

MOOSE's PorousFlow module

CSIRO, INL

March 30, 2017

Abstract

This document describes the physics captured by MOOSE's PorousFlow module, as well as describing various numerical aspects and tips to ensure good convergence.

Each nontrivial test in the test suite *has its own separate documentation* and these should help users understand more clearly how to build a PorousFlow input file.

Within this document, black text indicates functionality that has been implemented and tested and red text indicates functionality that should be implemented by December 2017.

Contents

1	Physical equations	6
1.1	Notation	6
1.2	Fluid flow	9
1.2.1	Mass density: M	10
1.2.2	Flux: \mathbf{F}	10
1.3	Heat flow	12
1.3.1	Energy density: \mathcal{E}	13
1.3.2	Heat flux: \mathbf{F}^T	13
1.3.3	A simple example	14
1.4	Solid mechanics	15
1.5	Chemical reactions	16
1.5.1	Adsorption and desorption	16
1.5.2	General geochemistry	17
1.6	Example use case: Single-phase single-component fully-saturated fluid	19
2	Boundaries and line-sinks	21
2.1	Boundaries	21
2.2	Line sinks	22

2.2.1	Poly-line sinks as functions of porepressure or temperature	23
2.2.2	Peaceman Boreholes	24
3	Equations of state and other functions	26
3.1	Fluid properties	26
3.1.1	Ideal gas	27
3.1.2	Simplified fluid	27
3.1.3	Water and steam	27
3.1.4	Brine	28
3.1.5	CO ₂	28
3.1.6	Methane	28
3.2	CO ₂ and brine	29
3.3	Capillary pressure relationships	29
3.3.1	Van Genuchten	29
3.3.2	Broadbridge-White	30
3.3.3	Rogers-Stallybrass-Clements	30
3.4	Density	30
3.5	Tortuosity and diffusion coefficients	31
3.6	Effective fluid pressure	31
3.7	Enthalpy and energy-density	31
3.8	Permeability	31
3.8.1	Exponential	32
3.8.2	Kozeny-Carman	32
3.8.3	Permeability with a solid phase	32

3.9	Porosity	32
3.10	Relative permeability relationships	34
3.10.1	Constant	34
3.10.2	Corey	34
3.10.3	Broadbridge-White	34
3.10.4	van Genuchten and cut van Genuchten	34
3.10.5	FLAC	35
3.11	Thermal conductivity	35
3.12	Viscosity	35
4	Implementation details and numerical issues	36
4.1	Independent (nonlinear) variables	36
4.2	Lumping and upwinding	36
4.2.1	Lumping	36
4.2.2	Upwinding	38
4.3	Preconditioners and linear solvers	39
4.4	Setting the global convergence criterion, and scaling the variables	40
4.4.1	Fluids	40
4.4.2	Heat	41
4.4.3	Mechanics	41
4.4.4	Scaling the variables	42
4.5	Fully saturated formulations	42
4.5.1	Non-upwinded flux Kernels for fully-saturated, single-phase simulations .	42
4.5.2	Non-lumped time derivative of fluid mass for fully-saturated, single-phase, single-component simulations	44

A The continuity equation 46

A.1 Eulerian coordinates 46

A.2 Lagrangian coordinates 47

B Benchmark studies to perform 48

B.1 Use case: Two-component, two-phase, nonisothermal 48

B.2 Two-component, two-phase, nonisothermal, forward coupled stress 49

C Evolution of porosity for an isothermal situation 51

Chapter 1

Physical equations

The equations for heat flow, fluid flow, solid mechanics, and chemical reactions are defined here. Notation is introduced throughout, but summarised in Section 1.1. The Lagrangian coordinate system is used since it co-moves with the mesh.

1.1 Notation

Symbol	Units	Physical description
A^κ	kg.kg^{-1}	Mass of absorbed species κ per mass of rock grain material.
\mathcal{A}_β^i	symbolic	Chemical species i in phase β
A_m	$\text{m}^2.\text{L}^{-1}$	specific reactive surface area for mineral m
$\alpha_{\beta,L}$	m	Longitudinal dispersivity of phase β .
$\alpha_{\beta,T}$	m	Transverse dispersivity of phase β .
α_B	dimensionless	Biot coefficient. I believe we can treat this as a constant
α_f	K^{-1}	Volumetric coefficient of thermal expansion of the fluid.
α_T	K^{-1}	Volumetric coefficient of thermal expansion of the drained porous skeleton (ie, the porous rock without fluid, or which a fluid that is free to move in and out of the rock).
\mathbf{b}	m.s^{-2}	External force density acting on the porous solid. This could be gravitational acceleration, or a load-density from a platten.
β	dimensionless	Index representing phase. For example, β might parameterise liquid ($\beta = 0$), gas ($\beta = 1$) and NAPL ($\beta = 2$).
C_R	$\text{J.kg}^{-1}.\text{K}^{-1}$	Specific heat capacity of rock grains.
C_{ijkl}	Pa	Drained compliance tensor of the porous solid (ie, inverse of E_{ijkl}).
C_β^i	moles per litre	concentration of species i

Symbol	Units	Physical description
C_v	$\text{J.kg}^{-1}.\text{K}^{-1}$	Fluid specific heat capacity at constant volume.
C_p	$\text{J.kg}^{-1}.\text{K}^{-1}$	Fluid specific heat capacity at constant pressure.
χ_β^κ	kg.kg^{-1}	Mass fraction of component κ present in phase β .
\mathcal{D}_β^κ	$\text{m}^2.\text{s}^{-1}$	Fluid dispersion tensor for species κ in phase β .
$D_{\beta,L}^\kappa$	$\text{m}^2.\text{s}^{-1}$	Longitudinal dispersion coefficient for species κ in phase β .
$D_{\beta,T}^\kappa$	$\text{m}^2.\text{s}^{-1}$	Transverse dispersion coefficient for species κ in phase β .
d_β^κ	$\text{m}^2.\text{s}^{-1}$	Molecular diffusion coefficient for component κ in phase β .
$\delta_{l\beta}$	dimensionless	Kronecker delta, unity if $\beta = l$, else zero.
\mathcal{E}	J.m^{-3}	Energy density of the rock-fluid system.
\mathcal{E}_β	J.kg^{-1}	Internal energy of fluid phase β . In the simplest case this is just $c_\beta T$, where c_β is the fluid's specific heat capacity.
ϵ_{ij}	dimensionless	Strain tensor of the porous solid $((\nabla_k u_l + \nabla_l u_k)/2)$.
$\epsilon_{ij}^{\text{elastic}}$	dimensionless	Elastic strain tensor of the porous solid. The total strain $\epsilon = \epsilon^{\text{elastic}} + \epsilon^{\text{plastic}}$.
$\epsilon_{ij}^{\text{plastic}}$	dimensionless	Plastic strain tensor of the porous solid. The total strain $\epsilon = \epsilon^{\text{elastic}} + \epsilon^{\text{plastic}}$.
η	dimensionless	exponent in rate expression
E_{ijkl}	Pa	“Drained” elasticity tensor of the porous skeleton (ie, this enters the stress-strain relation when fluid is allowed to freely drain from the skeleton, or when the skeleton is dry).
ϕ	dimensionless	Porosity of the solid.
ϕ_m	dimensionless	volume fraction of mineral in solid
\mathbf{F}	$\text{kg.s}^{-1}.\text{m}^{-2}$	Fluid flux. This is a sum of the advective (Darcy) flux, and a diffusive-and-dispersive flux.
\mathbf{F}^T	$\text{J.s}^{-1}.\text{m}^{-2}$	Heat flux. This is a sum of heat conduction through the rock-fluid system, and convection with the fluid.
\mathbf{g}	m.s^{-2}	Acceleration due to gravity. It is a vector pointing downwards (eg $(0, 0, -10)$).
γ_β^i	dimensionless	activity coefficient of species i in phase β
h_β	J.kg^{-1}	Specific enthalpy of fluid phase β . In the simplest case this is just $c_\beta T$ where c_β is the fluid's specific heat capacity.
H_κ	Pa	Henry coefficient for species κ which describes the solubility of the species in the aqueous phase.
I_m	$\text{mol.L}^{-1}.\text{s}^{-1}$	mineral reaction rate

Symbol	Units	Physical description
κ	dimensionless	Index representing species. For example, κ might parameterise water ($\kappa = 0$), air ($\kappa = 1$), and H_2 ($\kappa = 2$). It parameterises things that cannot be decomposed into other species, but can change phase. For instance, sometimes it might be appropriate to consider air as a single species, while at other times it might be appropriate to consider it to be a mixture of nitrogen and oxygen ($\kappa = 0$ and $\kappa = 1$, say).
k	m^2	Permeability tensor of rock.
$k_{r,\beta}$	dimensionless	Relative permeability of phase β . This is a nonlinear function of the independent variables. Often it is just a function of the phase's saturation, but with Klinkenberg effects it will be a function of the gas pressure too. In the single-phase, fully-saturated case it is unity.
K	Pa	Bulk modulus of the drained porous skeleton. I think in the anisotropic situation $1/K == \delta_{ij}\delta_{kl}C_{ijkl}$.
K_β^i	depends on reaction	equilibrium constant for secondary species i in phase β
K_f	Pa	Bulk modulus of the fluid.
k_m	$mol.m^{-2}.s^{-1}$	mineral rate constant
Λ	s^{-1}	Radioactive decay rate of a fluid species
λ	$J.s^{-1}.m^{-1}.K^{-1}$	Thermal conductivity of the rock-fluid system ($J.s^{-1}.m^{-1}.K^{-1} = kg.m.s^{-3}.K^{-1}$). It is a tensorial quantity to allow modelling of anisotropic situations, and is a function of the rock and fluid-phase's thermal conductivities
λ_{dry}	$J.s^{-1}.m^{-1}.K^{-1}$	Thermal conductivity of the rock-fluid system when aqueous phase saturation is zero.
λ_{wet}	$J.s^{-1}.m^{-1}.K^{-1}$	Thermal conductivity of the rock-fluid system when aqueous phase saturation is unity.
M	$kg.m^{-3}$	Mass density
μ	Pa.s	Dynamic viscosity measured in Pa.s or $kg.m^{-1}.s^{-1}$. This is a nonlinear function of the independent variables.
ν	dimensionless	Fraction of plastic-deformation energy that becomes heat energy. Probably $\nu = 1$ is correct.
ν_β^{ij}	dimensionless	stoichiometric coefficient of basis species j in equilibrium reaction for secondary species i
∇	m^{-1}	Spatial differential operator
Ω_m	dimensionless	mineral saturation ratio
P	Pa	Fluid porepressure
P_f	Pa	Measure of porepressure used in the effective stress. Often this is chosen to be $\sum_\beta S_\beta P_\beta$.
P_L	Pa	Langmuir pressure
Φ_β^j	moles per litre	Total concentration of basis species j in phase β
q	$kg.m^{-3}.s^{-1}$	Fluid source
q^j	$mol.L^{-1}.s^{-1}$	source of chemical species

Symbol	Units	Physical description
q^T	$\text{J.m}^{-3}.\text{s}^{-1}$	Heat source.
ρ	kg.m^{-3}	Fluid density
ρ_R	kg.m^{-3}	Grain density of the rock (so that $(1-\phi)\rho_R$ is the density of the dry porous rock).
ρ_{mat}	kg.m^{-3}	The mass-density of the fluid-filled porous solid $\rho_{\text{mat}} = (1-\phi)\rho^R + \phi \sum_{\beta} S_{\beta}\rho_{\beta}$.
ρ_L	kg.m^{-3}	Langmuir density
S	dimensionless	Saturation
S_l	dimensionless	saturation of aqueous phase
σ_{ij}^{tot}	Pa	Total stress. An externally applied mechanical force will create a nonzero σ^{tot} , and conversely, resolving σ^{tot} into forces yields the forces on nodes in the finite-element mesh.
σ_{ij}^{eff}	Pa	Effective stress.
t	s	Time
T	K	Temperature.
$\tau_0\tau_{\beta}$	dimensionless	The phase tortuosity, which includes a porous-medium dependent factor τ_0 and a coefficient $\tau_{\beta} = \tau_{\beta}(S_{\beta})$.
τ_L	s	Langmuir desorption time constant
θ	dimensionless	exponent in rate expression
\mathbf{u}	m.s^{-1}	Deformation vector of the porous solid.
\mathbf{v}	m.s^{-1}	Darcy velocity (volume flux)
\mathbf{v}_s	m.s^{-1}	Velocity of the solid $= \partial\mathbf{u}/\partial t$, where \mathbf{u} is the solid mechanical displacement vector of the porous solid.
\bar{V}_m	L.mol^{-1}	molar volume

Table 1.1: Notation introduced throughout Section 1.

1.2 Fluid flow

Mass conservation for fluid species κ is described by the continuity equation

$$0 = \frac{\partial M^{\kappa}}{\partial t} + M^{\kappa} \nabla \cdot \mathbf{v}_s + \nabla \cdot \mathbf{F}^{\kappa} + \Lambda M^{\kappa} - q^{\kappa} . \quad (1.1)$$

Here M is the mass of fluid per volume of rock (measured in kg.m^{-3}), \mathbf{v}_s is the velocity of the porous solid skeleton (measured in m.s^{-1}), \mathbf{F} is the flux (a vector, measured $\text{kg.s}^{-1}.\text{m}^{-2}$), Λ is a radioactive decay rate, and q is a source (measured in $\text{kg.m}^{-3}.\text{s}^{-1}$).

The coupling to the solid mechanics is via the $M \nabla \cdot \mathbf{v}_s$ term, as well as via changes in porosity and permeability described in Section 3.9 and Section 3.8. Coupling to heat flow and chemical reactions is via the equations of state used within the terms of Eqn (1.1), as well as the source term q^{κ} .

The species are parameterised by $\kappa = 1, \dots$. For example, κ might parameterise water, air, H_2 ,

a solute, and so on. κ parameterises things which cannot be decomposed into other species, but can change phase. For instance, sometimes it might be appropriate to consider air as a single species (say $\kappa = 1$), while other times it might be appropriate to consider it to be a mixture of nitrogen and oxygen ($\kappa = 1$ and $\kappa = 2$). To model chemical precipitation, a solid phase, which is distinct from the porous skeleton, may also be used (its relative permeability will be zero: see below).

1.2.1 Mass density: M

The mass of species κ per volume of rock is written as a sum over all phases present in the system:

$$M^\kappa = \phi \sum_{\beta} S_{\beta} \rho_{\beta} \chi_{\beta}^{\kappa} + (1 - \phi) C^{\kappa} \quad (1.2)$$

The solid's porosity is ϕ . S_{β} is the saturation of phase β (solid, liquid, gas, NAPL). ρ_{β} is the density of phase β . χ_{β}^{κ} is the mass fraction of component κ present in phase β . The final term represents fluid absorption into the porous-rock skeleton: C^{κ} is the mass of absorbed species per volume of solid rock-grain material, and its dynamics is specified in Section 1.5.1.

The density ρ_{β} is a function of pressure, mass fraction, and so on, as described by the equation of state used (Chapter 3).

The saturation and mass fractions must obey

$$\sum_{\beta} S_{\beta} = 1, \quad (1.3)$$

$$\sum_{\kappa} \chi_{\beta}^{\kappa} = 1 \quad \forall \beta. \quad (1.4)$$

1.2.2 Flux: \mathbf{F}

The flux is a sum of advective flux and diffusive-and-dispersive flux:

$$\mathbf{F}^{\kappa} = \sum_{\beta} \chi_{\beta}^{\kappa} \mathbf{F}_{\beta}^{\text{advective}} + \mathbf{F}_{\text{diffusion+dispersion}}^{\kappa}. \quad (1.5)$$

Advection

Advective flux is governed by Darcy's law. Each phase is assumed to obey Darcy's law. Each phase has its own density, ρ_{β} , relative permeability $k_{r,\beta}$, viscosity μ_{β} , and pressure P_{β} . These may all be nonlinear functions of the independent variables. With them, we can form the advective Darcy flux:

$$\mathbf{F}_{\beta}^{\text{advective}} = \rho_{\beta} \mathbf{v}_{\beta} = -\rho_{\beta} \frac{k k_{r,\beta}}{\mu_{\beta}} (\nabla P_{\beta} - \rho_{\beta} \mathbf{g}) \quad (1.6)$$

In this equation \mathbf{v}_β is the Darcy velocity (volume flux, measured in m.s^{-1}) in phase β . It is used below in the diffusive-and-dispersion flux too.

The absolute permeability is denoted by k and it is a tensor. The relative permeability¹ of phase β is denoted by $k_{r,\beta}$. It is always a function of the saturation(s), **but with Klinkenberg effects, it may also be a function of the gas pressure. Relative permeability can also be hysteretic, so that it depends on the history of saturation.**

In some circumstances $K_{ij}\nabla_j T$ is added to the above Darcy flux to model thermo-osmosis (with some K_{ij} tensor parameterising its strength), i.e. a gradient of temperature induces fluid flow.

Also note:

- The pressure in phase β is

$$P_\beta = P + P_{c,\beta} \quad (1.7)$$

where P is the reference pressure (usually taken to be the gas phase), and $P_{c,\beta}$ is the capillary pressure (nonpositive if the reference phase is the gas phase). The capillary pressure is often a function of saturation, and various forms have been coded into the PorousFlow module: see Section 3.3. **The capillary pressure relationship used in a model can have a great bearing on both the speed of convergence and the results obtained.**

The capillary pressure relationship can also be hysteretic, in that it can depend on the history of the saturation.

The pressure in a gas phase is the sum of the gas partial pressures: $P_{\text{gas}} = \sum_\kappa P_{\text{gas}}^\kappa$. (The partial pressure of a gaseous species is $P_{\text{gas}}^\kappa = P_{\text{gas}} N^\kappa / N$ where N is the number of molecules. This is Dalton's law.) The partial pressure concept is reasonable for dilute gases, but is less useful for dense gases.

- The mass-fraction of a species in the aqueous phase is often computed using Henry's law:

$$\chi_{\text{aqueous}}^\kappa = P_{\text{gas}}^\kappa / H_\kappa, \quad (1.8)$$

where H_κ is the Henry coefficient². Occasionally this law is not accurate enough, and there are more complicated alternatives. Chapter 3 discusses various equations of state that include computations of Henry's coefficients.

- **When a liquid and a gaseous phase exist, the simplest equation for vapour pressure is**

$$P_{\text{vapour}} = P_{\text{sat}}(T), \quad (1.9)$$

which is just the saturated vapour pressure of the liquid phase. This can set the temperature T from the gaseous pressure, or vice-versa, depending on the choice of independent

¹In reality relative permeability is actually a tensor (for example it's usually different in lateral and vertical directions) but is most often treated as a scalar, since it's hard to get parameters for the tensorial case. In the PorousFlow module it is treated as a scalar.

²Unfortunately, the Henry coefficient has multiple definitions which makes looking up numerical values complicated.

variables. A more complicated alternative is the Kelvin equation for vapour pressure that takes into account vapour pressure lowering due to capillarity and phase adsorption affects

$$P_{\text{vapour}} = \exp \left(\frac{M_w P_{c,l}(S_l)}{\rho_l R (T + 273.15)} \right) P_{\text{sat}}(T) . \quad (1.10)$$

Here M_w is the molecular weight of water; $P_{c,l} = P_{c,l}(S_l) \leq 0$ is the capillary pressure — the difference between aqueous (liquid water) and gas phase pressures — a function of S_l ; ρ_l is the aqueous (liquid water) density; R is the universal gas constant; T is the temperature; P_{sat} is the saturated vapour pressure of the bulk aqueous (liquid water) phase.

Diffusion and hydrodynamic dispersion

Diffusion and dispersion are proportional to the gradient of χ_β^κ . The TOUGH2 manual [1], in Appendix D, contains a nice discussion of diffusion-dispersion. Here we use the common expression

$$\mathbf{F}_{\text{diffusion+dispersion}}^\kappa = - \sum_\beta \rho_\beta \mathcal{D}_\beta^\kappa \nabla \chi_\beta^\kappa \quad (1.11)$$

Note \mathbf{F} is a vector, \mathcal{D} is a 2-tensor and ∇ a vector. The hydrodynamic dispersion tensor is

$$\mathcal{D}_\beta^\kappa = D_{\beta,T}^\kappa \mathcal{I} + \frac{D_{\beta,L}^\kappa - D_{\beta,T}^\kappa}{\mathbf{v}_\beta^2} \mathbf{v}_\beta \mathbf{v}_\beta , \quad (1.12)$$

where

$$D_{\beta,L}^\kappa = \phi \tau_0 \tau_\beta d_\beta^\kappa + \alpha_{\beta,L} |\mathbf{v}|_\beta , \quad (1.13)$$

$$D_{\beta,T}^\kappa = \phi \tau_0 \tau_\beta d_\beta^\kappa + \alpha_{\beta,T} |\mathbf{v}|_\beta . \quad (1.14)$$

These are called the longitudinal and transverse dispersion coefficients. d_β^κ is the molecular diffusion coefficient for component κ in phase β . $\tau_0 \tau_\beta$ is the tortuosity which includes a porous medium dependent factor τ_0 and a coefficient $\tau_\beta = \tau_\beta(S_\beta)$, and α_L and α_T are the longitudinal and transverse dispersivities. It is common to set the hydrodynamic dispersion to zero by setting $\alpha_{\beta,T} = 0 = \alpha_{\beta,L}$.

1.3 Heat flow

Energy conservation for heat is described by the continuity equation

$$0 = \frac{\partial \mathcal{E}}{\partial t} + \mathcal{E} \nabla \cdot \mathbf{v}_s + \nabla \cdot \mathbf{F}^T - \nu (1 - \phi) \sigma_{ij}^{\text{eff}} \frac{\partial}{\partial t} \epsilon_{ij}^{\text{plastic}} - q^T \quad (1.15)$$

Here \mathcal{E} is the energy per unit volume in the rock-fluid system, \mathbf{v}_s is velocity of the porous solid skeleton, \mathbf{F}^T is the heat flux, ν describes the ratio of plastic-deformation energy that gets

transferred to heat energy, σ_{ij}^{eff} is the effective stress (see Eqn (1.25)), $\epsilon_{ij}^{\text{plastic}}$ is the plastic strain, and q^T is a heat source.

The coupling to the solid mechanics is via the $\mathcal{E}\nabla \cdot \mathbf{v}_s$ term, the $\nu(1 - \phi)\sigma_{ij}^{\text{eff}} \frac{\partial}{\partial t} \epsilon_{ij}^{\text{plastic}}$ term, as well as via changes in porosity and permeability described in Section 3.9 and Section 3.8. Coupling to the fluid flow and chemical reactions is via the equations of state used within the terms of Eqn (1.15), as well as the source term q^T . Joule-Thompson effects³ may be included via the fluid properties (Chapter 3).

Here it is assumed the liquids and solid are in local thermal equilibrium i.e. there is a single local temperature in all phases. If this doesn't hold, one is also normally in the high-flow regime where the flow is non-Darcy as well.

Sometimes a term is added that captures the thermal power due to volumetric expansion of the fluid, $K\beta T\nabla \cdot \mathbf{v}$, where K is the bulk modulus of the fluid or solid, β is the thermal expansion coefficient, and \mathbf{v} is the Darcy velocity. This is not included in the Porous Flow module currently.

1.3.1 Energy density: \mathcal{E}

The energy per unit volume is

$$\mathcal{E} = (1 - \phi)\rho_R C_R T + \phi \sum_{\beta} S_{\beta} \rho_{\beta} \mathcal{E}_{\beta} + \sum_{\kappa} (1 - \phi) \rho^R \mathcal{E}_{\text{abs } \kappa} A^{\kappa} \quad (1.16)$$

The notation is: porosity ϕ ; grain density ρ_R ; specific heat capacity of rock C_R ; temperature T ; saturation of phase S_{β} (solid, liquid, gas, NAPL); density of phase ρ_{β} ; internal energy in phase \mathcal{E}_{β} ; **internal energy of absorbed species** $\mathcal{E}_{\text{abs } \kappa}$.

1.3.2 Heat flux: \mathbf{F}^T

The heat flux is a sum of heat conduction and convection with the fluid:

$$\mathbf{F}^T = -\lambda \nabla T + \sum_{\beta} h_{\beta} \mathbf{F}_{\beta} . \quad (1.17)$$

Here λ is the tensorial thermal conductivity of the rock-fluid system, which is a function of the thermal conductivities of rock and fluid phases. Usually λ will be diagonal but in anisotropic porous materials it may not be. The specific enthalpy of phase β is denoted by h_{β} , and \mathbf{F}_{β} is the advective Darcy flux.

³See for instance Eqn (1) of [2]

1.3.3 A simple example

Consider just the fluid terms of the heat-flow Eqn (1.15). For notational simplicity, consider just one fully-saturated phase so that $\mathcal{E} = \phi\rho_\beta\mathcal{E}_\beta$. Here \mathcal{E}_β has SI units J.kg^{-1} . Also consider the fluid flux with no conduction so that $\mathbf{F}^T = h_\beta\rho_\beta\mathbf{v}_\beta$. Integrating Eqn (1.15) over an arbitrary volume Ω , and employing integration by parts yields

$$0 = \frac{\partial}{\partial t} \int_{\Omega} \phi\rho_\beta\mathcal{E}_\beta + \int_{\partial\Omega} h_\beta\rho_\beta\mathbf{n} \cdot \mathbf{v}_\beta . \quad (1.18)$$

The first term on the RHS is the heat energy of the fluid in the fixed volume Ω . Since the volume is fixed, it is appropriate to use the specific heat at constant volume, C_v^β , in the expression for the fluid energy-per-mass:

$$\mathcal{E}_\beta = C_v^\beta T . \quad (1.19)$$

The second term on the RHS describes the change in heat energy due to fluid entering or leaving the volume with velocity \mathbf{v}_β . This change in heat energy can be decomposed into two parts in the following way. Imagine that fluid is entering the volume Ω . Firstly, imagine a new, slightly bigger volume Ω' that contains all the fluid that will enter Ω during the small time dt . The heat energy contained in the fluid within the small volume $\Omega' - \Omega$ is the extra heat energy added during fluid flow. This is simply

$$\int_{\partial\Omega} C_v T \rho \mathbf{n} \cdot \mathbf{v} dt . \quad (1.20)$$

We now know the total heat energy within the new volume, Ω' . The second step must be to squash this new volume back to the original Ω . This costs energy, which is just calculated using force \times distance. The force must be equal to the fluid pressure (multiplied by the infinitesimal area element), and the distance is exactly $\mathbf{n} \cdot \mathbf{v} dt$, meaning the second contribution is

$$\int_{\partial\Omega} P \mathbf{n} \cdot \mathbf{v} dt . \quad (1.21)$$

Collecting these ideas together Eqn (1.15) reads, in this simple example

$$0 = \frac{\partial}{\partial t} \phi \rho C_v T + \nabla \cdot \left(C_v T + \frac{P}{\rho} \right) \rho \mathbf{v} . \quad (1.22)$$

The superscript β denoting the phase number has been omitted.

Finally, two small observations might be useful.

1. Given the above discussion, it is clear that the enthalpy can be written as

$$h_\beta = \mathcal{E}_\beta + P_\beta / \rho_\beta . \quad (1.23)$$

2. For an ideal gas, $P/\rho = RT$, so the enthalpy,

$$h_\beta = C_v T + \frac{P}{\rho} = C_{\beta,v} T + RT = C_{\beta,p} T , \quad (1.24)$$

where $C_{\beta,p}$ is the specific heat capacity at constant pressure of the ideal gas phase.

1.4 Solid mechanics

Most of the solid mechanics used by the Porous Flow module is handled by the Tensor Mechanics module. This section provides a brief overview, concentrating on the aspects that differ from pure solid mechanics.

Denote the total stress tensor by σ^{tot} . An externally applied mechanical force will create a nonzero σ^{tot} , and conversely, resolving σ^{tot} into forces yields the forces on nodes in the finite-element mesh.

Denote the effective stress tensor by σ^{eff} . It is defined by

$$\sigma_{ij}^{\text{eff}} = \sigma_{ij}^{\text{tot}} + \alpha_B \delta_{ij} P_f . \quad (1.25)$$

The notation is as follows.

- P_f is a measure of porepressure. In single-phase, fully-saturated situations it is traditional to use $P_f = P_\beta$. However, for multi-phase situations $P_f = \sum_\beta S_\beta P_\beta$ is also used. **Yet other expressions involve Bishop's parameter.**
- α_B is the ‘‘Biot coefficient’’. This obeys $0 \leq \alpha_B \leq 1$. For a multi-phase system, the Biot coefficient is often chosen to be $\alpha_B = 1$. The Biot coefficient is interpreted physically by the following. If, by pumping fluid into a porous material, the P_f porepressure is increased by ΔP_f , and at the same time a mechanical external force applies an incremental pressure equaling $\alpha_B \Delta P_f$, then the volume of the porous solid remains static. (During this process, the porevolume and porosity will change, however, as quantified in Section 3.9.)

It is assumed that the elastic constitutive law reads

$$\sigma_{ij}^{\text{eff}} = E_{ijkl}(\epsilon_{kl}^{\text{elastic}} - \delta_{kl} \alpha_T T) , \quad (1.26)$$

with α_T being the thermal expansion coefficient of the drained porous skeleton, and $\epsilon_{kl} = (\nabla_k u_l + \nabla_l u_k)/2$ being the usual total strain tensor (u is the deformation of the porous solid), which can be split into the elastic and plastic parts, $\epsilon = \epsilon^{\text{elastic}} + \epsilon^{\text{plastic}}$, and E_{ijkl} being the elasticity tensor (the so-called ‘‘drained’’ version). The generalisation to large strain is obvious. The inverse of the constitutive law is

$$\epsilon_{ij}^{\text{elastic}} - \delta_{kl} \alpha_T T = C_{ijkl} \sigma_{ij}^{\text{eff}} , \quad (1.27)$$

with C being the compliance tensor.

It is assumed that the conservation of momentum is

$$\rho_{\text{mat}} \frac{\partial v_s^j}{\partial t} = \nabla_i \sigma_{ij}^{\text{tot}} + \rho_{\text{mat}} b_j = \nabla_i \sigma_{ij}^{\text{eff}} - \alpha_B \nabla_j P_f + \rho_{\text{mat}} b_j , \quad (1.28)$$

where $\mathbf{v}_s = \partial \mathbf{u} / \partial t$ is the velocity of the solid skeleton, ρ_{mat} is the mass-density of the material (this is the ‘‘undrained’’ density: $\rho_{\text{mat}} = (1 - \phi) \rho^R + \phi \sum_\beta S_\beta \rho_\beta$), and b_j are the components

of the external force density (for example, the gravitational acceleration). Here any terms of $O(v^2)$ have been explicitly ignored, and it's been assumed that to this order the velocity of each phase is identical to the velocity of the solid skeleton (otherwise there are terms involving $\partial \mathbf{F} / \partial t$ on the left-hand side).

It is assumed that the *effective stress not the total stress* enters into the constitutive law (as above), and the plasticity, and the insitu stresses, and almost everywhere else. One exception is specifying Neumann boundary conditions for the displacements where the total stresses are being specified, as can be seen from Eqn (1.28). Therefore, MOOSE will use effective stress, and not total stress, internally. If one needs to input or output total stress, one must subtract $\alpha_B \nabla_j P_f$ from MOOSE's stress.

1.5 Chemical reactions

1.5.1 Adsorption and desorption

The fluid mass Eqn 1.2 contains contributions from adsorped species: C^κ is the mass of adsorped species per volume of solid rock-grain material. Its dynamics involves no advective flux terms, as PorousFlow assumes that the adsorped species are trapped in the solid matrix. The governing equation is

$$0 = \frac{\partial}{\partial t}(1 - \phi)C^\kappa + (1 - \phi)C^\kappa \nabla \cdot \mathbf{v}_s + L^\kappa . \quad (1.29)$$

The $(1 - \phi)$ terms account for the porosity of the solid skeleton (so $(1 - \phi)C^\kappa$ is the mass of adsorped species per volume of porous skeleton), the coupling to the solid mechanics is via the second term, and L^κ governs the adsorption-desorption process.

Currently, PorousFlow assumes that C^κ is a primary MOOSE variable (that is, defined in the [Variables] block of the MOOSE input file). This is in contrast to the remainder of PorousFlow where the primary variables can be arbitrary (so that variable switching and persistent variables can be used in difficult problems). Usually C^κ will have `family=MONOMIAL` and `order=CONSTANT`.

PorousFlow assumes that L follows the Langmuir form:

$$L = \frac{1}{\tau_L} \left(C - \frac{\rho_L P}{P_L + P} \right) \quad (1.30)$$

Here L is the mass-density rate ($\text{kg.m}^{-3}.\text{s}^{-1}$) of material moving from the adsorped state to the porespace. The notation is as follows.

- ρ_L is the so-called Langmuir density, which is adsorped-gas-mass per volume of solid rock-grain material (kg.m^{-3}). In terms of the often used Langmuir volume, V_L , it is $\rho_L = V_L \times (\text{gas density at STP})$.

- P_L is the Langmuir pressure (Pa)
- P is the relevant phase pressure (which will be the pressure of the gas phase). Currently, PorousFlow assumes that P is a primary MOOSE variable (that is, defined in the [Variables] block of the MOOSE input file). This is in contrast to the remainder of PorousFlow where the primary variables can be arbitrary.
- τ_L is the time constant. Two time constants may be defined: one for desorption, and one for adsorption, so

$$\tau_L = \begin{cases} \tau_{L, \text{ desorption}} & \text{if } C > \frac{\rho_L P}{P_L + P} \\ \tau_{L, \text{ adsorption}} & \text{otherwise} \end{cases} \quad (1.31)$$

A mollifier may also be defined to mollify the step-change between desorption and adsorption.

1.5.2 General geochemistry

The first step in defining the chemistry is to choose a set of basis species in the liquid phase, such that every other chemical species, including minerals (now all referred to as secondary species) can be expressed via combinations of these basis species.

Let j index the basis species in the liquid phase, denoted by \mathcal{A}_l^j , and i index the secondary species, \mathcal{A}_β^i . Then the reactions that connect these are given by

$$\mathcal{A}_\beta^i \rightleftharpoons \sum_j \nu_\beta^{ji} \mathcal{A}_l^j \quad (1.32)$$

where the ν_β^{ji} are the stoichiometric coefficients.

Now define the total concentration of basis species j in fluid phase β , denoted Ψ_β^j , by

$$\Psi_\beta^j = \delta_{l\beta} C_l^j + \sum_i \nu_\beta^{ji} C_\beta^i \quad (1.33)$$

where the C_β^i is the concentration of secondary species i in phase β , C_l^j is the concentration of basis species j in the liquid phase, and $\delta_{l\beta}$ is the Kronecker delta ($= 1$ if $\beta = l$, else $= 0$).

The concentrations of the secondary species C_β^i are obtained from the basis species concentrations by mass action equations of the form

$$C_\beta^i = \frac{K_\beta^i}{\gamma_\beta^i} \prod_j (\gamma_l^j C_l^j)^{\nu_\beta^{ji}} \quad (1.34)$$

where K_β^i is the equilibrium constant and γ_β^i is the activity coefficient of species i in phase β . There are various possible relationships for computing activity coefficients that depend both on properties of that species (e.g. valence and ionic radius) and the total ionic strength (which depends on all the ion concentrations).

In all these, the concentration units are moles per litre.

Now define the total concentration accumulation of basis species j by M^j , which is given by

$$M^j = \phi \sum_{\beta} S_{\beta} \Psi_{\beta}^j \quad (1.35)$$

where the sum in Eqn (1.35) is over the fluid phases only, and as before ϕ is the porosity and S_{β} is the phase saturation.

The continuity equation is then

$$0 = \frac{\partial M^j}{\partial t} + M^j \nabla \cdot \mathbf{v}_s + \nabla \cdot \mathbf{F}^j - q^j . \quad (1.36)$$

Here \mathbf{F} is the flux (a vector, measured $\text{mol.s}^{-1}.\text{L}^{-1}$), and q^j is a source (measured in $\text{mol.L}^{-1}.\text{s}^{-1}$).

For minerals indexed by m one has

$$\frac{\partial \phi_m}{\partial t} = \bar{V}_m I_m \quad (1.37)$$

where ϕ_m is the volume fraction of that mineral, \bar{V}_m is the molar volume, and I_m is the mineral reaction rate. The porosity is related to the mineral volume fractions by

$$\phi = 1 - \sum_m \phi_m \quad (1.38)$$

The total flux \mathbf{F}^j in Eqn (1.36) is given by

$$\mathbf{F}^j = \sum_{\beta} (\mathbf{v}_{\beta} - \phi S_{\beta} \mathcal{D}_{\beta}^j \nabla) \Psi_{\beta}^j \quad (1.39)$$

which incorporates the advective and diffusive transport of chemical species in the sum over the fluid phases. \mathcal{D}_{β}^j is the hydrodynamic dispersion tensor in phase β of species j , including effects of tortuosity, dispersion etc, as in Eqn (1.12). It's common to assume the this tensor is the same for all species in that phase.

First define the mineral saturation ratio Ω_m by

$$\Omega_m = K_m^{-1} \prod_j (C^j)^{\nu_{mj}} (\gamma^j)^{\nu_{mj}} \quad (1.40)$$

where K_m is the equilibrium constant for the reaction. There are a variety of possible rate expressions. One of the simplest is to write

$$I_m = \pm k_m A_m [1 - \Omega_m^{\theta}]^{\eta} \quad (1.41)$$

where the rate I_m is positive for dissolution and negative for precipitation, A_m is the specific reactive surface area per litre for mineral m , and the exponents θ and η are specific to each mineral reaction, but are often both taken to be unity. The rate constant k_m ($\text{mole}/(\text{m}^2 \text{ s})$) is a function of temperature.

1.6 Example use case: Single-phase single-component fully-saturated fluid

Since the number of phases and components is unity, the subscripts β and κ may be dropped. Since $\chi = 1$ there is no diffusion nor dispersion. Assume that the relative permeability is unity. Mass conservation of the fluid (Eqns (1.1), (1.2), (1.5), (1.6), (1.11), (1.12), (1.13) and (1.14)) reads

$$0 = \frac{\partial}{\partial t} \phi \rho + \nabla \cdot (\phi \rho \mathbf{v}_s) - \nabla_i \cdot \left(\frac{\rho k_{ij}}{\mu} (\nabla_j P - \rho g_j) \right) - q^{\text{fluid}}, \quad (1.42)$$

where P is the porepressure of the fluid and will be one of MOOSE's variables in this case.

Make the following assumptions:

- the fluid's internal energy is $\mathcal{E} = cT$, where c is the fluid's specific heat capacity;
- the fluid's specific enthalpy is $h = cT$;
- the rock-fluid thermal conductivity is a weighted sum of the rock-grain and fluid thermal conductivities: $\lambda = (1 - \phi)\lambda^R + \phi\lambda^f$

The heat-energy conservation equation (Eqn (1.15), (1.16) and (1.17)) is

$$\begin{aligned} 0 = & \frac{\partial}{\partial t} ((1 - \phi)\rho_R C_R T + \phi \rho c T) + \nabla \cdot (((1 - \phi)\rho_R C_R T + \phi \rho c T) \mathbf{v}_s) \\ & - \nabla \cdot (((1 - \phi)\lambda^R + \phi\lambda^f) \nabla T) - \nabla \cdot \left(cT \frac{\rho k_{ij}}{\mu} (\nabla_j P - \rho g_j) \right) - q^T, \end{aligned} \quad (1.43)$$

where T is the temperature of the fluid-rock system and will be one of MOOSE's variables in this case. Under the additional assumptions that the following parameters are constant (or very nearly so): ϕ , ρ_R , C_R , c , λ^R , λ^f , this equations reads

$$0 = C \frac{\partial T}{\partial t} + C \nabla \cdot (T \mathbf{v}_s) - \lambda \nabla^2 T - \nabla \cdot \left(\rho c T \frac{k_{ij}}{\mu} (\nabla_j P - \rho g_j) \right) - q^T, \quad (1.44)$$

with $C = (1 - \phi)\rho_R C_R + \phi \rho c$ being the macroscopic heat capacity.

Eqns (1.42) and (1.44) with $\mathbf{v}_s = 0$ are the equations of FALCON. Using the assumption $\rho = \rho_0 \exp(P/K_f - \alpha_T^f T)$, where K_f is the constant bulk modulus of the fluid, and α_T^f is its constant thermal expansion coefficient, these two equations are those of FLAC's THM.⁴

The effective stress is given in Eqn (1.25), with $P_f = P$. The elastic constitutive equation is Eqn (1.26). Conservation of momentum is given in Eqn (1.28), with $\rho_{\text{mat}} = (1 - \phi)\rho^R + \phi\rho$. The variables are the 3 displacements (in 3 dimensional problems), u_i .

⁴There is one possible inconsistency: FLAC's Eqn(12) of FLAC7.0 "Thermal Analysis (2011) writes $\beta = n\beta_f + (1 - n)\beta_g$, whereas the formulation here would give $\beta = n\beta_f + (\alpha_B - n)\beta_g$. FLAC has a habit of sneakily putting $\alpha_B = 1$ without saying, because some of their formulation is only correct to $\alpha_B = 1$, so I'm not sure whether this is FLAC's error/assumption, or an error in the porosity evolution equation Eqn (3.21) here.

The evolution of porosity is given in Eqn (3.21). In many situations of interest, the change in porosity is very small.

In summary, the variables for this case can be the porepressure P , the temperature T , and the three displacements u_i .

The following parameters depend, in principle, on the porepressure, temperature and (derivatives of) the displacements: porous-rock insitu permeability tensor k_{ij} ; rock-grain density ρ_R ; rock-grain specific heat capacity C_R ; rock-grain thermal conductivity λ^R ; fluid density ρ ; fluid viscosity μ ; fluid thermal conductivity λ^f ; fluid specific heat capacity c . All these can be coded into a suite of Materials, as described in Chapter 3.

Chapter 2

Boundaries and line-sinks

2.1 Boundaries

MOOSE's Dirichlet and Neumann boundary conditions enable simulation of simple scenarios. The Porous Flow module includes a very flexible boundary condition that allows many different scenarios to be modelled. The boundary condition is built by adding various options to a basic sink in the following way.

The basic sink is

$$s = f(t, x) , \tag{2.1}$$

where f is a MOOSE Function of time and position on the boundary. If $f > 0$ then the boundary condition will act as a sink, while if $f < 0$ the boundary condition acts as a source. If applied to a fluid-component equation, the function f has units $\text{kg.m}^{-2}.\text{s}^{-1}$. If applied to the heat equation, the function f has units $\text{J.m}^{-2}.\text{s}^{-1}$. These units are potentially modified if the extra building blocks enumerated below are used.

The basic sink may be multiplied by a MOOSE Functions of the pressure of a fluid phase *or* the temperature:

$$s = f(t, x) \times g(P^\beta) \quad \text{or} \quad s = f(t, x) \times g(T) . \tag{2.2}$$

Here the units of $f \times g$ are $\text{kg.m}^{-2}.\text{s}^{-1}$ (for fluids) or $\text{J.m}^{-1}.\text{s}^{-1}$ (for heat). Some commonly use forms have been hard-coded into the Porous Flow module for ease of use in simulations:

- a piecewise linear function (for simulating fluid or heat exchange with an external environment via a conductivity term, for instance);
- a half-gaussian (for simulating evapotranspiration, for instance)
- a half-cubic (for simulating evapotranspiration).

In addition, the sink may be multiplied by any or all of the following quantities

- Fluid relative permeability
- Fluid mobility ($k_{ij}n_in_jk_r\rho/\nu$, where n is the normal vector to the boundary)
- Fluid mass fraction
- Fluid enthalpy
- Fluid internal energy
- Thermal conductivity

It is likely that multiplication by generalised concentrations will also be needed.

For example, a boundary condition corresponding to injection of fluid at a fixed temperature could involve: (1) using a Dirichlet condition for temperature; (2) using $f = -1$ without any multiplicative factors. More complicated examples with heat and fluid injection and production are detailed in the test suite documentation `newton_cooling` and `sinks`.

Commonly, if fluid or heat is exiting the porous material, multiplication by relative permeability, mobility, mass fraction, enthalpy or thermal conductivity is necessary, while if fluid or heat is entering the porous material this multiplication is not necessary. Two sinks can be constructed in a MOOSE input file: one involving “production” which is only active for $P > P_e$ using a Piecewise-linear sink (here P_e is the environmental pressure); and one involving “injection”, which is only active for $P < P_e$ using a piecewise-linear sink multiplied by the appropriate factors.

2.2 Line sinks

Polyline sinks and sources are modelled as sequences of discrete points:

$$\text{polyline} \sim \{x_0, x_1, x_2, \dots, x_N\} . \quad (2.3)$$

The sink is

$$s = \sum_i f(P_i)w_i\delta(x - x_i) . \quad (2.4)$$

Here s is a volume source, measured in $\text{kg.m}^{-3}.\text{s}^{-1}$ (or $\text{J.m}^{-3}.\text{s}^{-1}$ for heat flow), which when integrated over the finite element yields just the sink “strength”, f , which has units kg.s^{-1} for fluid flow, or J.s^{-1} for heat flow. The strength, f , is a function of porepressure or temperature, and may involve other quantities, as enumerated below. The convention followed is:

- A sink has $s > 0$. This removes fluid or heat from the simulation domain.
- A source has $s < 0$. This adds fluid or heat to the simulation domain.

The input parameters for each PorousFlow line sink involve a plaintext file whose lines are space-separated quantities:

$$w_i \ x_i \ y_i \ z_i \quad (2.5)$$

The weighting terms, w_i , are for user convenience, but for the Peaceman borehole case they are the borehole radius at point x_i .

The basic sink may be multiplied by any or all of the following quantities

- Fluid relative permeability
- Fluid mobility ($k_r \rho / \nu$)
- Fluid mass fraction
- Fluid enthalpy
- Fluid internal energy

That is, f in Eqn (2.4) may be replaced by $f k_r$, $f \times \text{mobility}$, etc. (The units of $f k_r$, $f \times \text{mobility}$, etc, are kg.s^{-1} for fluid flow, or J.s^{-1} for heat flow.) All these additional multiplicative factors are evaluated at the nodal positions, not at point x_i , to ensure superior numerical convergence (see Sec 4.2). For instance, a fluid sink may extract fluid at a given rate, and therefore in a simulation that includes, the same sink multiplied by fluid enthalpy should be applied to the temperature variable.

2.2.1 Poly-line sinks as functions of porepressure or temperature

PorousFlowPolyLineSinks are a special case of the general line sink. The function, f in Eqn (2.4) is assumed to be a piecewise linear function of porepressure or temperature. In addition, a multiplication by the line-length associated to x_i is also performed. Specifically:

$$f(P_i) = \frac{1}{2} (|x_i - x_{i-1}| + |x_i - x_{i+1}|) L(P_i) , \quad (2.6)$$

where L is a piecewise-linear function, specified by the user in the MOOSE input file.

These types of sinks are useful in describing groundwater—surface water interactions via streams and swamps. Often a riverbed conductance, measured in $\text{kg.Pa}^{-1}.\text{s}^{-1}$ is defined, which is

$$C = \frac{k_{zz} \rho}{L \mu} L_{\text{seg}} W_{\text{seg}} . \quad (2.7)$$

Here k_{zz} is the vertical component of the permeability tensor, ρ is the fluid density, μ is the fluid viscosity, and L is a distance variable related to the riverbed thickness. The other parameters are L_{seg} and W_{seg} , which are, respectively, the length and width of the segment of river that the point x_i is representing. The multiplication by L_{seg} is already handled by Eqn (2.6), and

the other terms of C will enter into the piecewise linear function, L . Three standard types of L are used in groundwater models.

- A perennial stream, where fluid can seep from the porespace to the stream, and vice versa. Then $L \propto (P - P_{\text{atm}})$, where the proportionality constant involves the terms in C , and P_{atm} involves the river stage height.
- An ephemeral stream, where fluid can only seep from the porespace to the stream, but not viceversa has $L \propto (P - P_{\text{atm}})$ if $P > P_{\text{atm}}$, and zero otherwise. This is a pure sink since $s > 0$ always.
- A rate-limited stream, where fluid can exchange between the groundwater and stream, but the rate is limited. This can be modelled by using a piecewise linear L that does not exceed given limits.

2.2.2 Peaceman Boreholes

Wellbores are implemented using the method first described by Peaceman [3]. Here f is a special function (measured in kg.s^{-1} in standard units) defined in terms of the pressure at a point at the wall of the wellbore (which is an input parameter)

$$P_{\text{wellbore}}(x_i) = P_{\text{bot}} + \gamma \cdot (x_i - x_i^{\text{bot}}) . \quad (2.8)$$

The form of f is

$$f(P_i, x_i) = W |C(P_i - P_{\text{wellbore}})| \frac{k_r \rho}{\mu} (P_i - P_{\text{wellbore}}) \quad (2.9)$$

For boreholes that involve heat only, the P in the above expressions should be replaced by the temperature.

There are a number of parameters in these expressions:

- P_{bot} is an input parameter. It is the pressure at the bottom of the wellbore.
- x_i^{bot} is the position of the bottom of the wellbore.
- γ is a weight vector pointing downwards (product of fluid density and gravity). This means that $P_{\text{wellbore}}(x_i)$ will be the pressure at point x_i in the wellbore, due to gravitational head. If these gravitational effects are undesirable, the user may simply specify $\gamma = (0, 0, 0)$.
- k_r , ρ , and μ are the fluid relative permeability, density and viscosity. Hence the term $k_r \rho / \mu$ is the mobility, so users should choose to multiply by the mobility when using `PorousFlowPeacemanBorehole`. Recall that all the multiplicative factors, including mobility, are evaluated at the nodal positions, not the position x_i . This ensures superior numerical convergence.

- In the PorousFlow implementation C is called the “character” of the wellbore. There are two standard choices (note that f depends only on the absolute value $|C|$):

- $C = 1$ for $P > P_{\text{wellbore}}$, and zero otherwise. In this case the wellbore is called a “production” wellbore, and it acts as a sink, removing fluid from the porespace.
- $C = -1$ for $P < P_{\text{wellbore}}$, and zero otherwise. In this case the wellbore is called an “injection” wellbore, and it acts as a source, adding fluid to the porespace.

Generalising, C may be chosen so that $|C| \neq 1$, which is useful for time-varying borehole strengths. The above two choices are generalised to: if $C > 0$ the wellbore is active only for $P > P_{\text{wellbore}}$ (and has zero strength otherwise); if $C < 0$ the wellbore is active only for $P < P_{\text{wellbore}}$.

- The function W is called the well-constant of the wellbore, and is measured in units of length³.

Peaceman described the form of W by performing 2D analytical and numerical calculations to obtain the following formulae

$$W = 2\pi\sqrt{\kappa_{xx}\kappa_{yy}}L_z/\ln(r_e/r_{\text{bh}}) , \quad (2.10)$$

In this formula: the borehole is oriented along the z direction; κ_{xx} and κ_{yy} are the diagonal components of the permeability tensor in the (x, y) plane; L_z is the length of the borehole, r_{bh} is the borehole radius (an input parameter, which is encoded in the plaintext file of points: see Eqn (2.5)); and, r_e is the effective borehole radius. For a cell-centred finite-difference approach, Peaceman found that

$$r_e = 0.28 \frac{\sqrt{\sqrt{\kappa_{xx}/\kappa_{yy}}L_x^2 + \sqrt{\kappa_{yy}/\kappa_{xx}}L_y^2}}{(\kappa_{xx}/\kappa_{yy})^{1/4} + (\kappa_{yy}/\kappa_{xx})^{1/4}} = 0.2 \frac{\sqrt{\frac{1}{2}\sqrt{\kappa_{xx}/\kappa_{yy}}L_x^2 + \frac{1}{2}\sqrt{\kappa_{yy}/\kappa_{xx}}L_y^2}}{\frac{1}{2}(\kappa_{xx}/\kappa_{yy})^{1/4} + \frac{1}{2}(\kappa_{yy}/\kappa_{xx})^{1/4}} . \quad (2.11)$$

Here L_x and L_y are the finite-difference spatial sizes. Other authors have generalised Peaceman’s approach to writing W for different geometrical situations. Some of these are contained in the article by Chen and Zhang [4], where they show that for a finite element situation with square elements of size L , the borehole at a nodal position, and isotropic permeability

$$r_e = 0.113L . \quad (2.12)$$

Note that the 0.113 is substantially different to Peaceman’s 0.2, demonstrating that this method of introducing wellbores dependent on the geometry. The user may specify this quantity via the `re_constant` input parameter (which defaults to Peaceman’s 0.28).

Chapter 3

Equations of state and other functions

This chapter briefly outlines the fluid properties available to the PorousFlow module.

3.1 Fluid properties

The PorousFlow module accesses fluids properties such as density or viscosity from the fluid descriptions available in the `FluidProperties` module. This module provides a consistent interface to fluid properties, enabling the user to easily swap fluids in the input file.

Each of the fluids available in the `FluidProperties` module provide the following thermophysical properties:

1. density
2. viscosity
3. internal energy
4. enthalpy
5. isobaric specific heat capacity
6. isochoric specific heat capacity
7. thermal conductivity
8. specific entropy

as well as derivatives of each of these with respect to their variables.

The following are the fluids that are currently available in the `FluidProperties` module.

3.1.1 Ideal gas

A simple formulation that is suitable for ideal gases, where properties are derived from the ideal gas law

$$P = \rho RT. \quad (3.1)$$

3.1.2 Simplified fluid

A simplified fluid based on a constant bulk modulus density fluid, where density is

$$\rho = \rho_0 \exp(P/K_f - \alpha_f T), \quad (3.2)$$

where K_f (the bulk modulus) and α_f (the thermal expansion coefficient) are constant, is also available.

In this formulation, viscosity and thermal conductivity are constant, while internal energy and enthalpy are given by

$$\mathcal{E} = C_v T, \quad h = C_p T, \quad (3.3)$$

where C_v (the specific heat capacity at constant volume), and C_p (the specific heat capacity at constant pressure), are both constant.

3.1.3 Water and steam

The IAPWS Industrial Formulation 1997 [5] for the properties of water and steam are implemented in the `FluidProperties` module. This formulation calculates properties of water and steam using pressure and temperature as inputs. The IAPWS-IF97 formulation is split into five different regions in the phase diagram that are valid for

$$\begin{aligned} 273.15 \text{ K} \leq T \leq 1073.15 \text{ K}, \quad p \leq 100 \text{ MPa}, \\ 1073.15 \text{ K} < T \leq 2273.15 \text{ K}, \quad p \leq 50 \text{ MPa}. \end{aligned} \quad (3.4)$$

All five regions are implemented in the `fluid_properties` module. To avoid iteration in region 3 of the IAPWS-IF97 formulation, the backwards equations from [6] are implemented.

Viscosity is calculated using the IAPWS 2008 formulation [7]. Note that the critical enhancement has not been implemented.

Thermal conductivity is calculated using the IAPS 1985 formulation [8]. Although there is a newer formulation available [9], it is significantly more complicated, so has not been implemented yet.

3.1.4 Brine

A high-precision and consistent formulation for fluid properties for binary salt (NaCl) and water mixtures at pressures and temperatures of interest is provided by Driesner and Heinrich [10] and Driesner [11]. Density, enthalpy, internal energy and specific heat capacity are calculated using the formulations provided in [10] and [11].

Viscosity and thermal conductivity of brine are calculated using the formulation of Phillips et al [12].

Brine vapor pressure is calculated using the formulation presented in Haas Jr.[13]. Solubility of solid salt (halite) in water is given by Potter et al [14].

3.1.5 CO₂

Fluid properties for CO₂ are mainly calculated using the Span and Wagoner equation of state [15]. This formulation uses density and temperature as the primary variables with which to calculate properties such as density, enthalpy and internal energy. However, the `FluidProperties` module uses pressure and temperature in its interface, which is suitable for use in the Porous-Flow module. As a result, CO₂ properties are typically calculated by first calculating density iteratively for a given pressure and temperature. This density is then used to calculate the other properties, such as internal energy, directly.

Viscosity is calculated using the formulation presented in Fenghour et al [16], while thermal conductivity is taken from Scalabrin et al [17].

3.1.6 Methane

Density of methane is calculated assuming an ideal gas, while all other properties are calculated using the formulations provided in Irvine and Liley [18].

3.2 CO₂ and brine

3.3 Capillary pressure relationships

3.3.1 Van Genuchten

van Genuchten's capillary-pressure relationship [19] is

$$S_{\text{eff}} = \begin{cases} 1 & \text{if } P \geq 0 , \\ (1 + (-\alpha P)^{1/(1-m)})^{-m} & \text{if } P < 0 . \end{cases} \quad (3.5)$$

or

$$P_c = \begin{cases} 0 & \text{if } S_{\text{eff}} \geq 1.0 , \\ \frac{1}{\alpha} (S_{\text{eff}}^{-1/m} - 1)^{1-m} & \text{if } S_{\text{eff}} < 1 \end{cases} \quad (3.6)$$

The effective saturation has been denoted by S_{eff} and P is the porepressure, which is the *negative* of the capillary pressure: $P = -P_c$. Here α and m are user-defined parameters. The parameter m must satisfy

$$0 < m < 1 . \quad (3.7)$$

Sometimes the van Genuchten function is defined in terms of the parameter

$$n = \frac{1}{1-m} > 1 . \quad (3.8)$$

In van Genuchten's paper, he finds good fits with experimental data for various soils and rock when the parameter m ranges between about 0.5 and 0.9 (meaning $2 < n < 10$, roughly), and α is between $4 \times 10^{-5} \text{ Pa}^{-1}$ and $2 \times 10^{-4} \text{ Pa}^{-1}$. Figure 3.1 shows the shape of the van Genuchten suction, P_c , as a function of S_{eff} .

Numerically there are three important features of Eqn (3.5).

1. S_{eff} is a monotonically decreasing function of P_c , which is necessary for a unique solution.
2. $P_c \rightarrow \infty$ as $S_{\text{eff}} \rightarrow 0$. As mentioned above, this is not justifiable physically, but numerically it is extremely advantageous over $P_c \rightarrow P_c^0 < \infty$, as this latter version often causes algorithms to “get stuck” around $S_{\text{eff}} = 0$. As also mentioned above, because of the low relative permeability around S_{eff} , physically realistic problems rarely explore the $S_{\text{eff}} \sim 0$ region.
3. $S_{\text{eff}} \rightarrow 1$ and $dS_{\text{eff}}/dP_c \rightarrow 0^+$ as $P_c \rightarrow 0$, for all m . This ensures that there is continuity in the porepressure, P , and the derivative dS/dP around full saturation (remember that by definition $S_{\text{eff}} = 1$ for $P_c < 0$). Also $d^2S_{\text{eff}}/dP_c^2 \rightarrow 0$ as $P_c \rightarrow 0^+$ if $m > 0.5$.

I encourage users to set $m > 0.5$.

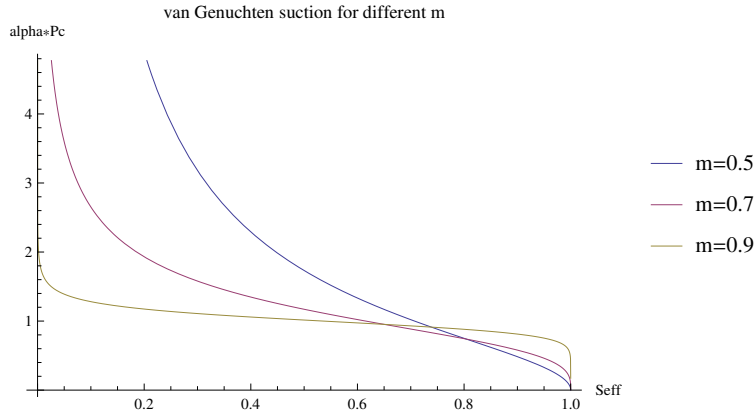


Figure 3.1: αP_c as a function of S_{eff} as given by van Genuchten's expression Eqn (3.5). Three values of m are shown: 0.5, 0.7 and 0.9 (the $m = 0.5$ case has largest P_c around $S_{\text{eff}} \sim 0$).

3.3.2 Broadbridge-White

The Broadbridge-White [20] capillarity relationship valid for small K_n is

$$S_{\text{eff}} = S_n + (S_s - S_n) \frac{c}{1 + L(x)} . \quad (3.9)$$

where

$$x = (c - 1)e^{c-1-cP/\lambda} , \quad (3.10)$$

and $L(x)$ is the Lambert W-function that satisfies $L(z)e^{L(z)} = z$. This is of limited use in real simulations, and is only used in the Porous Flow module for comparison with the analytical solutions of Broadbridge and White [20] and Warrick, Lomen and Islas [21] for multi-phase infiltration and drainage problems.

3.3.3 Rogers-Stallybrass-Clements

The Rogers-Stallybrass-Clements capillary relationship [22] is

$$S_{\text{eff}} = \frac{1}{\sqrt{1 + \exp((P_c - A)/B)}} , \quad (3.11)$$

when the oil viscosity is exactly twice the water viscosity. This is of limited use in real simulations, and is only used in the Porous Flow module for comparison with the analytical solutions offered by the authors for multi-phase infiltration and drainage problems.

3.4 Density

For simple calculations with water, the fluid bulk modulus, B , may be taken to be constant, so that the fluid density is

$$\rho = \rho_0 e^{P/B} , \quad (3.12)$$

where P is the fluid pressure. It is common to use $\rho_0 = 1000 \text{ kg.m}^{-3}$ and $B = 2 \text{ GPa}$.

3.5 Tortuosity and diffusion coefficients

The tortuosity may be constant, or it may follow the Millington-Quirk [23] form

$$\tau_\beta = \phi^{1/3} S_\beta^{10/3} \quad (3.13)$$

The molecular diffusion coefficients, d_β^κ , are assumed to be constant.

3.6 Effective fluid pressure

The effective fluid pressure used in the definition of the fluid effective stress Eqn (1.25) is

$$P = \sum_{\beta} S_\beta P_\beta . \quad (3.14)$$

3.7 Enthalpy and energy-density

The specific enthalpy h_β , for a phase β , may be defined using one of the high-precision equation of states (eg see Section 3.1.3), or the relationship

$$h_\beta = \mathcal{E}_\beta + a P_\beta / \rho_\beta . \quad (3.15)$$

It is usually appropriate to use $a = 1$, however for comparison with other codes it may be useful to use $a = 0$.

The internal energy density \mathcal{E}_β for a phase may be defined using one of the high-precision equation of states, or the relationship

$$\mathcal{E}_\beta = C_v T . \quad (3.16)$$

may be used.

3.8 Permeability

The porous-material's insitu permeability tensor can be constant, or it can take one of the following forms

3.8.1 Exponential

$$k_{ij} = k_{ij}^0 e^{a\phi} , \quad (3.17)$$

where ϕ is the porosity, and a is a user-defined constant.

3.8.2 Kozeny-Carman

Using the Kozeny-Carman relationship ¹

$$k_{ij} = k_{ij}^0 \frac{\phi^n}{(1 - \phi)^m} . \quad (3.18)$$

Here n and m are user-defined constants.

3.8.3 Permeability with a solid phase

A solid phase (from chemical precipitation, for instance) can be included in the framework described herein simply by setting its relative permeability to zero. However, in this case, the absolute permeability of the porous material should be

$$k = k^{\text{without solid phase}} (1 - S_s)^2 , \quad (3.19)$$

where S_s is the solid-phase saturation.

3.9 Porosity

Porosity may be fixed at a constant value, or it may be a function of the effective porepressure, the volumetric strain and/or the temperature, as discussed below.

The evolution of the porosity² is governed by

$$\frac{\partial \phi}{\partial t} = (\alpha_B - \phi) \frac{\partial}{\partial t} (\epsilon_{ii}^{\text{total}} - \alpha_T T) + \frac{(1 - \alpha_B)(\alpha_B - \phi)}{K} \frac{\partial P_f}{\partial t} . \quad (3.20)$$

Here K is the bulk modulus of the drained porous skeleton: $1/K = \delta_{ij} \delta_{kl} C_{ijkl}$. This has solution

$$\phi = \alpha_B + (\phi_0 - \alpha_B) \times \exp \left(\frac{\alpha_B - 1}{K} P_f - \epsilon_{ii}^{\text{total}} + \alpha_T T \right) , \quad (3.21)$$

¹Oelkers 1996: Reviews in Mineralogy v. 34, p. 131-192

²This is a generalisation of Eqn(18) in Y Chen, C Zhou and L Jing “Modeling coupled THM processes of geological porous media with multiphase flow: Theory and vldation against laboratory and field scale experiments” Computers and Geotechnics 36 (2009) 1308-1329

where ϕ_0 is the porosity at zero porepressure, zero elastic strain and zero temperature. Note this porosity can become negative, and an option for ensuring positivity is detailed below.

The evolution of porosity is motivated further in the appendix Chapter C.

Without porepressure effects, the correct expression for porosity as a function of volumetric strain and temperature is

$$\phi = 1 + (\phi_0 - 1) \times \exp(-\epsilon_{ii}^{\text{total}} + \alpha_T T) . \quad (3.22)$$

These expressions may be modified to include the effects of plasticity.

The exponential expressions Eqn (3.21) and Eqn (3.22) can yield negative porosity values, which are unphysical. To ensure positivity of ϕ , PorousFlow offers the following option. First write both equations in the form

$$\phi = A + (B - A) \exp(x) . \quad (3.23)$$

In deriving specific forms for A , B and x , above, it has been assumed that x is small. Since physically $1 \geq A > B > 0$, for $x < 0$ the porosity will be physically meaningful, but for $x \gg 0$, ϕ can become negative. For example, for positive volumetric strain ($x < 0$) the porosity is always physically meaningful. However, when the porous material is squashed with negative and large volumetric strain ($x \gg 0$) the porosity can become negative.

Define

$$C = \log(A/(A - B)) . \quad (3.24)$$

Then, for $x > 0$, the above expression for ϕ is replaced by

$$\phi = A + (B - A) \exp(C(1 - \exp(-x/C))) \geq 0 . \quad (3.25)$$

At first glance this expression may appear rather obscure. It has been constructed to satisfy the following requirements:

1. As $x \rightarrow 0$, $\phi \rightarrow B$, which is necessary for continuity at $x = 0$.
2. As $x \rightarrow \infty$, $\phi \rightarrow 0$, which is physically desirable.
3. The expression is monotonically decreasing, which is both physically desirable and computationally desirable (otherwise there may be non-unique solutions in a PorousFlow simulation).
4. Finally, at $x = 0$, its derivative is $\phi' = B - A$, as desired from continuity of the derivative at $x = 0$.

3.10 Relative permeability relationships

The relative permeability of a phase is a function of its effective saturation:

$$S_{\text{eff}}(S) = \frac{S - S_{\text{res}}^{\beta}}{1 - \sum_{\beta'} S_{\text{res}}^{\beta'}} \quad (3.26)$$

In this equation S_{res}^{β} is the residual saturation for phase β . If $S_{\text{eff}} < 0$ then the relative permeability is zero, while if $S_{\text{eff}} > 1$ then the relative permeability is unity. Otherwise, the relative permeability is given by the expressions below.

3.10.1 Constant

$$k_r = C . \quad (3.27)$$

This is not recommended because there is nothing to discourage phase disappearance, which manifests itself in poor convergence. Usually $k_r(S) \rightarrow 0$ as $S \rightarrow 0$ is a much better choice.

3.10.2 Corey

$$k_r = S_{\text{eff}}^n . \quad (3.28)$$

where n is a user-defined quantity.

3.10.3 Broadbridge-White

$$k_r = K_n + \frac{K_s - K_n}{(c - 1)(c - S_{\text{eff}})} S_{\text{eff}}^2 . \quad (3.29)$$

3.10.4 van Genuchten and cut van Genuchten

$$k_r = \sqrt{S_{\text{eff}}} \left(1 - (1 - S_{\text{eff}}^{1/m})^m \right)^2 . \quad (3.30)$$

This has the numerical disadvantage that its derivative as $S_{\text{eff}} \rightarrow 1$ is crappy. This means that simulations where the saturation oscillates around $S_{\text{eff}} = 1$ do not converge well. Therefore, a “cut” version of the van-Genuchten expression is also offered, which is almost definitely indistinguishable experimentally from the original expression:

$$k_r = \begin{cases} \text{van Genuchten} & \text{for } S_{\text{eff}} < S_c \\ \text{cubic} & \text{for } S_{\text{eff}} \geq S_c \end{cases} . \quad (3.31)$$

Here the cubic is chosen so that its value and derivative match the van Genuchten expression at $S = S_c$, and so that it is unity at $S_{\text{eff}} = 1$.

3.10.5 FLAC

$$k_r = (n + 1)S_{\text{eff}}^n - nS_{\text{eff}}^{n+1} . \quad (3.32)$$

This has the distinct advantage over the Corey formulation that its derivative is continuous at $S_{\text{eff}} = 1$.

3.11 Thermal conductivity

Currently the Porous Flow module implements a simple approach to the functional dependence of thermal conductivity on the rock and fluid-phase conductivities. A “wet” thermal conductivity, λ_{wet} , and a “dry” thermal conductivity λ_{dry} , are defined (both of which are tensors), and

$$\lambda = \lambda_{\text{dry}} + S_{\text{aqueous}}^n (\lambda_{\text{wet}} - \lambda_{\text{dry}}) \quad (3.33)$$

where S_{aqueous} is the aqueous phase saturation, and n is a user-defined exponent.

The dependence of fluid thermal conductivity on temperature may be handled within the fluid property modules (to accommodate the kind of large changes that can occur when water flashes to steam). The solid thermal conductivities also have a slight dependence on pressure, but it is sometimes difficult to obtain reliable data for these dependencies.

3.12 Viscosity

Viscosity may be taken as constant, or the high-precision equations of state may be used to define it.

Chapter 4

Implementation details and numerical issues

4.1 Independent (nonlinear) variables

The Porous Flow module was designed from the ground up to be able to employ variable switching, or persistent variables, instead of the more common pressure-temperature-displacement set of variables. Most simulations in the test-suite use these common set of variables, but it is up to the user to specify which variables are independent, and which variables are dependent (usually the dependent variables will be MOOSE AuxVariables). A good choice of variables can lead to much improved convergence.

4.2 Lumping and upwinding

The Porous Flow module employs fluid-mass and heat-energy lumping to the nodes, as well as full upwinding of the advective flow terms. This ensures far superior numerical convergence, especially in situations where mass-fractions of phases are close to disappearing.

4.2.1 Lumping

Consider here just the fluid-flow equation, as the heat-energy equation is analogous. The time-derivative term in Eqn (1.1) is discretised as

$$\psi \frac{M^\kappa - M_{\text{old}}^\kappa}{dt} . \quad (4.1)$$

Here ψ is the test function that we are integrating against. An alternative discretisation would be $\psi \phi(S\rho)' \dot{P}$ (in the single-phase situation), but Eqn (4.1) conserves mass more effectively than

other alternatives.

In the standard finite-element scheme, M^κ and its individual parts (porosity, saturation, etc) are evaluated at each quadrature point. However, in PorousFlow, everything in the time derivative is evaluated at the nodes. Specifically, M^κ at a node depends only on the independent variables at that node. It has been shown in many papers that this lumping is advantageous for mass conservation and reduces spurious oscillations of the pressure around sharp fronts [24].

The cause of oscillations around sharp fronts, and how mass lumping removes the oscillations, can be illustrated through a simple example.

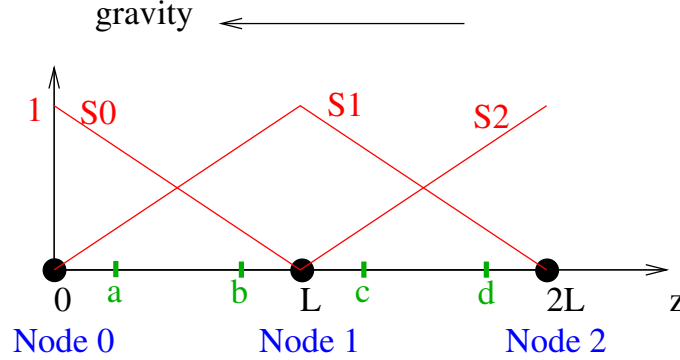


Figure 4.1: Two elements of length L . Linear Lagrange shape/test functions for each node are shown in red (S_0 for node 0, S_1 for node 1, and S_2 for node 2). Gravity acts in the direction $-z$. Gauss points are shown in green.

Consider the situation in Figure 4.1, and suppose that Node 2 has high potential, and that nodes 0 and 1 are at residual saturation where the relative permeability is zero. Then fluid will flow from node 2 to node 1 (and then to node 0 in the next time step). For simplicity, imagine that the fluid mass, M , is a linear function of the potential. Then, up to constants, the discretised mass-conservation equation without mass lumping reads

$$\begin{pmatrix} 2 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} \dot{M}_0 \\ \dot{M}_1 \\ \dot{M}_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix} \quad (4.2)$$

The matrix on the LHS comes from performing the numerical integration of M over the two elements. Note that it is not diagonal because the integration over an element depends on the mass at both of its two nodes. The RHS encodes that no fluid is flowing between nodes 0 and 1, but fluid is flowing from node 2 to node 1.

The important point is the solution of these sets of equations is

$$\dot{M}_0 < 0. \quad (4.3)$$

This means the finite element solution of the mass-conservation equation will be oscillatory around fronts.

However, with mass lumping, the matrix in the above equation becomes diagonal, and the solution is $\dot{M}_0 = 0$. Explicitly, the contribution to the residual at node a is

$$\sum_{\text{qp}} \psi_a^{\text{qp}} \frac{M_a^\kappa - M_{a, \text{old}}^\kappa}{dt} . \quad (4.4)$$

where M is evaluated at node a using the independent (Nonlinear) variables evaluated at that node, and qp are the quadpoints.

There is one small complication. Porosity may be dependent on volumetric strain, which is dependent on the gradients of the displacement variables, which are evaluated at the quadpoints, not the nodes. In this case, the porosity at node a is assumed to be dependent on the volumetric strain evaluated at the closest quadpoint to the node.

4.2.2 Upwinding

Upwinding is necessary¹ in scenarios with nonlinear advection, such as the physics modelled by PorousFlow. For multi-phase situations many upwinding schemes can lead to disaster as the algorithm attempts to withdraw fluid from a node where there is no fluid. I believe that for situations where one phase disappears, or almost disappears, full upwinding is advisable, and hence PorousFlow always employs full upwinding. It has the numerical disadvantage that it is not smooth (in contrast to the SUPG upwinding scheme², for instance), so that close to steady-state where the fluxes are zero, the upwind direction can oscillate, leading to nonconvergence, however this is dealt with by placing a cutoff on the upwinding in PorousFlow. The remainder of this section describes full upwinding for the single-phase unsaturated situation. The multi-phase, multi-component scenario, and the advective term in the heat-flow equation are analogous.

The weak form of the Darcy flux of Eqn (1.6) for a single element is

$$R_a = \int_{\text{element}} \nabla_i \psi_a \frac{\rho k_{ij} k_r}{\mu} (\nabla_j P - \rho g_j) . \quad (4.5)$$

Here ψ_a is the test function that we are integrating against, and R_a is the contribution to the residual for this test function. Define

$$m = \frac{\rho k_r}{\mu} , \quad (4.6)$$

¹Eg, see PS Huyakorn and GF Pinder “A new finite element technique for the solution of two-phase flow through porous media” *Advances in Water Resources* 1 (1978) 285–298. See also V Dalen “Simplified finite-element models for reservoir flow problems” *SPEJ* (Oct 1979) 333–343. See also R Helmig and R Huber “Comparison of Galerkin-type discretization techniques for two-phase flow in heterogeneous porous media” *Advances in Water Resources* 21 (1998) 697–711.

²AN Brooks and TJR Hughes “Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations” *Computer Methods in Applied Mechanics and Engineering* 32 (1982) 199–259. TJR Hughes and M Mallet “A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective-diffusive systems” *Computer Methods in Applied Mechanics and Engineering* 58 (1986) 305–328. TJR Hughes, M Mallet and A Mizukami “A new finite element formulation for computational fluid dynamics: II. Beyond SUPG” *Computer Methods in Applied Mechanics and Engineering* 54 (1986) 341–355

which is traditionally called the mobility.

Upwinding is all a matter of choosing where in the element to evaluate m in the above integral. The sophisticated SUPG approach was designed to weight m on the upstream side of the element. Consider for a moment taking m outside the integral:

$$\tilde{R}_a = m \int_{\text{element}} \nabla_i \psi_a \kappa_{ij} (\nabla_j P - \rho g_j) = m I_a . \quad (4.7)$$

\tilde{R} is not exactly R , but note:

- the original R_a is the mass flux flowing out of node a ;
- so I_a is thereby interpreted as a measure of fluid flow out of node a .

This leads to the following definition of upwinding:

$$R_a \equiv m_a I_a \quad \text{if} \quad I_a \geq 0 . \quad (4.8)$$

The nodes for which $I_a \geq 0$ are called “upwind nodes”, since fluid is flowing from them to the “downwind nodes”. I think this approach was first introduced by Dalen [25] in 1979.

The residual at the downwind nodes is determined by conserving mass. Specifically, let

$$I_{\text{up}} = \sum_{I_a \geq 0} I_a \quad \text{and} \quad I_{\text{down}} = - \sum_{I_a < 0} I_a . \quad (4.9)$$

Then

$$R_a = I_a \frac{I_{\text{up}}}{I_{\text{down}}} \quad \text{for} \quad I_a < 0 . \quad (4.10)$$

Then $\sum_a R_a = 0$ as required by mass conservation within the element (which originates from $\sum_a \psi_a = 1$).

The fully-upwind method is extremely advantageous to use if fluid saturations ever approach residual saturation (where $\kappa_{\text{rel}} = 0$) or zero density, for then the mobility is zero and it becomes impossible to withdraw fluid from such a node (in practice this may still happen due to precision loss or other related nasty artifacts that I will not describe here).

Prescribed sinks, either from the boundary or from internal objects such as wellbores, are also fully-upwinded in PorousFlow since they also potentially suffer from phase-disappearance problems.

4.3 Preconditioners and linear solvers

MOOSE allows users to utilise the full power of the PETSc preconditioners and linear solvers. The following choices have been found to be effective for various types of PorousFlow simulations.

- `pc_type = bjacobi, ksp_type = bcgs.`
- `pc_type = bjacobi, ksp_type = gmres.`
- `pc_type = asm, pc_asm_overlap = 2, sub_pc_type = lu, sub_pc_factor_shift_type = NONZERO, ksp_type = gmres.`
- `pc_type = asm, pc_asm_overlap = 2, sub_pc_type = lu, sub_pc_factor_shift_type = NONZERO, ksp_type = gmres` along with the following options `-ksp_diagonal_scale -ksp_diagonal_scale_fix -ksp_gmres_modifiedgramschmidt.`

4.4 Setting the global convergence criterion, and scaling the variables

It is very important to set the global convergence criterion appropriately. This is the `nl_abs_tol`, or equivalently, PETSc’s `-snes_atol`. Of course the arguments below will also inform the setting of the relative tolerance `nl_rel_tol` (or PETSc’s `-snes_rtol`). If set too small, then MOOSE will appear not to converge and will need very small timesteps in order to “do anything”. If set too large, then MOOSE will converge to the “wrong” result. Here are some tips to estimate the global convergence criterion.

4.4.1 Fluids

1. Determine an appropriate tolerance on what you mean by “steadystate”. For instance, in a single-phase simulation with reasonably large constant fluid bulk modulus, and gravity acting in the $-z$ direction, the steadystate solution is $P = -\rho_0 g z$ (up to a constant). In the case of water, this reads $P = -10000z$ (approximately).

Instead of this, suppose you would be happy to say the model is at steadystate if $P = -(\rho_0 g + \epsilon)z$. Here ϵ needn’t be constant: it may switch in sign, but its magnitude $|\epsilon|$ is roughly the same over the domain of interest. For instance, for water, an error of $|\epsilon| = 1 \text{ Pa.m}^{-1}$ might be suitable in your problem.

Then recall that the residual is just

$$R = \left| \int \nabla_i \left(\frac{\kappa_{ij} \kappa_{\text{rel}} \rho}{\mu} (\nabla_j P + \rho g_j) \right) \right| \quad (4.11)$$

Evaluate this for your “almost-steadystate” solution.

For instance, in the case of water just quoted, $\nabla P + \rho g \sim |\epsilon|$, and the $\rho \sim \rho_0$, and the integral just becomes V , the volume of the region of interest, so

$$R \approx V |\kappa| \rho_0 / \mu \epsilon . \quad (4.12)$$

For water, this gives $R \approx V |\kappa| \times 10^6 \epsilon$, using standard values for ρ_0 and μ .

2. In the previous step, an appropriate tolerance on the residual was given as $V|\kappa|\rho_0\epsilon/\mu$. Now the volume of interest, V , must be specified. Often this is not the entire mesh, but a small region where most of the interesting dynamics occurs, and the remainder of the mesh exists just to provide reasonable boundary conditions for this “interesting” region. The residual in the “boring” region can be thought of as virtually zero, while the residual in the “interesting” region is $V_{\text{interesting}}|\kappa|\rho_0\epsilon/\mu$. This is the absolute nonlinear residual that you should aim for.
3. In the previous steps, I’ve implicitly assumed κ is constant, ρ is virtually constant at ρ_0 , only a single-phase situation, etc. In many cases these assumptions are not valid, so the integral of Eqn (4.11) cannot be done as trivially as in the previous steps. In these cases, I simply suggest to build a model with initial conditions like $P = -(\rho_0 g + \epsilon)z$, and just see what the initial residual is. That will give you an idea of how big a reasonable residual tolerance should be.

4.4.2 Heat

A similar method can be used to estimate the convergence criterion for heat-flow problems. Suppose there is both heat conduction, and fluid advection. Suppose that ϵ_{fluid} is the acceptable error in the fluid equations (as in the previous section). Suppose that $\nabla T \approx \epsilon_{\text{heat}}$ is an acceptable error for the temperature. For instance being incorrect by $\epsilon = 10^{-3} \text{ K.m}^{-1}$ might be appropriate for your problem.

Then the residual is approximately

$$R \approx V(\lambda\epsilon_{\text{heat}} + h|\kappa|\rho_0\epsilon_{\text{fluid}}/\mu) , \quad (4.13)$$

The last term is just the fluid enthalpy, h , multiplied by the fluid residual.

4.4.3 Mechanics

Very similar methods can be used in simulations where mechanical deformations are active. Roughly speaking, MOOSE is attempting to set $\nabla\sigma^{\text{tot}} = 0$. Determine the error in $\nabla\sigma$ that you are willing to accept, and label it $|\epsilon|$. For instance $\epsilon = 1 \text{ Pa.m}^{-1}$ might be appropriate for your problem. You may determine ϵ from consideration of stresses directly, or you may wish to consider what the acceptable error in strains or displacements would be, and then use the elasticity tensor to find ϵ .

The nonlinear residual will be

$$R \approx V|\epsilon| , \quad (4.14)$$

where V is the volume of interest.

4.4.4 Scaling the variables

Often it is appropriate to scale the variables in order to weight their contributions to the overall nonlinear residual appropriately. For instance, suppose the previous arguments provided

$$\begin{aligned} R_{\text{fluid}} &\sim 10^{-9}V, \\ R_{\text{heat}} &\sim (10^{-3} + 10^6 \times 10^{-9})V, \\ R_{\text{mech}} &\sim V, \end{aligned} \tag{4.15}$$

(with the same V in each case). Then, a scaling of around 10^9 on the porepressure variable (or whatever MOOSE variable is associated to the fluid equation) would be appropriate. Similarly, a scaling of around 10^3 on the temperature variable would be appropriate.

4.5 Fully saturated formulations

In fully-saturated, single-phase simulations the upwinding described in Section 4.2.2 is typically unnecessary. Moreover, the standard PorousFlow Kernels are somewhat inefficient in the fully-saturated case since Material Properties such as relative permeabilities and saturations actually do not need to be computed. PorousFlow offers some Kernels that may be used in this situation, as described below. They also have some minor disadvantages that are described below.

In fully-saturated, single-phase, single-component simulations, the mass lumping described in Section 4.2.1 is typically unnecessary. More importantly, in many real-life situations, the time derivative of the fluid mass may be linearised, which leads to improved convergence. PorousFlow offers a Kernel that may be used in this situation, as described below. It too has some disadvantages and other unique features.

4.5.1 Non-upwinded flux Kernels for fully-saturated, single-phase simulations

The following Kernels are available.

1. `PorousFlowFullySaturatedDarcyBase` models $\nabla_i ((\rho)k_{ij}(\nabla_j P - \rho g_j) / \mu)$. This is fully-saturated, single-phase, single-component fluid flow. No upwinding is performed. The initial multiplication by ρ is optional (indicated by the parentheses): the reason for this is described in Section 4.5.2.
2. `PorousFlowFullySaturatedDarcyFlow` models $\nabla_i (\chi^\kappa \rho k_{ij}(\nabla_j P - \rho g_j) / \mu)$. This is fully-saturated, single-phase, multi-component fluid flow, for fluid component κ . No upwinding is performed.

3. **PorousFlowFullySaturatedHeatAdvection** models $\nabla_i (h\rho k_{ij}(\nabla_j P - \rho g_j)) / \mu$. This is heat advection via fully-saturated, single-phase, multi-component fluid flow. No upwinding is performed.

Note that no Kernel includes the relative permeability, for it is always unity. This means that a RelativePermeability Material is not needed, which saves in computation time. Similarly, the saturation does not need to be computed, and various Nodal Materials (such as the density evaluated at the nodes) do not need to be computed. This makes input files smaller, and saves on computation time. In a full-scale simulation, we have recorded computational saving of about 4%.

Possibly the greatest advantage in using these Kernels is the reduced diffusion of fronts of temperature and fluid species. The full upwinding used in the remainder of PorousFlow adds diffusion compared with the non upwinded case. These effects are quantified by example in the documentation for the heat advection tests, and the documentation for sinks where the transport of a fluid component is explored. An example is shown in Figure 4.2.

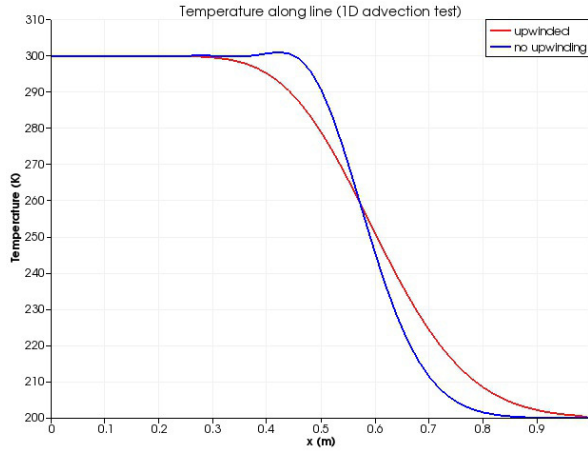


Figure 4.2: Heat advection via a fluid in 1D. The fluid flows with a constant velocity from the left to the right. The initial temperature distribution is a uniform 200 K everywhere. Then the left-hand side is raised to 300 K and fixed at that temperature. This causes a front of 300 K to move right. Without any diffusion a sharp front would advect from left to right. The upwinding adds diffusion, while no upwinding adds less diffusion.

The disadvantage of using the FullySaturated Kernels is that the physics is no longer completely controlled. The whole point of full upwinding is to ensure that no unphysical behaviour manifests itself, such as negative saturations or negative mass fractions. It is clear from Figure 4.2 that the temperature has risen slightly above 300 K ahead of the front, which is unphysical.

4.5.2 Non-lumped time derivative of fluid mass for fully-saturated, single-phase, single-component simulations

Consider a fully-saturated, single-phase, single-component fluid in a THM simulation. The time-derivative terms from Eqn (1.1) are

$$\frac{\partial}{\partial t} \phi \rho + \phi \rho \dot{\epsilon}_v , \quad (4.16)$$

where $\dot{\epsilon}_v = \nabla \cdot \mathbf{v}_s$. Using the THM evolution of porosity given by Eqn (3.20), along with the assumption that the fluid bulk modulus, K_f , and its volumetric thermal expansion coefficient, α_f , are constant with fluid density given by

$$\rho = \rho_0 \exp(P/K_f - \alpha_f T) , \quad (4.17)$$

the time derivative terms may be written as

$$\rho \left(\frac{1}{M} \dot{P} + \alpha_B \dot{\epsilon}_v - A \dot{T} \right) . \quad (4.18)$$

Here M is the so-called Biot Modulus:

$$\frac{1}{M} = \frac{\phi}{K_f} + \frac{(1 - \alpha_B)(\alpha_B - \phi)}{K} , \quad (4.19)$$

and A is an effective volumetric thermal expansion coefficient:

$$A = (\alpha_B - \phi) \alpha_T + \phi \alpha_f . \quad (4.20)$$

Notice that disregarding the premultiplication by ρ , the above time-derivative terms would be linear in the variables P , displacement, and T , if M and A were constant.

In standard poro-mechanics it is usual to calculate M and A at the initial stage of simulation, and keep them fixed forever afterwards. Of course this is an approximation since M and A were derived using the explicit assumption of a porosity that depended on P , ϵ_v , and T , but it makes finding analytical solutions much easier. Therefore, PorousFlow offers the following Materials and Kernels. Using these Materials and Kernel allows immediate and precise comparison with analytical and numerical solutions of poro-mechanics.

1. **PorousFlowConstantBiotModulus** Material, which computes M given by Eqn (4.19) during the initial stage of simulation, and keeps it fixed thereafter. It requires a Porosity Material, but that Material's Property is only used during the initial computation.
2. **PorousFlowConstantThermalExpansionCoefficient** Material, which computes A given by Eqn (4.20) during the the initial stage of simulation, and keeps it fixed thereafter. It requires a Porosity Material, but that Material's Property is only used during the initial computation.
3. **PorousFlowFullySaturatedMassTimeDerivative** Kernel, which computes one of the following contributions, depending upon a user-defined flag:

- (a) $(\rho) \dot{P}/M$ for fluid-flow-only problems;

- (b) $(\rho)(\dot{P}/M - A\dot{T})$ for TH problems;
- (c) $(\rho)(\dot{P}/M + \alpha_B \dot{\epsilon}_v)$ for HM problems;
- (d) $(\rho)(\dot{P}/M + \alpha_B \dot{\epsilon}_v - A\dot{T})$ for THM problems.

None of these employ lumping, which is largely unnecessary in this single-phase, single-component situation. This means only “quad-point” Materials are needed. In fact, when using the FullySaturated flow Kernels described in Sec 4.5.1, *only* standard Materials evaluated at the quadpoints are needed, which saves on computation time and input-file length.

In each case, the initial pre-multiplication by ρ is optional (indicated by the parentheses around ρ) as described in the next paragraph.

When the Kernel is pre-multiplied by ρ , which is the default, it is computing the time derivative of fluid mass. This allows the Kernel to be easily used with the remainder of PorousFlow: the BCs, the Postprocessors, the AuxKernels, and the DiracKernels are all based on mass.

When the pre-multiplication is not performed, they are computing the time derivative of fluid volume. This has two great advantages:

1. the time-derivatives are linearised, resulting in excellent nonlinear convergence;
2. comparing with results from poro-mechanics theory is straightforward.

However, this means additional care must be taken.

1. The flow Kernel `PorousFlowFullySaturatedDarcyBase` should also not pre-multiply by density.
2. The BCs, Postprocessors, AuxKernels, and DiracKernels must be written in such a way to operate in the fluid-volume scenario rather than the default fluid-mass scenario.
3. The flag `consistent_with_displaced_mesh` should be set to `false` in the `PorousFlowVolumetricStrainMaterial`.

Appendix A

The continuity equation

The derivation of the continuity equation is fundamental to the fluid and heat flow DEs, and is included in this section. The notation in this appendix is different from that in the main report.

A.1 Eulerian coordinates

Introduce the notion of the “spatial coordinate frame”. It is the coordinate frame of a stationary observer who is looking at the deforming porous solid from the outside. Denote its coordinates by \mathbf{x} , and the derivatives with respect those coordinates by ∇ .

Let Ω be a volume that is attached to particles of the porous-solid skeleton. As the porous solid deforms, so too will Ω . Denote the velocity of the porous solid is \mathbf{v}_s , measured in the spatial coordinate frame: $\mathbf{v}_s = \mathbf{v}_s(x, t)$. Then the change of a small volume element $d\Omega$ is computed by calculating the Jacobian, and it is

$$\frac{d}{dt}(d\Omega) = \nabla \cdot \mathbf{v}_s d\Omega . \quad (\text{A.1})$$

Remember the derivative ∇ is differentiating with respect to the coordinates of the spatial coordinate frame. This formula is easy to motivate because $\nabla \cdot \mathbf{v}_s = \dot{\epsilon}_{ii}$ which is the time derivative of the volumetric strain.

Let M represent a quantity that is attached to the porous-solid skeleton, for instance the mass density of the solid. Express M in the spatial coordinate frame: $M = M(x, t)$. As the porous solid deforms

$$\frac{d}{dt}M = \frac{\partial}{\partial t}M + \mathbf{v}_s \cdot \nabla M . \quad (\text{A.2})$$

The second term is easy to motivate if you consider a constant velocity with the spatially-dependent but temporally-constant M .

The continuity equation is

$$\frac{d}{dt} \int_{\Omega} M d\Omega + \int_{\partial\Omega} \mathbf{F} \cdot \mathbf{n} dA = 0 , \quad (\text{A.3})$$

where \mathbf{F} is the flux of M out of Ω , n is the outward unit normal, and dA is the area element on $\partial\Omega$ (which is the surface of Ω). Using the above expressions, and the divergence theorem, the continuity equation reads

$$\int_{\Omega} \left(\frac{\partial}{\partial t} M + \nabla \cdot (M \mathbf{v}_s) + \nabla \cdot \mathbf{F} \right) d\Omega = 0 . \quad (\text{A.4})$$

Specialising M and \mathbf{F} to fluids and heat gives the equations mentioned in the main text.

A.2 Lagrangian coordinates

Introduce the notion of the “material coordinate frame”. It is the coordinate frame of an observer who is fixed to a certain point in the porous solid (eg, a particular finite-element node). Denote the coordinates in this frame by X . This is the frame used throughout this report. Fluid properties (pressures, mass fractions), the temperature, etc, are all stored at the finite-element nodes or the quadpoints, and move with the mesh. At the very least, an Eulerian description would be inconvenient when visualising with paraview.

Introduce the material derivative D/Dt . If M is any property that is expressed in terms of the “spatial coordinate frame” (Eulerian coordinates): $M = \tilde{M}(x, t)$, then the material derivative is

$$\frac{D}{Dt} \tilde{M}(x, t) = \frac{\partial}{\partial t} \tilde{M}(x, t) + \mathbf{v}_s \cdot \nabla \tilde{M}(x, t) . \quad (\text{A.5})$$

The continuity Eqn (A.4) can be re-written as

$$0 = \frac{D}{Dt} M + M \nabla \cdot \mathbf{v}_s . \quad (\text{A.6})$$

However, if M is expressed in terms of the Lagrangian coordinates:

$$M = M(X, t) , \quad (\text{A.7})$$

(generally $M(X, t)$ will have a different functional form than $\tilde{M}(x, t)$, thus the tilde to emphasise the difference) then the material derivative is expressed by

$$\frac{D}{Dt} M(X, t) = \frac{\partial}{\partial t} M(X, t) , \quad (\text{A.8})$$

but the continuity equation has the identical form:

$$0 = \frac{D}{Dt} M + M \nabla \cdot \mathbf{v}_s . \quad (\text{A.9})$$

Appendix B

Benchmark studies to perform

B.1 Use case: Two-component, two-phase, nonisothermal

This solution will benchmark the two-phase immiscible flow and thermal equations in of the code against the semi-analytical solutions in LaForce et al (AWR, 73 (2014a) 227 – 241). (Note: These will have to be written because they are copy/pasted from manuscript)

The analytical solutions assume that pressure changes in the reservoir are small, so that fluid volume, density, and viscosity are all constant. Fluids and the solid are treated as incompressible. It is also assumed that changes in temperature do not impact any of the solid or fluid properties. With these assumptions the thermal and saturation equations become

$$[S_1 + \beta] \frac{\partial T_{D1}}{\partial t_D} + [f_1 + \alpha] \frac{\partial D1}{\partial r_D} = k_{Tt} \gamma_j \frac{\partial^2 D1}{\partial z_D^2} \quad (\text{B.1})$$

$$\frac{\partial S_1}{\partial t_D} + \frac{\partial f_1}{\partial r_D} = 0 \quad (\text{B.2})$$

$$\alpha = \frac{\rho_2 C_2}{(\rho_1 C_1 - \rho_2 C_2)} \quad (\text{B.3})$$

$$\beta = \frac{\rho_2 C_2 + \frac{1-\phi_r}{\phi_r} \rho_s C_s}{(\rho_1 C_1 - \rho_2 C_2)} \quad (\text{B.4})$$

where S_1 is the saturation of the water phase, f_1 is the fractional flow of the water (resident) phase, T_{D1} is temperature and r_D is radial distance. The C_j are the mass-based heat capacity of fluid 1, 2, or the formation rock. The parameters α and β in Eqns B.3—(B.4) are constant throughout the displacement. Dimensionless variables are defined in LaForce et al (2014a).

The relative permeability of the phases are from one of the TOUGH2 models

$$\begin{aligned} k_{r1} &= \hat{S}^4 \\ k_{r2} &= \left(1 - \hat{S}\right)^2 \left(1 - \hat{S}^2\right) \\ \hat{S} &= \frac{S_1 - S_{1r}}{1 - S_{1r} - S_{2r}} \end{aligned} \tag{B.5}$$

In the analytical model, heat loss from the reservoir to the over/under burden is handled via a heat sink term that approximates the diffusive process.

$$k_{Tt} \frac{\partial^2 T_D}{\partial z^2} \rightarrow -UT_D. \tag{B.6}$$

Note: It may be useful to solve these equations with $U = 0$ initially.

In this formulation the thermal properties of the reservoir and the adjacent formations may be different, but the over- and under-burden must have the same thermal properties for vertical symmetry. It will be interesting to test the simulator on the full Eqn (B.1) with vertical heat diffusion vs a simulated solution with the heat sink.

B.2 Two-component, two-phase, nonisothermal, forward coupled stress

This solution will benchmark the code against the semi-analytical solutions in LaForce et al (AWR, 73 (2014) 242 – 253), which takes the saturation and temperature solution from above and also calculates pressure and stress in the reservoir. The semi-analytical solutions are strictly forward-coupled, so that pressure does not impact the saturation or temperature profiles, and stress has no impact on pressure, saturation or temperature. In that work, pressure and stress are calculated at a given snapshot in time, t_D^* . This gives pressure and stress equations with no time dependence:

$$\frac{\partial P}{\partial r_D} = \frac{q}{4\pi h k r_D} \frac{1}{M} \tag{B.7}$$

where M is the mobility of the overall fluid composition at r_D .

The displacement u , of the porous medium at a fixed time is given by **Note: Notation may not be consistent with the above!!**:

$$\frac{d}{d\xi} \left(\frac{1}{\xi} \frac{d}{d\xi} (\xi u_D) \right) = \frac{\partial T_{D2}}{\partial \xi} + \frac{(1 - 2\nu)}{(1 - \nu)} \frac{\partial P_D}{\partial \xi} \quad (\text{B.8})$$

where $\xi = \frac{r}{r_w}$ for well radius r_w , and ν is the drained Poisson ratio. All other dimensionless groups given in LaForce et al (2014b).

In these equations stresses are defined as positive if they are tensile and negative if they are compressive. The stress at the wall of the well will have equal magnitude as the fluid pressure, but opposite sign so that at the wellbore the effective radial stress is always zero. These equations can be solved analytically on a finite reservoir domain for either zero stress or zero displacement outer boundary conditions.

Appendix C

Evolution of porosity for an isothermal situation

The evolution of porosity is fundamental to the coupling between flow and solid mechanics. Consider the isothermal situation with no plasticity.

Denote the change of a quantity, q , by Δq . Recall that the porosity is defined by $\phi = V_f/V$, where V is an arbitrary volume of the porous material, and V_f is the porevolume within that volume. Also, by definition of the effective stress,

$$\Delta\epsilon_{ij} = C_{ijkl}(\Delta\sigma_{ij}^{\text{tot}} + \alpha_B\delta_{ij}\Delta P_f) . \quad (\text{C.1})$$

Taking the trace of this equation, and using $V^{-1}\Delta V = \Delta\epsilon_{ii}$ yields

$$\frac{\Delta V}{V} = \delta_{ij}C_{ijkl}(\Delta\sigma_{ij}^{\text{tot}} + \alpha_B\delta_{ij}\Delta P_f) . \quad (\text{C.2})$$

In most instances it is appropriate to write this equation as

$$\frac{\Delta V}{V} = -\frac{1}{K}(P_{\text{mech}} - \alpha_B\delta_{ij}\Delta P_f) . \quad (\text{C.3})$$

where the total mechanical pressure is

$$P_{\text{mech}} = -\text{Tr}\sigma/3 . \quad (\text{C.4})$$

and K is the so-called “drained” bulk modulus $K = \delta_{ij}C_{ijkl}\delta_{kl}$. To find the evolution equation for porosity, a similar equation for $\Delta V_f/V_f$ must be derived.

Assuming linearity

$$\frac{\Delta V_f}{V_f} = A_{ij}(\Delta\sigma_{ij}^{\text{tot}} + B\delta_{ij}\Delta P_f) . \quad (\text{C.5})$$

The Betti-Maxwell reciprocal theorem yields A_{ij} and B , as is now shown.

The work increment is

$$dW = -P_{\text{mech}}dV + P_fdV_f , \quad (\text{C.6})$$

So during some deformation that takes P_{mech} from P_{mech}^i to P_{mech}^f , and P_f from P_f^i to P_f^f , the total work is

$$\begin{aligned} W &= - \int P_{\text{mech}} dV + \int P_f dV_f , \\ &= \frac{V}{K} \int_{P_{\text{mech}}^i}^{P_{\text{mech}}^f} P_{\text{mech}} dP_{\text{mech}} - \frac{V\alpha_B}{K} \int_{P_f^i}^{P_f^f} P_{\text{mech}} dP_f + V_f A_{ii} \int_{P_{\text{mech}}^i}^{P_{\text{mech}}^f} P_f dP_{\text{mech}} + V_f A_{ii} B \int_{P_f^i}^{P_f^f} P_{\text{mech}} dP_f \end{aligned} \quad (\text{C.7})$$

Now consider two experiments:

1. First take P_{mech} from 0 to P_{mech} with P_f fixed at 0. Then, leaving P_{mech} fixed, take P_f from 0 to P_f . The first takes work $V P_{\text{mech}}^2 / (2K)$, while the second takes work $-\alpha_B V P_{\text{mech}} P_f / K + V_f A_{ii} B P_f^2 / 2$.
2. First take P_f from 0 to P_f with P_{mech} fixed at 0. Then, leaving P_f fixed, take P_{mech} from 0 to P_{mech} . The first takes work $V_f A_{ii} B P_f^2 / 2$, and the second takes work $V P_{\text{mech}}^2 / (2K) + V_f P_{\text{mech}} A_{ii} P_f$.

The two experiments must give the same work done (this is called the Betti-Maxwell reciprocal theorem), which yields

$$A_{ij} = \alpha_B C_{ijkl} \delta_{kl} / \phi . \quad (\text{C.8})$$

Now to identify B . Consider a so-called ‘‘ideal porous material’’, which is characterised by a fully-connected pore space and a homogeneous and isotropic matrix material. In this case, applying a uniform porepressure, P_f , and an equal mechanical pressure, $P_{\text{mech}} = P_f$, the solid material will experience a uniform pressure throughout its skeleton. This means it will deform uniformly without any shape change, and

$$\frac{\Delta V_f}{V_f} = \frac{\Delta V}{V} . \quad (\text{C.9})$$

Substituting this equation, this specific pressure condition, and Eqn (C.8) into Eqns (C.3) and (C.5), yields

$$B = 1 + \phi - \phi / \alpha_B . \quad (\text{C.10})$$

Now that A_{ij} and B have been identified, they may be substituted into Eqn (C.5). Rearranging yields

$$\frac{\Delta V_f}{V_f} = \frac{\alpha_B}{\phi} \delta_{ij} C_{ijkl} [\Delta \sigma_{kl}^{\text{tot}} + \alpha_B \delta_{kl} \Delta P_f] + \frac{\delta_{ij} \delta_{kl} C_{ijkl}}{\phi} (1 - \alpha_B) (\alpha_B - \phi) \Delta P_f \quad (\text{C.11})$$

Using the expression for $\Delta V/V$ yields

$$\Delta V_f = V \alpha_B \Delta \epsilon_{ii} + V \delta_{ij} \delta_{kl} C_{ijkl} (1 - \alpha_B) (\alpha_B - \phi) \Delta P_f \quad (\text{C.12})$$

Now $\Delta \phi = V^{-1} \Delta V_f - V_f V^{-2} \Delta V$, so using the definition of K yields

$$\frac{\partial \phi}{\partial t} = (\alpha_B - \phi) \frac{\partial \epsilon_{ii}}{\partial t} + \frac{(1 - \alpha_B) (\alpha_B - \phi)}{K} \frac{\partial P_f}{\partial t} , \quad (\text{C.13})$$

as written in Eqn (3.20).

Bibliography

- [1] K. Pruess, C. Oldenburg, and G. Moridis. TOUGH2 User’s Guide, Version 2.0. Technical Report LBNL-43134, Lawrence Berkeley National Laboratory, Berkeley CA, USA, 1999.
- [2] Simon A. Mathias, Jon G. Gluyas, Curtis M. Oldenburg, and Chin-Fu Tsang. Analytical solution for joulethomson cooling during CO₂ geo-sequestration in depleted oil and gas reservoirs. *International Journal of Greenhouse Gas Control*, 4:806–810, 2010.
- [3] D. W. Peaceman. Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability. *SPE J.*, 23:531–543, 1983.
- [4] Z. Chen and Y. Zhang. Well flow models for various numerical methods. *International Journal of Numerical Analysis and Modelling*, 6:375–388, 2009.
- [5] Revised Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam. Technical report, IAPWS, 2007.
- [6] Revised Supplementary Release on Backward Equations for Specific Volume as a Function of Pressure and Temperature $v(p,T)$ for Region 3 of the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam. Technical report, IAPWS, 2014.
- [7] Release on the IAPWS Formulation 2008 for the Viscosity of Ordinary Water Substance. Technical report, IAPWS, 2008.
- [8] Revised Release on the IAPS Formulation 1985 for the Thermal Conductivity of Ordinary Water Substance. Technical report, IAPS, 1985.
- [9] Release on the IAPWS Formulation 2011 for the Thermal Conductivity of Ordinary Water Substance. Technical report, IAPWS, 2011.
- [10] T. Driesner and C. A. Heinrich. The system H₂O-NaCl. Part I: Correlation formulae for phase relations in temperaturepressurecomposition space from 0 to 1000 C, 0 to 5000 bar, and 0 to 1 X_{NaCl}. *Geochimica et Cosmochimica Acta*, 71:4880–4901, 2007.
- [11] T. Driesner. The system H₂O-NaCl. Part II: Correlations for molar volume, enthalpy, and isobaric heat capacity from 0 to 1000 C, 1 to 5000 bar, and 0 to 1 X_{NaCl}. *Geochimica et Cosmochimica Acta*, 71:4902–4919, 2007.
- [12] S. L. Phillips, A. Igbene, J. A. Fair, H. Ozbek, and M. Tavana. A technical databook for geothermal energy utilization. Technical Report LBL-12810, Lawrence Berkeley National Laboratory, Berkeley CA, USA, 1981.

- [13] J. L. Haas Jr. Physical properties of the coexisting phases and thermochemical properties of the H_2O component in boiling NaCl solutions. Technical Report USGS Bulletin 1421-A, United States Geological Survey, 1976.
- [14] R. W. Potter, R. S. Babcock, and D. L. Brown. A new method for determining the solubility of salts in aqueous solutions at elevated temperatures. *J. Res. US Geol. Surv.*, 5:389–395, 1977.
- [15] R. Span and W. Wagner. A new equation of state for carbon dioxide covering the fluid region from the triple-point temperature to 1100 k at pressures up to 800 mpa. *J. Phys. Chem. Ref. Data*, 25:1509–1596, 1996.
- [16] A. Fenghour, W. A. Wakeham, and V. Vesovic. The viscosity of carbon dioxide. *J. Phys. Chem. Ref. Data*, 27:31–44, 1998.
- [17] G. Scalabrin, P. Marchi, F. Finezzo, and R. Span. A reference multiparameter thermal conductivity equation for carbon dioxide with an optimized functional form. *J. Phys. Chem. Ref. Data*, 35:1549–1575, 2006.
- [18] T. F. Irvine Jr and P. E. Liley. *Steam and gas tables with computer equations*. Academic Press Inc., Orlando, 1984.
- [19] M. Th. van Genuchten. A closed for equation for predicting the hydraulic conductivity of unsaturated soils. *Soil Sci. Soc.*, 44:892–898, 1980).
- [20] P Broadbridge and I. White. Constant rate rainfall infiltration: A versatile nonlinear model, 1. analytical solution. *Water Resources Research*, 24:145–154, 1988.
- [21] A. W. Warrick, D. O. Lomen, and A. Islas. An analytical solution to richards’ equation for a draining soil profile. *Water Resources Research*, 26:253–258, 1990.
- [22] C. Rogers, M. P. Stallybrass, and D. L. Clements. On two phase filtration under gravity and with boundary infiltration: application of a backlund transformation. *Nonlinear Analysis, Theory, Methods and Applications*, 7:785–799, 1983.
- [23] R. J. Millington and J. P. Quirk. Permeability of porous solids. *Trans. Faraday Soc.*, 57:1200–1207, 1961.
- [24] E. T. Bouloutas M.A. Celia and R. L. Zabra. A general mass-conservative numerical solution for the unsaturated flow equation. *Water Resources Research*, 26:1483–1496, 1990.
- [25] V. Dalen. Simplified finite-element models for reservoir flow problems. *SPE J.*, 19:333–342, 1979.