overline{p^w})^{k+1}, \phi^{k+1}, S_w^{k+1}, \overline{X}^{k+1}$ 
28: end procedure

```

---

**Algorithm 22** Computing the drained hydrostatic strength and its derivative

---

**Require:**  $\overline{\varepsilon_v^{p,k}}, \text{fluidParams}, \text{crushParams}$

```

1: procedure COMPUTEDRAINEDHYDROSTATICSTRENGTHANDDERIV
2:    $\phi_0 \leftarrow \text{fluidParams}.\phi_0$ 
3:    $p_0 \leftarrow \text{crushParams}.p_0, p_1 \leftarrow \text{crushParams}.p_1, p_1^{\text{sat}} \leftarrow \text{crushParams}.p_1^{\text{sat}}, p_2 \leftarrow \text{crushParams}.p_2$ 
4:    $\overline{p}_3 \leftarrow -\log(1 - \phi_0)$ 
5:    $\overline{X}_d \leftarrow \text{MAX}(p_0, 1000); \triangleright \overline{X}_d \text{ has a minimum value of 1000 pressure units}$ 
6:    $\frac{d\overline{X}_d}{d\varepsilon_v^p} \leftarrow 0$ 
7:   if  $\varepsilon_v^{p,k} > 0$  then
8:      $\phi_{\text{temp}} \leftarrow \exp(-p_3 + \overline{\varepsilon_v^{p,k}})$ 
9:      $\phi \leftarrow 1 - \phi_{\text{temp}}$ 
10:     $\tilde{\xi} \leftarrow p_1 \text{POW}\left(\frac{\phi_0}{\phi} - 1, \frac{1}{p_2}\right)$ 
11:     $\overline{X}_d \leftarrow \overline{X}_d + \tilde{\xi}$ 

```

```

12:    $\frac{d\bar{X}_d}{d\varepsilon_v^p} \leftarrow \frac{1}{p_2} \frac{\phi_0}{\phi} \phi_{\text{temp}} \frac{\bar{\xi}}{\phi \left( \frac{\phi_0}{\phi} - 1 \right)}$ 
13:   end if
14:   return  $\bar{X}_d, \frac{d\bar{X}_d}{d\varepsilon_v^p}$ 
15: end procedure

```

### 3.2.8 Rate-dependent plastic update

Recall from (74) and (77) that

$$\dot{\sigma}_{\text{over}} = \dot{\sigma}_{\text{proj}} - \frac{1}{\tau} \sigma_{\text{over}} \quad \text{where} \quad \dot{\sigma}_{\text{over}} := \dot{\sigma} - \dot{\sigma}_{qs} \quad \text{and} \quad \dot{\sigma}_{\text{proj}} := \dot{\sigma}^{\text{trial}} - \dot{\sigma}_{qs}. \quad (133)$$

and the exact solution (82) when  $\tau$  and  $\dot{\sigma}_{\text{proj}}$  are constant:

$$\sigma_{\text{over}}(t) = \tau \left[ 1 - \exp\left(-\frac{t_0 - t}{\tau}\right) \right] \dot{\sigma}_{\text{proj}} + \exp\left(-\frac{t_0 - t}{\tau}\right) \sigma_{\text{over}}(t_0). \quad (134)$$

For a timestep  $\Delta t = t_{n+1} - t_n$ , we can write the above as

$$\sigma_{\text{over}}(t_{n+1}) = \tau(t_n) \left[ 1 - \exp\left(-\frac{t_n - t_{n+1}}{\tau(t_n)}\right) \right] \dot{\sigma}_{\text{proj}}(t_n) + \exp\left(-\frac{t_n - t_{n+1}}{\tau(t_n)}\right) \sigma_{\text{over}}(t_n) \quad (135)$$

or

$$\sigma_{\text{over}}^{n+1} = \left[ \frac{1 - \exp(-\Delta t / \tau^n)}{\Delta t / \tau^n} \right] (\Delta t \dot{\sigma}_{\text{proj}}^n) + \exp(-\Delta t / \tau^n) \sigma_{\text{over}}^n. \quad (136)$$

If we approximate the rate of  $\sigma_{\text{proj}}$  via approximations of the rates of the trial and quasistatic stresses using

$$\dot{\sigma}_{\text{proj}} \approx \frac{\sigma_{n+1}^{\text{trial}} - \sigma_n^{\text{trial}}}{\Delta t} - \frac{\sigma_{qs}^{n+1} - \sigma_{qs}^n}{\Delta t} \quad (137)$$

noting that

$$\sigma_n^{\text{trial}} = \sigma^n, \quad \sigma_{n+1}^{\text{trial}} = \sigma^n + (\mathbf{C}^e : \mathbf{d}^s) \Delta t, \quad (138)$$

and using the definition of  $\sigma_{\text{over}}$ , we have

$$\sigma^{n+1} - \sigma_{qs}^{n+1} = \left[ \frac{1 - \exp(-\Delta t / \tau^n)}{\Delta t / \tau^n} \right] [(\sigma_{n+1}^{\text{trial}} - \sigma^n) - (\sigma_{qs}^{n+1} - \sigma_{qs}^n)] + \exp(-\Delta t / \tau^n) (\sigma^n - \sigma_{qs}^n). \quad (139)$$

If we define

$$R_H := \frac{1 - \exp(-\Delta t / \tau^n)}{\Delta t / \tau^n} \quad \text{and} \quad r_h := \exp(-\Delta t / \tau^n) \quad (140)$$

we have the dynamic stress state

$$\sigma^{n+1} = \sigma_{qs}^{n+1} + [(\sigma_{n+1}^{\text{trial}} - \sigma^n) - (\sigma_{qs}^{n+1} - \sigma_{qs}^n)] R_H + (\sigma^n - \sigma_{qs}^n) r_h. \quad (141)$$

Our implementation does not consider rate-dependent updates of the internal variables. Equation 141 requires that we compute the trial stress for the step, but this is not known since the bulk modulus is evolving through the substeps. It would be necessary to loop through the substeps to compute the trial stress assuming nonlinear elasticity, but instead we approximate the trial stress using the average of the elastic moduli at the start and end of the step.

---

**Algorithm 23** Computing the correction to the stress due to rate-dependent plasticity

---

**Require:**  $\Delta t, \mathbf{d}^{n+1}, \boldsymbol{\sigma}^n, K^n, G^n, \phi^n, S_w^n, X^n, \boldsymbol{\alpha}^n, \boldsymbol{\varepsilon}^{p,n}, p_3^n, \boldsymbol{\sigma}_{qs}^n, \boldsymbol{\sigma}_{qs}^{n+1}, K^{n+1}, G^{n+1}, \phi^{n+1}, S_w^{n+1}, X^{n+1}, \boldsymbol{\alpha}^{n+1}, \boldsymbol{\varepsilon}^{p,n+1}, p_3^{n+1}, a_1, a_2, a_3, a_4, I_1^{\text{peak}}, R_c, \text{yieldParams}$

- 1: **procedure** RATEDEPENDENTPLASTICUPDATE
- 2:    $T_1 \leftarrow \text{yieldParams}.T_1, T_2 \leftarrow \text{yieldParams}.T_2$
- 3:   **if**  $T_1 = 0$  **or**  $T_2 = 0$  **then**  $\triangleright$ Check if rate-dependent plasticity has been turned on
- 4:     **return** isRateDependent  $\leftarrow$  FALSE,  $\boldsymbol{\sigma}_{qs}^{n+1}$
- 5:   **end if**
- 6:    $K_{\text{dyn}} \leftarrow \frac{1}{2}(K^n + K^{n+1}), G_{\text{dyn}} \leftarrow \frac{1}{2}(G^n + G^{n+1})$   $\triangleright$ Compute mid-step bulk and shear modulus
- 7:    $\Delta \boldsymbol{\varepsilon} \leftarrow \Delta t \mathbf{d}^{n+1}$
- 8:    $\boldsymbol{\sigma}_{\text{trial,dyn}} \leftarrow \text{COMPUTETRIALSTRESS}(\boldsymbol{\sigma}^n, K_{\text{dyn}}, G_{\text{dyn}}, \mathbf{d}^{n+1}, \Delta t)$   $\triangleright$ Compute substep trial stress
- 9:    $\dot{\boldsymbol{\varepsilon}} \leftarrow \text{MAX}(\|\mathbf{d}^{n+1}\|, \text{ABS\_DOUBLE\_MIN})$
- 10:    $\tau \leftarrow T_1 \text{ POW}(\dot{\boldsymbol{\varepsilon}}, T_2)$   $\triangleright$ The characteristic time is defined from the rate-dependence  
      $\rightarrow$  input parameters and the magnitude of the strain rate
- 11:    $r_h \leftarrow \exp\left(-\frac{\Delta t}{\tau}\right)$
- 12:    $R_H \leftarrow \frac{1-r_h}{\frac{\Delta t}{\tau}}$
- 13:    $\boldsymbol{\sigma}^{n+1} \leftarrow \boldsymbol{\sigma}_{qs}^{n+1} + [(\boldsymbol{\sigma}_{\text{trial,dyn}} - \boldsymbol{\sigma}^n) - (\boldsymbol{\sigma}_{qs}^{n+1} - \boldsymbol{\sigma}_{qs}^n)] R_H + (\boldsymbol{\sigma}^n - \boldsymbol{\sigma}_{qs}^n) r_h$   $\triangleright$ Stress update
- 14:   **return**  $\boldsymbol{\sigma}^{n+1}$ , isRateDependent  $\leftarrow$  TRUE
- 15: **end procedure**

---

## References

- Berryman, J. G. (2006). “Effective medium theories for multicomponent poroelastic composites”. In: *Journal of Engineering Mechanics* 132.5, pp. 519–531 (cit. on p. 45).
- Berryman, J. G. and G. W. Milton (1991). “Exact results for generalized Gassmann’s equations in composite porous media with two constituents”. In: *Geophysics* 56.12, pp. 1950–1960 (cit. on p. 45).
- Borja, R. I. (2004). “Cam-Clay plasticity. Part V: A mathematical framework for three-phase deformation and strain localization analyses of partially saturated porous media”. In: *Computer Methods in Applied Mechanics and Engineering* 193.48, pp. 5301–5338 (cit. on p. 37).
- (2006). “On the mechanical energy and effective stress in saturated and unsaturated porous continua”. In: *International Journal of Solids and Structures* 43.6, pp. 1764–1786 (cit. on p. 37).
- Borja, R. I. and J. A. White (2010). “Conservation laws for coupled hydromechanical processes in unsaturated porous media: Theory and implementation”. In: *Mechanics of Unsaturated Geomaterials*, pp. 185–208 (cit. on p. 37).
- Brannon, R. M. (2007). “Elements of phenomenological plasticity: geometrical insight, computational algorithms, and topics in shock physics”. In: *Shock Wave Science and Technology Reference Library*. Springer, pp. 225–274 (cit. on pp. 8, 12, 14).
- Dvorkin, J. et al. (1999). “Identifying patchy saturation from well logs”. In: *Geophysics* 64.6, pp. 1756–1759 (cit. on pp. 19, 45).
- Grujicic, M. et al. (2009). “Development, parameterization, and validation of a visco-plastic material model for sand with different levels of water saturation”. In: *Proceedings of the Institution of Mechanical Engineers, Part L: Journal of Materials Design and Applications* 223.2, pp. 63–81 (cit. on p. 11).
- Hassanizadeh, S. M. and W. G. Gray (1990). “Mechanics and thermodynamics of multiphase flow in porous media including interphase boundaries”. In: *Advances in water resources* 13.4, pp. 169–186 (cit. on pp. 7, 42).
- Homel, M. A., J. E. Guilkey, and R. M. Brannon (2015). “Continuum effective-stress approach for high-rate plastic deformation of fluid-saturated geomaterials with application to shaped-charge jet penetration”. In: *Acta Mechanica*, pp. 1–32 (cit. on pp. 7, 21, 24, 25, 29).

- Li, C., R. I. Borja, and R. A. Regueiro (2004). “Dynamics of porous media at finite strain”. In: *Computer methods in applied mechanics and engineering* 193.36, pp. 3837–3870 (cit. on p. 37).
- Norris, A. N. (2008). “Eulerian conjugate stress and strain”. In: *Journal of Mechanics of Materials and Structures* 3.2, pp. 243–260 (cit. on p. 8).
- Pabst, W and E Gregorová (2015). “Critical Assessment 18: elastic and thermal properties of porous materials—rigorous bounds and cross-property relations”. In: *Materials Science and Technology* 31.15, pp. 1801–1808 (cit. on p. 52).
- Song, X. and R. I. Borja (2014a). “Finite deformation and fluid flow in unsaturated soils with random heterogeneity”. In: *Vadose Zone Journal* 13.5 (cit. on p. 37).
- (2014b). “Mathematical framework for unsaturated flow in the finite deformation range”. In: *International Journal for Numerical Methods in Engineering* 97.9, pp. 658–682 (cit. on p. 37).
- Uzuoka, R. and R. I. Borja (2012). “Dynamics of unsaturated poroelastic solids at finite strain”. In: *International Journal for Numerical and Analytical Methods in Geomechanics* 36.13, pp. 1535–1573 (cit. on pp. 6, 37).

# Appendices

## A Governing equations from mixture theory

An alternative way of determining the governing equations is using mixture theory, of which there are several alternatives. We explore the Borja et al. (Borja, 2004; Li, Borja, and Regueiro, 2004; Borja, 2006; Borja and White, 2010; Uzuoka and Borja, 2012; Song and Borja, 2014b; Song and Borja, 2014a) approach in this appendix.

### A.1 Volume fraction, porosity, saturation, density

Let us once again consider a three-phase porous medium with constituents  $\alpha = \{s, w, a\}$  where  $s$  is the solid skeleton,  $w$  is water, and  $a$  is air. Let  $\mathbf{X}$  and  $\mathbf{x}$  be the reference and current positions, respectively, of a point inside a body  $\Omega$ . Each point may be considered to be the centroid of a representative volume element (RVE) region ( $\omega$ ) of volume  $dv$  and mass  $dm$ . The region ( $\omega^\alpha \subset \omega$ ) occupied by phase  $\alpha$  at each point in the body has a volume  $dv^\alpha$  and mass  $dm^\alpha$  such that

$$dv = dv^s + dv^w + dv^a \quad \text{and} \quad dm = dm^s + dm^w + dm^a. \quad (142)$$

. The volume fraction ( $f^\alpha$ ) of phase  $\alpha$  at each point is defined as

$$f^\alpha = \frac{dv^\alpha}{dv} \quad \text{where} \quad \sum_{\alpha} f^\alpha = 1. \quad (143)$$

The volume fraction of the solid skeleton is

$$f^s = 1 - \phi, \quad \phi = \frac{dv^w + dv^a}{dv} \quad (144)$$

where the  $\phi$  is the porosity. The volume fractions of water and air are

$$f^w = \phi S_w, \quad f^a = \phi S_a = \phi(1 - S_w), \quad S_w = \frac{dv^w}{dv^w + dv^a} \quad (145)$$

where  $S_w$  is the saturation. Let the mixture mass density ( $\rho$ ) and the mass density of each phase ( $\rho^\alpha$ ) be defined as

$$\rho = \frac{dm}{dv} \quad \text{and} \quad \rho^\alpha = \frac{dm^\alpha}{dv^\alpha}. \quad (146)$$

We define the *partial mass density* of each phase as

$$\langle \rho^\alpha \rangle = f^\alpha \rho^\alpha \quad \text{such that} \quad \langle \rho^s \rangle + \langle \rho^w \rangle + \langle \rho^a \rangle = \rho. \quad (147)$$

### A.2 Material derivatives

As before, the material time derivative of  $\mathbf{f}^\alpha$  with respect to phase  $\alpha$  is defined as

$$\frac{D^\alpha \mathbf{f}^\alpha}{Dt} = \frac{\partial \mathbf{f}^\alpha}{\partial t} + \nabla \mathbf{f}^\alpha \cdot \mathbf{v}^\alpha \quad (148)$$

where  $\mathbf{v}^\alpha$  is the velocity of phase  $\alpha$ . The relative material derivative of phase  $\beta$  with respect to phase  $\alpha$  is

$$\frac{D^\alpha \mathbf{f}^\beta}{Dt} = \frac{D^\beta \mathbf{f}^\beta}{Dt} + \nabla \mathbf{f}^\beta \cdot (\mathbf{v}^\alpha - \mathbf{v}^\beta) = \frac{D^\beta \mathbf{f}^\beta}{Dt} + \nabla \mathbf{f}^\beta \cdot \mathbf{v}^{\alpha\beta}. \quad (149)$$

### A.3 Mass balance

As discussed in the appendix on the averaging approach, the phase mass balance equation for the RVE in the absence of mass exchange between the phases is

$$\frac{\partial}{\partial t} (\langle \rho^\alpha \rangle) + \nabla \cdot (\langle \rho^\alpha \rangle \mathbf{v}^\alpha) = 0. \quad (150)$$

Using the definition of the material derivative, we can write the mass balance equations as

$$\frac{D^\alpha}{Dt} (\langle \rho^\alpha \rangle) + \langle \rho^\alpha \rangle \nabla \cdot \mathbf{v}^\alpha = 0. \quad (151)$$

Alternatively, relative to phase  $\beta$ ,

$$\frac{D^\beta}{Dt} (\langle \rho^\alpha \rangle) - \nabla \langle \rho^\alpha \rangle \cdot \mathbf{v}^{\beta\alpha} + \langle \rho^\alpha \rangle \nabla \cdot \mathbf{v}^\alpha = 0. \quad (152)$$

#### A.3.1 Solid phase

For the solid phase,

$$\frac{D^s}{Dt} (\langle \rho^s \rangle) + \langle \rho^s \rangle \nabla \cdot \mathbf{v}^s = 0. \quad (153)$$

Expanding out in terms of volume fractions,

$$\frac{D^s f^s}{Dt} + \frac{f^s}{\rho^s} \frac{D^s \rho^s}{Dt} + f^s \nabla \cdot \mathbf{v}^s = 0. \quad (154)$$

Assume a smooth function  $\rho^s = \rho^s(p^s)$ , where  $p^s$  is the intrinsic pressure in the solid phase (actual force per unit area acting on the solid phase - without considering voids). Then

$$\frac{D^s f^s}{Dt} + \frac{f^s}{\rho^s} \frac{d\rho^s}{dp^s} \frac{D^s p^s}{Dt} + f^s \nabla \cdot \mathbf{v}^s = 0. \quad (155)$$

The bulk modulus of the solid phase is defined as

$$K_s(p^s) = -\frac{dp^s}{d\varepsilon_v^e} = -\frac{dp^s}{d\rho^s} \frac{d\rho^s}{d\varepsilon_v^e} = \rho^s \frac{dp^s}{d\rho^s} \quad \text{where} \quad \varepsilon_v^e = \ln \left( \frac{v^s}{v_0^s} \right) = -\ln \left( \frac{\rho^s}{\rho_0^s} \right) = \ln J^s. \quad (156)$$

We can use the above definition to get

$$\frac{D^s f^s}{Dt} + \frac{f^s}{K_s(p^s)} \frac{D^s p^s}{Dt} + f^s \nabla \cdot \mathbf{v}^s = 0. \quad (157)$$

Note that there is some ambiguity here about the conservation of intrinsic mass and the definition of intrinsic volume ( $v^s$ ).

For our purposes, it is more convenient to work with a solid phase pressure that is defined over a RVE containing voids. Borja and coworkers define this pressure as

$$p^s = \tilde{p}^s(\rho^s, f^s) \quad (158)$$

such that

$$\frac{D^s p^s}{Dt} = \frac{\partial \tilde{p}^s}{\partial \rho^s} \frac{D^s \rho^s}{Dt} + \frac{\partial \tilde{p}^s}{\partial f^s} \frac{D^s f^s}{Dt}. \quad (159)$$

Now,

$$\frac{D^s \varepsilon_v^e}{Dt} = -\frac{1}{\rho^s} \frac{D^s \rho^s}{Dt} = \frac{D^s}{Dt} (\ln J^s) = \frac{1}{J^s} \frac{D^s J^s}{Dt}. \quad (160)$$

Recall that

$$\frac{D^s J^s}{Dt} = J^s \nabla \cdot \mathbf{v}^s. \quad (161)$$

Therefore,

$$\frac{D^s \rho^s}{Dt} = -\rho^s \nabla \cdot \mathbf{v}^s \quad (162)$$

Plugging into (159) gives

$$\frac{D^s p^s}{Dt} = -\rho^s \frac{\partial \tilde{p}^s}{\partial \rho^s} \nabla \cdot \mathbf{v}^s + \frac{\partial \tilde{p}^s}{\partial f^s} \frac{D^s f^s}{Dt}. \quad (163)$$

Therefore,

$$\frac{D^s f^s}{Dt} = \frac{1}{\frac{\partial \tilde{p}^s}{\partial f^s}} \left[ \frac{D^s p^s}{Dt} + \rho^s \frac{\partial \tilde{p}^s}{\partial \rho^s} \nabla \cdot \mathbf{v}^s \right]. \quad (164)$$

Plugging this into the mass balance equation (157) for the solid phase,

$$\frac{D^s p^s}{Dt} + \rho^s \frac{\partial \tilde{p}^s}{\partial \rho^s} \nabla \cdot \mathbf{v}^s + \frac{f^s}{K_s(p^s)} \frac{\partial \tilde{p}^s}{\partial f^s} \frac{D^s p^s}{Dt} + f^s \frac{\partial \tilde{p}^s}{\partial f^s} \nabla \cdot \mathbf{v}^s = 0 \quad (165)$$

or

$$\left( 1 + \frac{f^s}{K_s(p^s)} \frac{\partial \tilde{p}^s}{\partial f^s} \right) \frac{D^s p^s}{Dt} + \left( \rho^s \frac{\partial \tilde{p}^s}{\partial \rho^s} + f^s \frac{\partial \tilde{p}^s}{\partial f^s} \right) \nabla \cdot \mathbf{v}^s = 0 \quad (166)$$

or,

$$\frac{D^s p^s}{Dt} = - \left( \frac{\rho^s \frac{\partial \tilde{p}^s}{\partial \rho^s} + f^s \frac{\partial \tilde{p}^s}{\partial f^s}}{1 + \frac{f^s}{K_s(p^s)} \frac{\partial \tilde{p}^s}{\partial f^s}} \right) \nabla \cdot \mathbf{v}^s. \quad (167)$$

Let us define the bulk modulus of the solid matrix ( $\tilde{K}^s$ ) using

$$\frac{D^s \tilde{p}^s}{Dt} = \frac{\partial \tilde{p}^s}{\partial \varepsilon_v^e} \frac{D^s \varepsilon_v^e}{Dt} =: -\tilde{K}^s(\tilde{p}^s) \nabla \cdot \mathbf{v}^s \quad (168)$$

In terms of the intrinsic pressure in the solid, Borja's definition leads to

$$f^s \frac{D^s p^s}{Dt} = -\tilde{K}^s(\tilde{p}^s) \nabla \cdot \mathbf{v}^s. \quad (169)$$

The reason for this definition is not obvious and needs re-examination but leads to the definition of the matrix bulk modulus used by Borja:

$$\tilde{K}^s := f^s \left( \frac{\rho^s \frac{\partial \tilde{p}^s}{\partial \rho^s} + f^s \frac{\partial \tilde{p}^s}{\partial f^s}}{1 + \frac{f^s}{K_s(p^s)} \frac{\partial \tilde{p}^s}{\partial f^s}} \right). \quad (170)$$

If we substitute (169) into (157), we get

$$\frac{D^s f^s}{Dt} - \frac{\tilde{K}^s(\tilde{p}^s)}{K_s(p^s)} \nabla \cdot \mathbf{v}^s + f^s \nabla \cdot \mathbf{v}^s = 0. \quad (171)$$

Defining

$$B := 1 - \frac{\tilde{K}^s(\tilde{p}^s)}{K_s(p^s)} \quad (172)$$

we have the solid mass balance equation

$$\frac{D^s f^s}{Dt} + (B - 1 + f^s) \nabla \cdot \mathbf{v}^s = 0. \quad (173)$$

### A.3.2 Water phase

For the water phase, relative to the solid phase, with  $\mathbf{v}^{sw} = \mathbf{v}^s - \mathbf{v}^w$ ,

$$\frac{D^s}{Dt} (\langle \rho^w \rangle) - \nabla \langle \rho^w \rangle \cdot \mathbf{v}^{sw} + \langle \rho^w \rangle \nabla \cdot \mathbf{v}^w = 0. \quad (174)$$

Expanded,

$$\frac{D^s f^w}{Dt} + \frac{f^w}{\rho^w} \frac{D^s \rho^w}{Dt} - \left[ \frac{f^w}{\rho^w} \nabla \rho^w + \nabla f^w \right] \cdot \mathbf{v}^{sw} + f^w \nabla \cdot \mathbf{v}^w = 0. \quad (175)$$

Once again we assume the existence of a smooth function  $\rho^w = \rho^w(p^w)$  where  $p^w$  is the intrinsic water pressure and get

$$\frac{D^s f^w}{Dt} + \frac{f^w}{\rho^w} \frac{d\rho^w}{dp^w} \frac{D^s p^w}{Dt} - \left[ \frac{f^w}{\rho^w} \frac{d\rho^w}{dp^w} \nabla p^w + \nabla f^w \right] \cdot \mathbf{v}^{sw} + f^w \nabla \cdot \mathbf{v}^w = 0. \quad (176)$$

As was done for the solid phase, the bulk modulus of water is defined as

$$K_w(p^w) = \rho^w \frac{dp^w}{d\rho^w} \quad (177)$$

which leads to

$$\frac{D^s f^w}{Dt} + \frac{f^w}{K_w(p^w)} \frac{D^s p^w}{Dt} - \left[ \frac{f^w}{K_w(p^w)} \nabla p^w + \nabla f^w \right] \cdot \mathbf{v}^{sw} + f^w \nabla \cdot \mathbf{v}^w = 0. \quad (178)$$

Recalling from the definitions of porosity and saturation that

$$f^w = \phi S_w = (1 - f^s) S_w \quad (179)$$

and using the mass balance equation for the solid phase (157), we have

$$\frac{D^s f^w}{Dt} = -S_w \frac{D^s f^s}{Dt} + (1 - f^s) \frac{D^s S_w}{Dt} = S_w \left[ \frac{f^s}{K_s(p^s)} \frac{D^s p^s}{Dt} + f^s \nabla \cdot \mathbf{v}^s \right] + (1 - f^s) \frac{D^s S_w}{Dt}. \quad (180)$$

Therefore the mass balance of water can be expressed as

$$(1 - f^s) \frac{D^s S_w}{Dt} + \frac{f^w}{K_w(p^w)} \frac{D^s p^w}{Dt} + \frac{S_w f^s}{K_s(p^s)} \frac{D^s p^s}{Dt} + S_w f^s \nabla \cdot \mathbf{v}^s - \left[ \frac{f^w}{K_w(p^w)} \nabla p^w + \nabla f^w \right] \cdot \mathbf{v}^{sw} + f^w \nabla \cdot \mathbf{v}^w = 0. \quad (181)$$

Recall from equation (169) that

$$f^s \frac{D^s p^s}{Dt} = -\tilde{K}^s(\tilde{p}^s) \nabla \cdot \mathbf{v}^s. \quad (182)$$

Plugging this into the mass balance equation for water gives

$$(1 - f^s) \frac{D^s S_w}{Dt} + \frac{f^w}{K_w(p^w)} \frac{D^s p^w}{Dt} - \frac{S_w \tilde{K}^s(\tilde{p}^s)}{K_s(p^s)} \nabla \cdot \mathbf{v}^s + S_w f^s \nabla \cdot \mathbf{v}^s - \left[ \frac{f^w}{K_w(p^w)} \nabla p^w + \nabla f^w \right] \cdot \mathbf{v}^{sw} + f^w \nabla \cdot \mathbf{v}^w = 0. \quad (183)$$



Using the definition

$$B := 1 - \frac{\tilde{K}^s(\tilde{p}^s)}{K_s(p^s)} \quad (184)$$

we have

$$(1-f^s) \frac{D^s S_w}{Dt} + \frac{f^w}{K_w(p^w)} \frac{D^s p^w}{Dt} + S_w(B-1+f^s) \nabla \cdot \mathbf{v}^s - \left[ \frac{f^w}{K_w(p^w)} \nabla p^w + \nabla f^w \right] \cdot \mathbf{v}^{sw} + f^w \nabla \cdot \mathbf{v}^w = 0. \quad (185)$$

Using

$$\nabla \cdot \mathbf{v}^w = -\nabla \cdot \mathbf{v}^{sw} + \nabla \cdot \mathbf{v}^s \quad (186)$$

and

$$S_w(B-1+f^s) + f^w = S_w B - S_w + S_w(1-\phi) + S_w \phi = S_w B \quad (187)$$

we get

$$(1-f^s) \frac{D^s S_w}{Dt} + \frac{f^w}{K_w(p^w)} \frac{D^s p^w}{Dt} + B S_w \nabla \cdot \mathbf{v}^s - \left[ \frac{f^w}{K_w(p^w)} \nabla p^w + \nabla f^w \right] \cdot \mathbf{v}^{sw} - f^w \nabla \cdot \mathbf{v}^{sw} = 0. \quad (188)$$

### A.3.3 Air phase

For the air phase, relative to the solid phase, with  $\mathbf{v}^{sa} = \mathbf{v}^s - \mathbf{v}^a$ ,

$$\frac{D^s}{Dt} (\langle \rho^a \rangle) - \nabla \langle \rho^a \rangle \cdot \mathbf{v}^{sa} + \langle \rho^a \rangle \nabla \cdot \mathbf{v}^a = 0 \quad (189)$$

or

$$\frac{D^s f^a}{Dt} + \frac{f^a}{\rho^a} \frac{D^s \rho^a}{Dt} - \left[ \frac{f^a}{\rho^a} \nabla \rho^a + \nabla f^a \right] \cdot \mathbf{v}^{sa} + f^a \nabla \cdot \mathbf{v}^a = 0. \quad (190)$$

For the air phase, with

$$\rho^a = \rho^a(p^a) \quad \text{and} \quad K_a(p^a) = \rho^a \frac{dp^a}{d\rho^a} \quad (191)$$

we get

$$\frac{D^s f^a}{Dt} + \frac{f^a}{K_a(p^a)} \frac{D^s p^a}{Dt} - \left[ \frac{f^a}{K_a(p^a)} \nabla p^a + \nabla f^a \right] \cdot \mathbf{v}^{sa} + f^a \nabla \cdot \mathbf{v}^a = 0. \quad (192)$$

From the definitions of porosity and saturation,

$$f^a = \phi S_a = (1-f^s)(1-S_w). \quad (193)$$

Using the procedure used for the water phase, the mass balance of air is

$$(1-f^s) \frac{D^s S_a}{Dt} + \frac{f^a}{K_a(p^a)} \frac{D^s p^a}{Dt} + \frac{S_a f^s}{K_s(p^s)} \frac{D^s p^s}{Dt} + S_a f^s \nabla \cdot \mathbf{v}^s - \left[ \frac{f^a}{K_a(p^a)} \nabla p^a + \nabla f^a \right] \cdot \mathbf{v}^{sa} + f^a \nabla \cdot \mathbf{v}^a = 0. \quad (194)$$

In terms of the matrix bulk modulus,

$$(1-f^s) \frac{D^s S_a}{Dt} + \frac{f^a}{K_a(p^a)} \frac{D^s p^a}{Dt} + B S_a \nabla \cdot \mathbf{v}^s - \left[ \frac{f^a}{K_a(p^a)} \nabla p^a + \nabla f^a \right] \cdot \mathbf{v}^{sa} - f^a \nabla \cdot \mathbf{v}^{sa} = 0. \quad (195)$$

## A.4 Linear momentum balance

The linear momentum balance for the three phases in the absence of mass exchange between phases can be written as

$$\langle \rho^\alpha \rangle \frac{D^\alpha \mathbf{v}^\alpha}{Dt} = \nabla \cdot \boldsymbol{\sigma}^\alpha + \langle \rho^\alpha \rangle \mathbf{b}^\alpha + \mathbf{p}^\alpha + \mathbf{t}^\alpha, \quad \sum_\alpha (\mathbf{p}^\alpha + \mathbf{t}^\alpha) = \mathbf{0} \quad (196)$$

where  $\boldsymbol{\sigma}^\alpha$  is the average stress of phase  $\alpha$  in the RVE,  $\mathbf{b}^\alpha$  is the body force experience by phase  $\alpha$ , and  $\mathbf{p}^\alpha$  is the interaction force term due to the motion of the phases and  $\mathbf{t}^\alpha$  is the interaction force vector due to surface tractions, between phase  $\alpha$  and the other phases. Also note that the total stress is given by

$$\boldsymbol{\sigma} = \sum_\alpha \boldsymbol{\sigma}^\alpha. \quad (197)$$

Borja uses expressions for  $\mathbf{p}^\alpha$  and  $\mathbf{t}^\alpha$  similar (but not identical) to that derived by Haasanizadeh and Gray (Hassanizadeh and Gray, 1990) by the application of an averaged energy balance and a reduced entropy inequality to the RVE balance equations discussed in the previous appendix. These expressions are

$$\mathbf{t}^\alpha = p^\alpha \nabla f^\alpha, \quad \mathbf{p}^\alpha = -f^\alpha \boldsymbol{\mu}^\alpha \cdot (\mathbf{v}^\alpha - \mathbf{v}^s) \quad (198)$$

where  $\boldsymbol{\mu}^\alpha$  is a material parameter related to the relative permeability of the phases.

### A.4.1 Solid phase

The material time derivative for the solid phase is

$$\mathbf{a}^s = \frac{D^s \mathbf{v}^s}{Dt} = \frac{\partial \mathbf{v}^s}{\partial t} + \nabla \mathbf{v}^s \cdot \mathbf{v}^s. \quad (199)$$

The momentum equation for the solid phase is

$$\langle \rho^s \rangle \mathbf{a}^s = \nabla \cdot \boldsymbol{\sigma}^s + \langle \rho^s \rangle \mathbf{b}^s + p^s \nabla f^s. \quad (200)$$

### A.4.2 Water phase

The material time derivative for the water phase is

$$\mathbf{a}^w = \frac{D^w \mathbf{v}^w}{Dt} = \frac{\partial \mathbf{v}^w}{\partial t} + \nabla \mathbf{v}^w \cdot \mathbf{v}^w. \quad (201)$$

If the solid phase acts as a reference, we have

$$\mathbf{a}^w = \frac{D^s \mathbf{v}^w}{Dt} - \nabla \mathbf{v}^w \cdot (\mathbf{v}^s - \mathbf{v}^w). \quad (202)$$

The momentum equation for the water phase is

$$\langle \rho^w \rangle \mathbf{a}^w = \nabla \cdot \boldsymbol{\sigma}^w + \langle \rho^w \rangle \mathbf{b}^w + p^w \nabla f^w - f^w \boldsymbol{\mu}^w \cdot (\mathbf{v}^w - \mathbf{v}^s). \quad (203)$$

### A.4.3 Air phase

The material time derivative for the air phase is

$$\mathbf{a}^a = \frac{D^w \mathbf{v}^a}{Dt} = \frac{\partial \mathbf{v}^a}{\partial t} + \nabla \mathbf{v}^a \cdot \mathbf{v}^a. \quad (204)$$

If the solid phase acts as a reference, we have

$$\mathbf{a}^a = \frac{D^s \mathbf{v}^a}{Dt} - \nabla \mathbf{v}^a \cdot (\mathbf{v}^s - \mathbf{v}^a). \quad (205)$$

The momentum equation for the air phase is

$$\langle \rho^a \rangle \mathbf{a}^a = \nabla \cdot \boldsymbol{\sigma}^a + \langle \rho^a \rangle \mathbf{b}^a + \mathbf{t}^a + p^a \nabla f^a - f^a \boldsymbol{\mu}^a \cdot (\mathbf{v}^a - \mathbf{v}^s). \quad (206)$$

#### A.4.4 Mixture

Adding the momentum equations of the three phases, and assuming that all three phases experience identical body forces, gives

$$\langle \rho^s \rangle \mathbf{a}^s + \langle \rho^w \rangle \mathbf{a}^w + \langle \rho^a \rangle \mathbf{a}^a = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b}^s \quad (207)$$

where we have used the relation

$$\rho = \sum_{\alpha} \langle \rho^{\alpha} \rangle. \quad (208)$$

#### A.5 Energy balance

The energy balance of the mixture is

$$\rho \frac{D^s e}{Dt} - \boldsymbol{\sigma}^s : \mathbf{d}^s - \boldsymbol{\sigma}^w : \mathbf{d}^w - \boldsymbol{\sigma}^a : \mathbf{d}^a + \nabla \cdot \mathbf{q} - \rho h = 0. \quad (209)$$

If the fluid stresses are isotropic, i.e.,

$$\boldsymbol{\sigma}^w = -\langle p^w \rangle \mathbf{I} = -f^w p^w \mathbf{I}, \quad \boldsymbol{\sigma}^a = -\langle p^a \rangle \mathbf{I} = -f^a p^a \mathbf{I} \quad (210)$$

we have

$$\rho \frac{D^s e}{Dt} - \boldsymbol{\sigma}^s : \mathbf{d}^s + f^w p^w \nabla \cdot \mathbf{v}^w + f^a p^a \nabla \cdot \mathbf{v}^a + \nabla \cdot \mathbf{q} - \rho h = 0. \quad (211)$$

Recall that the total stress is

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^s + \boldsymbol{\sigma}^w + \boldsymbol{\sigma}^a = \boldsymbol{\sigma}^s - f^w p^w \mathbf{I} - f^a p^a \mathbf{I}. \quad (212)$$

Therefore we can write the mixture balance equation as

$$\rho \frac{D^s e}{Dt} - \boldsymbol{\sigma} : \mathbf{d}^s - f^w p^w \nabla \cdot \mathbf{v}^{sw} - f^a p^a \nabla \cdot \mathbf{v}^{sa} + \nabla \cdot \mathbf{q} - \rho h = 0. \quad (213)$$

After multiplying by  $p^w$  and  $p^a$ , respectively, we can write (188) and (195) as

$$\begin{aligned} p^w (1 - f^s) \frac{D^s S_w}{Dt} + \frac{p^w f^w}{K_w(p^w)} \frac{D^s p^w}{Dt} + B p^w S_w \nabla \cdot \mathbf{v}^s - \\ \frac{p^w f^w}{K_w(p^w)} \nabla p^w \cdot \mathbf{v}^{sw} + p^w \nabla f^w \cdot \mathbf{v}^{sw} = f^w p^w \nabla \cdot \mathbf{v}^{sw} \end{aligned} \quad (214)$$

and

$$\begin{aligned} -p^a (1 - f^s) \frac{D^s S_w}{Dt} + \frac{p^a f^a}{K_a(p^a)} \frac{D^s p^a}{Dt} + B p^a S_a \nabla \cdot \mathbf{v}^s - \\ \frac{p^a f^a}{K_a(p^a)} \nabla p^a \cdot \mathbf{v}^{sa} + p^a \nabla f^a \cdot \mathbf{v}^{sa} = f^a p^a \nabla \cdot \mathbf{v}^{sa}. \end{aligned} \quad (215)$$

Adding these equations and substitution into the mixture energy balance equation, and noting that

$$\nabla \cdot \mathbf{v}^s = \mathbf{I} : \mathbf{d}^s \quad (216)$$

gives

$$\begin{aligned} \rho \frac{D^s e}{Dt} - \boldsymbol{\sigma} : \mathbf{d}^s - B \left[ \sum_{\alpha=\{w,a\}} p^{\alpha} S_{\alpha} \right] \mathbf{I} : \mathbf{d}^s - (p^w - p^a) (1 - f^s) \frac{D^s S_w}{Dt} - \sum_{\alpha=\{w,a\}} \frac{p^{\alpha} f^{\alpha}}{K^{\alpha}(p^{\alpha})} \frac{D^s p^{\alpha}}{Dt} \\ + \sum_{\alpha=\{w,a\}} \left[ \frac{p^{\alpha} f^{\alpha}}{K^{\alpha}(p^{\alpha})} \nabla p^{\alpha} \cdot \mathbf{v}^{s\alpha} - p^{\alpha} \nabla f^{\alpha} \cdot \mathbf{v}^{s\alpha} \right] + \nabla \cdot \mathbf{q} - \rho h = 0. \end{aligned} \quad (217)$$

We can define the effective stress to be

$$\boldsymbol{\sigma}_{\text{eff}} := \boldsymbol{\sigma} + B \left[ \sum_{\alpha=\{w,a\}} p^\alpha S_\alpha \right] \mathbf{I} \quad (218)$$

to get the energy balance equation

$$\begin{aligned} \rho \frac{D^s e}{Dt} - \boldsymbol{\sigma}_{\text{eff}} : \mathbf{d}^s - (p^w - p^a)(1 - f^s) \frac{D^s S_w}{Dt} - \sum_{\alpha=\{w,a\}} \frac{p^\alpha f^\alpha}{K^\alpha(p^\alpha)} \frac{D^s p^\alpha}{Dt} \\ + \sum_{\alpha=\{w,a\}} \left[ \frac{p^\alpha f^\alpha}{K^\alpha(p^\alpha)} \nabla p^\alpha - p^\alpha \nabla f^\alpha \right] \cdot \mathbf{v}^{s\alpha} + \nabla \cdot \mathbf{q} - \rho h = 0. \end{aligned} \quad (219)$$

Thus we see that the effective stress is energy conjugate to the rate of deformation of the solid. We also find that the capillary stress is energy conjugate to the saturation.

## A.6 Simplified model

Assume that the water and the air phases do not move relative to the solid phase, i.e.,

$$\mathbf{v}^s = \mathbf{v}^w = \mathbf{v}^a \quad \text{and} \quad \mathbf{a}^s = \mathbf{a}^w = \mathbf{a}^a. \quad (220)$$

Then the mass balance equations of the three phases reduce to

$$\begin{aligned} \frac{D^s f^s}{Dt} + (B - 1 + f^s) \nabla \cdot \mathbf{v}^s &= 0 \\ (1 - f^s) \frac{D^s S_w}{Dt} + \frac{f^w}{K_w(p^w)} \frac{D^s p^w}{Dt} + B S_w \nabla \cdot \mathbf{v}^s &= 0 \\ - (1 - f^s) \frac{D^s S_w}{Dt} + \frac{f^a}{K_a(p^a)} \frac{D^s p^a}{Dt} + B(1 - S_w) \nabla \cdot \mathbf{v}^s &= 0. \end{aligned} \quad (221)$$

The momentum balance equation for the mixture is

$$\rho \mathbf{a}^s = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b}^s \quad \text{where} \quad \rho = \sum_{\alpha} \langle \rho^\alpha \rangle. \quad (222)$$

The energy balance equation is

$$\rho \frac{D^s e}{Dt} - \boldsymbol{\sigma}_{\text{eff}} : \mathbf{d}^s - (p^w - p^a)(1 - f^s) \frac{D^s S_w}{Dt} - \sum_{\alpha=\{w,a\}} \frac{p^\alpha f^\alpha}{K^\alpha(p^\alpha)} \frac{D^s p^\alpha}{Dt} + \nabla \cdot \mathbf{q} - \rho h = 0 \quad (223)$$

where

$$\boldsymbol{\sigma}_{\text{eff}} = \boldsymbol{\sigma} + B[p^w S_w + p^a S_a] \mathbf{I} = \boldsymbol{\sigma} + B[(p^w - p^a) S_w + p^a] \mathbf{I} \quad (224)$$

and

$$B := 1 - \frac{\tilde{K}^s(\tilde{p}^s)}{K_s(p^s)}. \quad (225)$$

If we make the further assumption that at high strain-rates the water and air pressures are equal,  $p^w = p^a$ , the mass balance equations can be written as (with the substitution  $f^s = 1 - \phi$ , where  $\phi$  is the porosity)

$$\begin{aligned} \frac{D^s \phi}{Dt} - (B - \phi) \nabla \cdot \mathbf{v}^s &= 0 \\ \left[ \frac{S_w \phi}{K_w(p^w)} + \frac{(1 - S_w) \phi}{K_a(p^w)} \right] \frac{D^s p^w}{Dt} + B \nabla \cdot \mathbf{v}^s &= 0. \end{aligned} \quad (226)$$

The momentum balance equation retains the same form, while the energy balance equation becomes

$$\rho \frac{D^s e}{Dt} - \sigma_{\text{eff}} : \mathbf{d}^s - \bar{p}^w \left[ \frac{S_w \phi}{K_w(\bar{p}^w)} + \frac{(1 - S_w) \phi}{K_a(\bar{p}^w)} \right] \frac{D^s \bar{p}^w}{Dt} + \nabla \cdot \mathbf{q} - \rho h = 0 \quad (227)$$

where

$$\sigma_{\text{eff}} = \sigma + B \bar{p}^w \mathbf{I}. \quad (228)$$

## B Partially saturated bulk modulus

In previous reports we have chosen to compute the tangent bulk modulus of the partially saturated soil using a variant of the Biot-Gassman model for fully saturated rocks that is valid for long wavelength displacements (Berryman and Milton, 1991; Berryman, 2006; Dvorkin et al., 1999). We add a few more details in this appendix.

The Biot-Gassman model is based on the Biot equations of poroelasticity and assumes small strains and long wavelength disturbances relative to the pore size. In this model, the average stress tensor ( $\sigma$ ) of the saturated porous medium is given by

$$\sigma = [(H - 2G)\text{tr}(\epsilon) - C\zeta] \mathbf{I} + 2G\epsilon \quad (229)$$

and the fluid pressure ( $\bar{p}^w$ ) is

$$\bar{p}^w = M\zeta - C\text{tr}(\epsilon) \quad (230)$$

where  $G$  is the shear modulus,  $\epsilon$  is the strain in the mixture,  $C$  is the Biot cross coefficient, and

$$C = \frac{B}{\frac{B}{K_s} + \phi \left( \frac{1}{K_w} - \frac{1}{K_s} \right)}, \quad H = K_d + BC + \frac{4}{3}G, \quad \zeta = \phi[\text{tr}(\epsilon) - \epsilon_v^w], \quad M = \frac{C}{B}, \quad B := 1 - \frac{K_d}{K_s}. \quad (231)$$

In the above  $K_d$  is the bulk modulus of the drained porous frame,  $K_s$  is the bulk modulus of the solid grains,  $\phi$  is the porosity, and  $\epsilon_v^w$  is the volumetric strain in the fluid.

Using these, the fully saturated bulk modulus for a soil (with a single material in the solid matrix) is

$$\widehat{K} = K_d + B^2 D \quad (232)$$

where

$$\frac{1}{D} = \frac{B(1 - B)}{K_d} + \phi \left( \frac{1}{K_w} - \frac{1}{K_s} \right). \quad (233)$$

At partial saturation, we compute the pore fluid bulk modulus using a harmonic mean (lower bound) on the air and water bulk moduli ( $K_a, K_w$ ):

$$\frac{1}{K_f(\bar{p}^w)} = \frac{S_w}{K_w(\bar{p}^w)} + \frac{1 - S_w}{K_a(\bar{p}^w)} \quad (234)$$

and get the expression for the bulk modulus that is use in the partially saturated Arena model

$$\widehat{K}(\bar{p}_{\text{eff}}, \bar{p}^w, \bar{\epsilon}_v^p, \phi, S_w) = K_d(\bar{p}_{\text{eff}}) + \frac{B^2(\bar{p}_{\text{eff}})}{\frac{B(\bar{p}_{\text{eff}})}{K_s(\bar{p}_{\text{eff}})} + \phi \left( \frac{1}{K_f(\bar{p}^w)} - \frac{1}{K_s(\bar{p}_{\text{eff}})} \right)} \quad (235)$$

where  $\widehat{K}$  is the effective bulk modulus of the partially saturated soil,  $K_d$  is the bulk modulus of the drained soil,  $K_f$  is the bulk modulus of the pore fluid, and  $K_s$  is the bulk modulus of the solid grains. For the situation where  $B \approx 1$ ,

$$\widehat{K}(\bar{p}_{\text{eff}}, \bar{p}^w, \bar{\epsilon}_v^p, \phi, S_w) = K_d(\bar{p}_{\text{eff}}) + \frac{K_f(\bar{p}^w)}{\phi}. \quad (236)$$

## C Pore pressure equation

From Borja's mass balance equations we have

$$\begin{aligned} \frac{D\phi}{Dt} - (B - \phi) \nabla \cdot \mathbf{v}^s &= 0 \\ \left[ \frac{S_w \phi}{K_w(p^w)} + \frac{(1 - S_w) \phi}{K_a(p^w)} \right] \frac{D\bar{p}^w}{Dt} + B \nabla \cdot \mathbf{v}^s &= 0. \end{aligned} \quad (237)$$

Upon combining the two equations we get

$$\left[ \frac{S_w \phi}{K_w(p^w)} + \frac{(1 - S_w) \phi}{K_a(p^w)} \right] \frac{D\bar{p}^w}{Dt} + \left( \frac{B}{B - \phi} \right) \frac{D\phi}{Dt} = 0. \quad (238)$$

From equation (52),

$$\phi = (1 - S_0) \phi_0 \exp(\varepsilon_v^a - \varepsilon_v) + S_0 \phi_0 \exp(\varepsilon_v^w - \varepsilon_v) \quad (239)$$

Therefore, the material derivative of the porosity is

$$\frac{D\phi}{Dt} = (1 - S_0) \phi_0 \exp(\varepsilon_v^a - \varepsilon_v) \left( \frac{D\varepsilon_v^a}{Dt} - \frac{D\varepsilon_v}{Dt} \right) + S_0 \phi_0 \exp(\varepsilon_v^w - \varepsilon_v) \left( \frac{D\varepsilon_v^w}{Dt} - \frac{D\varepsilon_v}{Dt} \right) \quad (240)$$

Plugging (240) into (238) gives

$$\begin{aligned} \left[ \frac{S_w \phi}{K_w(p^w)} + \frac{(1 - S_w) \phi}{K_a(p^w)} \right] \frac{D\bar{p}^w}{Dt} + \left( \frac{B \phi_0}{B - \phi} \right) \left[ (1 - S_0) \exp(\varepsilon_v^a - \varepsilon_v) \left( \frac{D\varepsilon_v^a}{Dt} - \frac{D\varepsilon_v}{Dt} \right) + \right. \\ \left. S_0 \exp(\varepsilon_v^w - \varepsilon_v) \left( \frac{D\varepsilon_v^w}{Dt} - \frac{D\varepsilon_v}{Dt} \right) \right] = 0. \end{aligned} \quad (241)$$

For equilibrated fluid pressures,  $\bar{p}^a = \bar{p}^w$ , and we can write the above using quantities that are positive in compression as

$$\begin{aligned} \left[ \frac{S_w \phi}{K_w} + \frac{(1 - S_w) \phi}{K_a} \right] \frac{D\bar{p}^w}{Dt} - \left( \frac{B \phi_0}{B - \phi} \right) \left[ (1 - S_0) \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) \frac{\partial \bar{\varepsilon}_v^a}{\partial \bar{p}^w} + S_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \frac{\partial \bar{\varepsilon}_v^w}{\partial \bar{p}^w} \right] \frac{D\bar{p}^w}{Dt} \\ = - \left( \frac{B \phi_0}{B - \phi} \right) \left[ (1 - S_0) \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) + S_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \right] \frac{D\bar{\varepsilon}_v}{Dt} \end{aligned} \quad (242)$$

Noting that

$$\frac{\partial \bar{\varepsilon}_v^a}{\partial \bar{p}^w} = \frac{1}{K_a} \quad \text{and} \quad \frac{\partial \bar{\varepsilon}_v^w}{\partial \bar{p}^w} = \frac{1}{K_w} \quad (243)$$

we have

$$\begin{aligned} \left[ \frac{S_w \phi}{K_w} + \frac{(1 - S_w) \phi}{K_a} \right] \frac{D\bar{p}^w}{Dt} - \left( \frac{B \phi_0}{B - \phi} \right) \left[ \frac{1 - S_0}{K_a} \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) + \frac{S_0}{K_w} \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \right] \frac{D\bar{p}^w}{Dt} \\ = - \left( \frac{B \phi_0}{B - \phi} \right) \left[ (1 - S_0) \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) + S_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \right] \frac{D\bar{\varepsilon}_v}{Dt} \end{aligned} \quad (244)$$

or

$$\begin{aligned} \frac{D\bar{\varepsilon}_v}{Dt} = \frac{1}{(1 - S_0) \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) + S_0 \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w)} \left[ - \frac{(B - \phi) \phi}{B \phi_0} \left( \frac{S_w}{K_w} + \frac{1 - S_w}{K_a} \right) + \right. \\ \left. \frac{1 - S_0}{K_a} \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) + \frac{S_0}{K_w} \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \right] \frac{D\bar{p}^w}{Dt} \end{aligned} \quad (245)$$

We can convert this equation into the form

$$\frac{d\bar{p}^w}{d\bar{\varepsilon}_v} = \frac{1}{B} \quad (246)$$

where

$$B := \frac{1}{(1 - S_0)\exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) + S_0\exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w)} \left[ -\frac{(B - \phi)\phi}{B\phi_0} \left( \frac{S_w}{K_w} + \frac{1 - S_w}{K_a} \right) + \frac{1 - S_0}{K_a}\exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^a) + \frac{S_0}{K_w}\exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \right]. \quad (247)$$

We can then find the total pore pressure using

$$\bar{p}^w = \int \frac{d\bar{p}^w}{d\bar{\varepsilon}_v} d\bar{\varepsilon}_v. \quad (248)$$

For a fully saturated medium,  $S_0 = 1$ , we have

$$B = \frac{1}{K_w\exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w)} \left[ -\frac{(B - \phi)\phi}{B\phi_0} + \exp(\bar{\varepsilon}_v - \bar{\varepsilon}_v^w) \right]. \quad (249)$$

Note that a medium with zero initial porosity,  $\phi_0 = 0$ , cannot be modeled using our approach.

## D Computing saturation from weight ratio

The experimental SHPB data provided by UT Dallas includes the dry sand density and the water content in the form of the ratio by weight of the water to the sand/soil. We have to convert that information into porosity, saturation, and mass density.

To do that we assume that the soil skeleton material has the density of quartz such that the mass ( $m$ ) of the dry sand can be written as

$$m_{\text{drysand}} = m_{\text{air}} + m_{\text{quartz}}. \quad (250)$$

Expressed in terms of volume ( $v$ ) and density ( $\rho$ ),

$$\rho_{\text{drysand}} v_{\text{drysand}} = \rho_{\text{air}} v_{\text{air}} + \rho_{\text{quartz}} v_{\text{quartz}}. \quad (251)$$

If we neglect the density of air, we have

$$\rho_{\text{drysand}} v_{\text{drysand}} \approx \rho_{\text{quartz}} v_{\text{quartz}} \implies \frac{\rho_{\text{drysand}}}{\rho_{\text{quartz}}} = \frac{v_{\text{quartz}}}{v_{\text{drysand}}}. \quad (252)$$

The porosity ( $\phi$ ) is defined as the ratio of the void volume to the total volume of the sand, i.e.,

$$\phi = \frac{v_{\text{air}}}{v_{\text{drysand}}} = 1 - \frac{v_{\text{quartz}}}{v_{\text{drysand}}} = 1 - \frac{\rho_{\text{drysand}}}{\rho_{\text{quartz}}}. \quad (253)$$

For partially saturated sand, the SHPB tests provide the ratio

$$\alpha = \frac{m_{\text{water}}}{m_{\text{drysand}}} = \frac{\rho_{\text{water}} v_{\text{water}}}{\rho_{\text{drysand}} v_{\text{drysand}}}. \quad (254)$$

The saturation is defined as

$$S_w = \frac{v_{\text{water}}}{v_{\text{air}} + v_{\text{water}}} \quad (255)$$

and the porosity of the wet soil is

$$\phi = \frac{v_{\text{air}} + v_{\text{water}}}{v_{\text{wetsand}}}. \quad (256)$$

Therefore, if the volume of the sand does not change due to the introduction of water,

$$S_w = \frac{1}{\phi} \frac{v_{\text{water}}}{v_{\text{wetsand}}} = \frac{1}{\phi} \frac{v_{\text{water}}}{v_{\text{drysand}}} . \quad (257)$$

Plugging the above relation into the expression for  $\alpha$  gives us

$$S_w = \frac{\alpha \rho_{\text{drysand}}}{\phi \rho_{\text{water}}} . \quad (258)$$

The mass of the wet sand is

$$m_{\text{wetsand}} = m_{\text{drysand}} + m_{\text{water}} \implies \rho_{\text{wetsand}} v_{\text{wetsand}} = \rho_{\text{drysand}} v_{\text{drysand}} + \rho_{\text{water}} v_{\text{water}} . \quad (259)$$

Since  $v_{\text{wetsand}} = v_{\text{drysand}}$ , we have

$$\rho_{\text{wetsand}} = \rho_{\text{drysand}} + \rho_{\text{water}} \frac{v_{\text{water}}}{v_{\text{drysand}}} = \rho_{\text{drysand}} + \phi S_w \rho_{\text{water}} . \quad (260)$$

## E The Kayenta damage model

The Kayenta damage model uses a scalar parameter  $D \in [0, 1]$  to collapse the limit surface as damage increases. The parameters of the limit surface that are modified are FSLOPE, PEAKI1, STREN, and YSLOPE. Let these four parameters be

$$I_1^{\text{peak}} := \text{PEAKI1} , \quad f^{\text{slope}} := \text{FSLOPE} , \quad \sqrt{J_2^{\text{coh}}} := \text{STREN} , \quad y^{\text{slope}} := \text{YSLOPE} . \quad (261)$$

Let the input undamaged values of these parameters be

$$I_1^{\text{peak,intact}} , \quad f^{\text{slope,intact}} , \quad \sqrt{J_2^{\text{coh,intact}}} , \quad y^{\text{slope,intact}} . \quad (262)$$

Also, let the input fully damaged values of the four parameters be

$$I_1^{\text{peak,failed}} , \quad f^{\text{slope,failed}} , \quad \sqrt{J_2^{\text{coh,failed}}} , \quad y^{\text{slope,failed}} . \quad (263)$$

Then the current values of the four parameters are computed using linear interpolation between the intact and failed values:

$$\begin{aligned} I_1^{\text{peak}} &= (1 - D) I_1^{\text{peak,intact}} + D I_1^{\text{peak,failed}} \\ f^{\text{slope}} &= (1 - D) f^{\text{slope,intact}} + D f^{\text{slope,failed}} \\ \sqrt{J_2^{\text{coh}}} &= (1 - D) \sqrt{J_2^{\text{coh,intact}}} + D \sqrt{J_2^{\text{coh,failed}}} \\ y^{\text{slope}} &= (1 - D) y^{\text{slope,intact}} + D y^{\text{slope,failed}} . \end{aligned} \quad (264)$$

Also, to incorporate variability and size effects during the failure process, the parameters are scaled by the material point volume using a relation of the form

$$\begin{aligned} I_1^{\text{peak}} &= I_1^{\text{peak,intact}} \left[ \frac{v^{\text{expt}} \ln R}{v^{\text{elem}} \ln(1/2)} \right]^{1/m} , \quad f^{\text{slope}} = f^{\text{slope,intact}} \left[ \frac{v^{\text{expt}} \ln R}{v^{\text{elem}} \ln(1/2)} \right]^{1/m} \\ \sqrt{J_2^{\text{coh}}} &= \sqrt{J_2^{\text{coh,intact}}} \left[ \frac{v^{\text{expt}} \ln R}{v^{\text{elem}} \ln(1/2)} \right]^{1/m} , \quad y^{\text{slope}} = y^{\text{slope,intact}} \left[ \frac{v^{\text{expt}} \ln R}{v^{\text{elem}} \ln(1/2)} \right]^{1/m} \end{aligned} \quad (265)$$

where  $\{I_1^{\text{peak}}, f^{\text{slope}}, \sqrt{J_2^{\text{coh}}}, y^{\text{slope}}\}^{\text{intact}}$  are mean values from experiments performed on samples that have volume  $v^{\text{expt}}$ ,  $v^{\text{elem}}$  is the volume of the material point,  $R \sim \mathcal{U}[0, 1]$  is a uniformly distributed random number, and  $m$  is a Weibull modulus.



At this stage we need a model for the evolution of  $D$ . Kayenta uses a time to failure model that depends on two new input parameters: the time to failure ( $t^{\text{fail}}$ ) and the failure speed ( $f^{\text{speed}}$ ), and defines a “coherence” parameter ( $C$ ) as

$$C := 1 - D = \frac{\exp \left[ -f^{\text{speed}} \left( 1 - \frac{t^{\text{grow}}}{t^{\text{fail}}} \right) \right]}{1 + \exp \left[ -f^{\text{speed}} \left( 1 - \frac{t^{\text{grow}}}{t^{\text{fail}}} \right) \right]} \quad (266)$$

where  $t^{\text{grow}}$  is state variable that increases only when the stress state is on the limit surface. The value of  $C$  is never allowed to increase.

## F Implementing parameter variability

The probability density function of random variable  $x$  that is Weibull distributed can be expressed as

$$\text{We}(x) = \beta \alpha x^{\alpha-1} \exp(-\beta x^\alpha) \quad \text{for } x \geq 0. \quad (267)$$

The expression used in the C++11 standard implementation is

$$\text{We}(x) = \frac{a}{b} \left( \frac{x}{b} \right)^{a-1} \exp \left[ -\left( \frac{x}{b} \right)^a \right]. \quad (268)$$

The relationship between these two expressions is

$$\alpha \equiv a \quad \text{and} \quad \beta \equiv \frac{1}{b^a} \implies b = \beta^{-1/\alpha}. \quad (269)$$

The shape parameter is  $a = \alpha > 0$  and the scale parameter is  $b > 0$ . The shape parameter  $a$  is also called the **Weibull modulus** in the content of material strength distribution.

The mean of the distribution is

$$\mathbb{E}(x) = b \Gamma \left( 1 + \frac{1}{a} \right) = \beta^{-1/\alpha} \Gamma \left( 1 + \frac{1}{\alpha} \right) \quad (270)$$

where  $\Gamma$  is the gamma function. If we assume that the expected value is better represented by the median, we have

$$\mathbb{E}(x) = b [\ln(2)]^{1/a}. \quad (271)$$

To generate the Weibull distribution for a random variable, we typically use a transformation from a uniformly distributed random variable. To find the transformation between two probability distributions  $f(y)$  and  $g(x)$ , we use the fundamental relation

$$f(y) = g(x) \left| \frac{dx}{dy} \right| \quad (272)$$

where the absolute value of the Jacobian of the transformation is used to make sure that probabilities sum to 1. For the special case where the distribution  $g(x)$ ,  $x \in \mathcal{U} \sim [0, 1]$  is uniform, we have

$$f(y) = \left| \frac{dx}{dy} \right|. \quad (273)$$

Therefore,

$$x = \int_0^y f(z) dz. \quad (274)$$

For the Weibull distribution, the right hand side is the cumulative distribution function,

$$x = \int_0^y \text{We}(z) dz = \int_0^y \beta \alpha z^{\alpha-1} \exp(-\beta z^\alpha) dz = 1 - \exp(-\beta y^\alpha) = 1 - \exp \left[ -\left( \frac{y}{b} \right)^a \right]. \quad (275)$$

This relation can be inverted to give the transformed uniformly distributed random number between 0 and 1:

$$y = \left[ -\frac{1}{\beta} \ln(1-x) \right]^{1/\alpha} = b [-\ln(1-x)]^{1/a} . \quad (276)$$

For a random variable that has the mean  $\mathbb{E}(y) \approx \bar{y}$ , from (270), the scale parameter is

$$b = \frac{\mathbb{E}(x)}{\Gamma(1 + \frac{1}{a})} \approx \frac{\bar{y}}{\Gamma(1 + \frac{1}{a})} . \quad (277)$$

Therefore, the Weibull-transformed uniformly distributed random variable can be written as

$$y = \frac{\bar{y}}{\Gamma(1 + \frac{1}{a})} [-\ln(1-x)]^{1/a} . \quad (278)$$

At this stage one typically invokes the fact that if  $x$  is uniformly distributed then so is  $1-x$  and we can simplify the computation by using

$$y = \frac{\bar{y}}{\Gamma(1 + \frac{1}{a})} [-\ln(x)]^{1/a} . \quad (279)$$

Alternatively, we can assume that the sample median is a better approximation of the expected value and use equation (271) to compute the scale parameter:

$$b = \frac{\bar{y}}{[\ln(2)]^{1/a}} . \quad (280)$$

In that case we have

$$y = \bar{y} \left[ \frac{\ln(x)}{\ln(2)} \right]^{1/a} = \bar{y} \left[ \frac{\ln(x)}{\ln(1/2)} \right]^{1/a} . \quad (281)$$

The existing implementation of the Weibull generator in Uintah uses the following approach. A uniformly distributed random number  $x$  is generated. This number is used to compute the quantity

$$F = [-\ln(x)]^{1/a} \quad (282)$$

where  $a$  is the Weibull modulus. Two other quantities are computed:

$$C = \left[ \frac{v_{\text{expt}}}{v_{\text{elem}}} \right]^{1/m} \quad \text{and} \quad \eta = \frac{\bar{y}}{\Gamma(1 + \frac{1}{a})} \quad (283)$$

where  $v_{\text{expt}}$  is a reference volume,  $v_{\text{elem}}$  is the particle volume,  $m$  is an exponent, and  $\bar{y}$  is the mean value of the parameter ( $y$ ) that is Weibull distributed. The value of  $y$  is computed using the product of  $F$ ,  $C$ , and  $\eta$ , giving

$$y = \left[ \frac{v_{\text{expt}}}{v_{\text{elem}}} \right]^{1/m} \frac{\bar{y}}{\Gamma(1 + \frac{1}{a})} [-\ln(x)]^{1/a} \quad (284)$$

The code typically uses  $m = a$  to get

$$y = \frac{\bar{y}}{\Gamma\left(1 + \frac{1}{a}\right)} \left[ -\frac{v_{\text{expt}}}{v_{\text{elem}}} \ln(x) \right]^{1/a}. \quad (285)$$

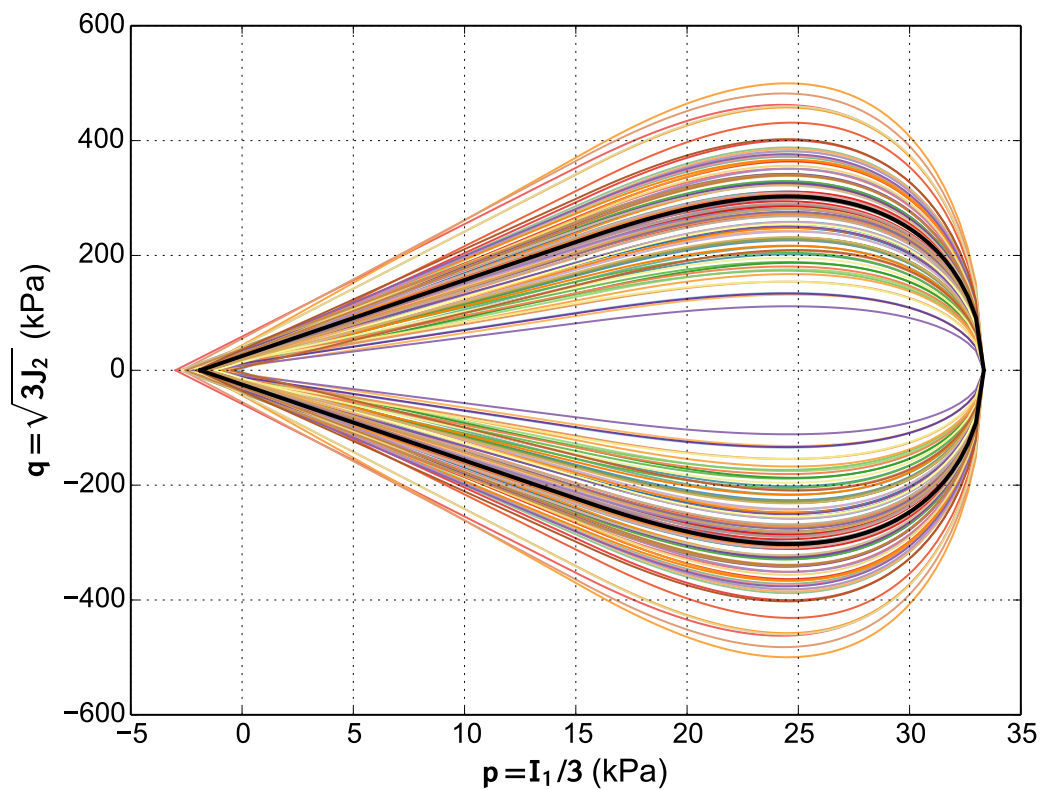
This expression is identical to equation (279) except for a size-effect factor. Note that (281) is the form used in Scott Swan's thesis (previously implemented in Uintah):

$$y = \left[ \frac{v_{\text{expt}}}{v_{\text{elem}}} \right]^{1/a} \bar{y} \left[ \frac{\ln x}{\ln(1/2)} \right]^{1/a}. \quad (286)$$

For our purposes, if we use the C++11 Weibull distribution generator, we can incorporate the volume scaling by just multiplying the scaling factor to the number generated, i.e.,

$$y = \left[ \frac{v_{\text{expt}}}{v_{\text{elem}}} \right]^{1/m} \text{We}(\bar{y}, a, b, R) \quad (287)$$

where  $R$  is the uniformly distributed pseudorandom number in  $[0, 1]$  generated by the Mersenne twister algorithm. The variability produced in the yield surface is shown in Figure 1.



**Figure 1** – Variability in the yield surface by applying volume scaling and Weibull variability in the yield surface parameters. The median yield surface is shown by the black line. The median parameter values are  $PEAKI1 = 1$  kPa,  $FSL OPE = 0.453$ ,  $YSLOPE = 0.31$ ,  $STREN = 10$  MPa. The element volume is  $10^{-6} \text{ m}^3$  while the reference volume is  $10^{-3} \text{ m}^3$ . The Weibull modulus for all the parameters is 4.

## G Density-dependence model

We use the Pabst-Gregorova model (Pabst and Gregorová, 2015) to scale the bulk and shear moduli. For the hydrostatic strength we use an ad-hoc model that is calibrated based on the high density dry Mason sand SHPB tests.

Let  $\phi_0$  be the initial porosity of the test material and let  $\phi_{\text{ref}}$  be the initial porosity of the reference material that was used to calibrate the bulk modulus parameters and the crush curve. Following Pabst and Gregorova, we define a modulus scaling factor  $K_{\text{fac}}$

$$K_{\text{fac}} = \exp \left[ -\frac{\phi_0}{1 - \phi_0} + \frac{\phi_{\text{ref}}}{1 - \phi_{\text{ref}}} \right] \quad (288)$$

such that the bulk and shear moduli of the test material is

$$K_d \leftarrow K_{\text{fac}} K_d, \quad \text{and} \quad G \leftarrow K_{\text{fac}} G. \quad (289)$$

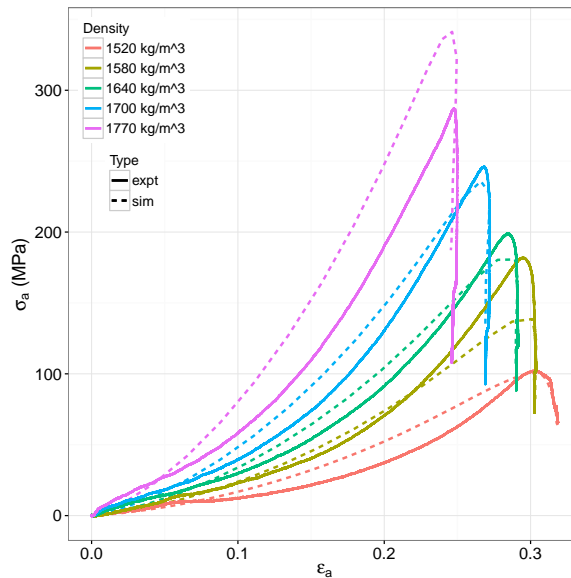
These scaled moduli are used in the partially saturated model.

For the hydrostatic strength we use the ad-hoc density scaling model

$$p_1 \leftarrow p_1 \exp [\rho_{\text{fac}} K_{\text{fac}} (K_{\text{fac}} - 1)] \quad (290)$$

where  $p_1$  is the parameter of the crush curve model, and  $\rho_{\text{fac}}$  is calibrated using high density data and has a value between 1 and 10.

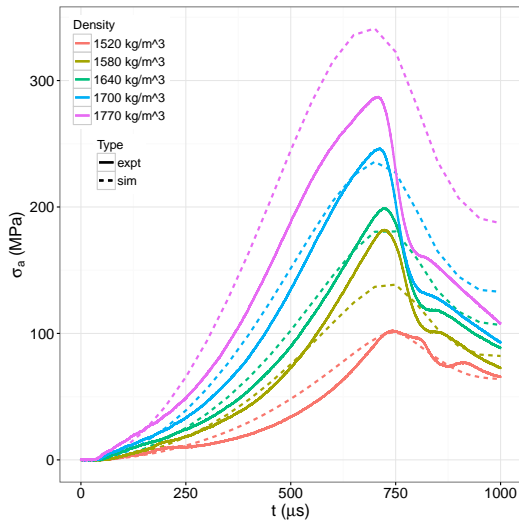
Plots of the predicted and experimental SHPB test results for a range of initial Mason sand densities are shown in the figures below.



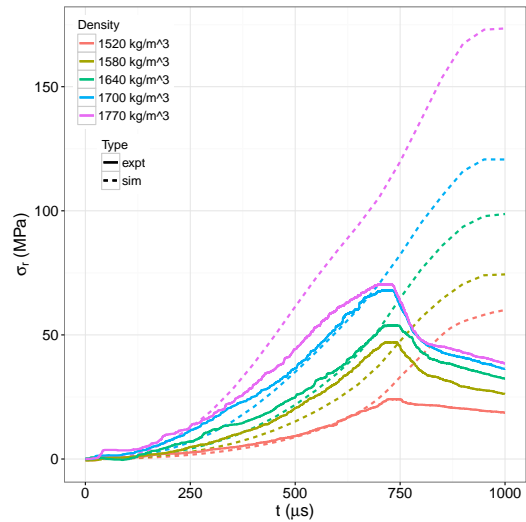
**Figure 2** – Axial stress vs. axial strain in split-Hopkinson pressure bar: dry Mason sand: Various initial densities. We have used  $\phi_{\text{ref}} = 0.42$  and  $\rho_{\text{fac}} = 5$ .

## H Verifying stress paths for uniaxial strain loading

Most of the simulations discussed in this report have been driven by uniaxial strain. We can check that the code is doing the right thing by comparing the slope of the loading path in  $p$ - $q$  space for elastic states. This appendix discusses what we should expect for linearly elastic materials.

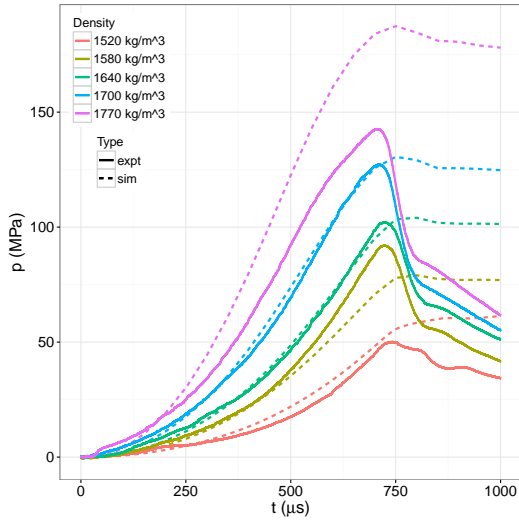


(a) Axial stress vs. time.

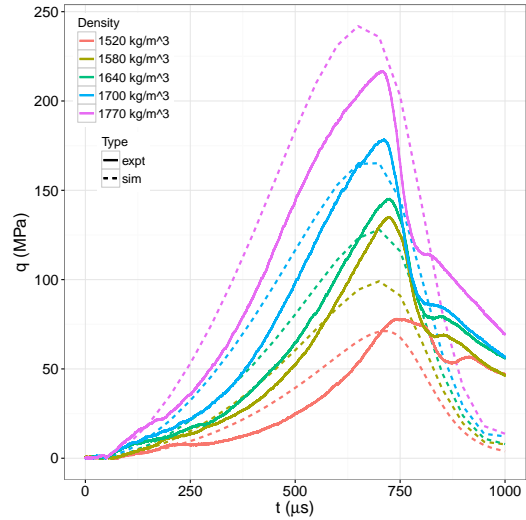


(b) Radial stress vs. time.

**Figure 3** – Axial stress and radial stress in split-Hopkinson pressure bar: dry Mason sand: Various initial densities

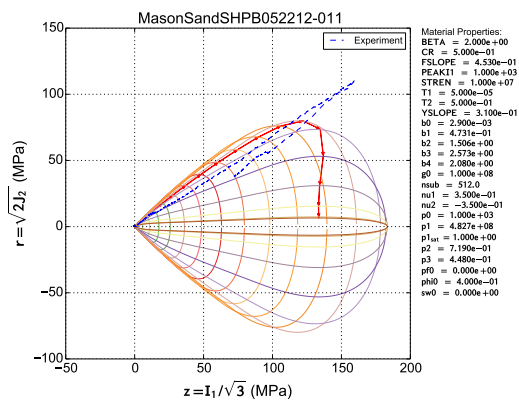


(a)  $p$  vs. time.

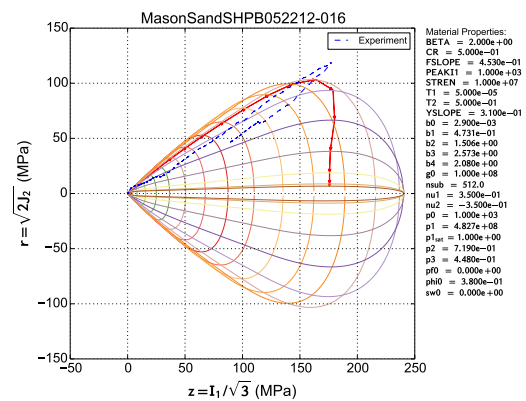


(b)  $q$  vs. time.

**Figure 4** – Axial stress and radial stress in split-Hopkinson pressure bar: dry Mason sand: Various initial densities

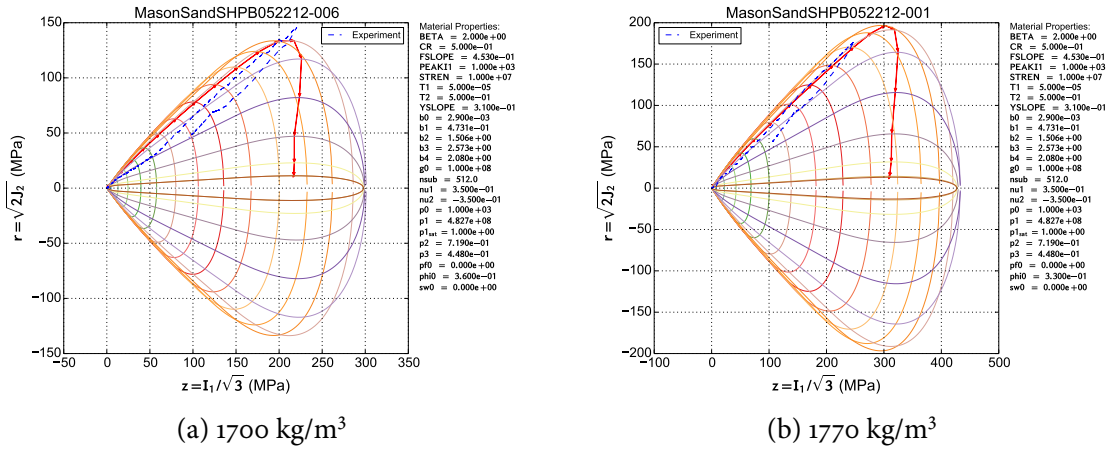


(a) 1580 kg/m<sup>3</sup>



(b) 1640 kg/m<sup>3</sup>

**Figure 5** – Plot of stress path and yield surface in  $z$ - $r$  space. Moderate initial densities



**Figure 6** – Plot of stress path and yield surface in  $z$ - $r$  space. High initial densities

From linear elasticity,

$$\boldsymbol{\sigma} = \lambda \text{tr}(\boldsymbol{\varepsilon}) \mathbf{I} + 2\mu \boldsymbol{\varepsilon} \quad (291)$$

where

$$\lambda := K - \frac{2}{3}G \quad \text{and} \quad \mu := G \implies K = \lambda + \frac{2}{3}\mu \quad (292)$$

and  $K, G$  are the bulk and shear modulus, respectively. Therefore,

$$\begin{aligned} p &:= \frac{1}{3} \text{tr}(\boldsymbol{\sigma}) = \left( \lambda + \frac{2}{3}\mu \right) \text{tr}(\boldsymbol{\varepsilon}) \\ \mathbf{s} &:= \boldsymbol{\sigma} - p\mathbf{I} = \lambda \text{tr}(\boldsymbol{\varepsilon}) \mathbf{I} + 2\mu \boldsymbol{\varepsilon} - \left( \lambda + \frac{2}{3}\mu \right) \text{tr}(\boldsymbol{\varepsilon}) \mathbf{I} = 2\mu \left[ \boldsymbol{\varepsilon} - \frac{1}{3} \text{tr}(\boldsymbol{\varepsilon}) \mathbf{I} \right]. \end{aligned} \quad (293)$$

From the second equation above, we can compute

$$q := \sqrt{3J_2} = \sqrt{\frac{3}{2} \mathbf{s} : \mathbf{s}} = \sqrt{\frac{3}{2} (2\mu)^2 \left[ \boldsymbol{\varepsilon} : \boldsymbol{\varepsilon} - \frac{1}{3} [\text{tr}(\boldsymbol{\varepsilon})]^2 \right]} = 2\mu \sqrt{\frac{3}{2} \left[ \boldsymbol{\varepsilon} : \boldsymbol{\varepsilon} - \frac{1}{3} [\text{tr}(\boldsymbol{\varepsilon})]^2 \right]}. \quad (294)$$

For uniaxial strain in the 1-direction,  $\text{tr}(\boldsymbol{\varepsilon}) = \varepsilon_{11}$  and  $\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon} = \varepsilon_{11}^2$ , and the above equations for  $p$  and  $q$  become

$$p = \left( \lambda + \frac{2}{3}\mu \right) \varepsilon_{11} = K \varepsilon_{11} \quad \text{and} \quad q = 2G \varepsilon_{11}. \quad (295)$$

Therefore, for uniaxial strain linear elastic deformations starting from zero strain, the slope of the loading path is

$$\frac{q}{p} = \frac{2G}{K} = \frac{2}{K} \frac{3K(1-2\nu)}{2(1+\nu)} = \frac{3(1-2\nu)}{1+\nu}. \quad (296)$$

For  $\nu = 0.35$ , the slope of the loading path is  $\frac{2}{3}$  which can be used as a check of the algorithm.