

# Groundwater flow, heat and solute transport constructing your own numerical models using Python

Elco Luijendijk  
McGill University, Montreal, Canada  
elco.luijendijk at mcgill.ca

## 1 Escript

Escript is a generic finite element model that solves the following partial differential equation:

$$-\nabla(A\nabla u + Bu) + C\nabla u + Du = -\nabla X + Y \quad (1)$$

escript equation in Einstein notation:

$$-(A_{ij}u_j)_{,i} - (B_i u)_{,i} + C_j u_{,j} + Du = -(X_{,j})_{,j} + Y \quad (2)$$

The Einstein notation for a derivative is:

$$u_{,t} = \frac{\partial u}{\partial t} \quad (3)$$

## 1.1 Groundwater flow

### 1.2 Simplified flow equation, constant density

For a fluid with constant density ( $\rho_f = \text{constant}$ ), fluid flow is driven by changes in hydraulic head ( $h$ ). Recall that hydraulic head is equal to the pressure head and the elevation head:  $h = P/(\rho_f g) + z$ . If  $h$  is constant the fluid is static.

The fluid flow equation for a constant density fluid is:

$$S_{sh} \frac{\partial h}{\partial t} = \nabla(K \nabla h) + W \quad (4)$$

where  $S_s$  is the specific storativity  $m^{-1}$ , which is the volume of fluid released per unit decrease in hydraulic head.  $h$  is the hydraulic head  $[m]$ ,  $t$  is time  $[s]$ ,  $K$  is the hydraulic conductivity  $[ms^{-1}]$  and  $W$  is a source or sink term  $[s^{-1}]$ . The source/sink term  $W$  can be used to represent processes that add fluids to the system, such as groundwater recharge, or processes that take fluid out of the system, such as pumping.

#### 1.2.1 Steady state

Steady-state flow means that the hydraulic head is stable over time, in mathematical terms:  $\frac{\partial h}{\partial t} = 0$ . For steady state conditions the groundwater flow equation reduces to:

$$\nabla K \nabla h + W = 0 \quad (5)$$

The PDE form solved by escript/Finley is:

$$-\nabla(A \nabla u + B u) + C \nabla u + D u = -\nabla X + Y \quad (6)$$

Rewriting the steady-state flow equation gives:

$$-\nabla \underbrace{-K}_{A} \nabla \underbrace{h}_u = \underbrace{-W}_Y \quad (7)$$

And the resulting escript coefficients are:

$$u = h$$

$$A = -K$$

$$Y = -W$$

#### 1.2.2 Explicit formulation (forward in time)

We use forward Euler discretization in time. This means that the hydraulic head at time  $t = t + 1$  is calculated as a function of the hydraulic head at the preceding timestep  $t = t$

$$\frac{\partial h}{\partial t} = \frac{h_{t+1} - h_t}{\Delta t} \quad (8)$$

and taking  $\nabla h = \nabla h_t$  results in:

$$S_{sh} \frac{h_{t+1} - h_t}{\Delta t} = \nabla K \nabla h_t + W \quad (9)$$

Rearranging this in the shape of the escript equation gives:

$$S_{sh} h_{t+1} = \nabla dt K \nabla h_t + dt W + S_{sh} h_t \quad (10)$$

$$-\nabla(A\nabla u + Bu) + C\nabla u + Du = -\nabla X + Y \quad (11)$$

$$\underbrace{S_{sh}}_D \underbrace{h_{t+1}}_u = -\nabla \underbrace{-dtK\nabla h_t}_X + \underbrace{dtW + S_{sh}h_t}_Y \quad (12)$$

The resulting coefficients of the escript PDE are:

$$u = h_{t+1}$$

$$D = S_{sh}$$

$$X = -dt * K * \nabla h_t$$

$$Y = dtW + S_{sh}h_t$$

The remaining coefficients are not used:

$$A = 0, B = 0, C = 0$$

### 1.2.3 Implicit solution (backward in time)

Here we again take  $h_{t+1}$  as the unknown variable. However, this time we rewrite the equation to calculate the hydraulic head as a function of the divergence of the hydraulic head at the same timestep  $\nabla h_{t+1}$ .

$$S_{sh} \frac{h_{t+1} - h_t}{\Delta t} = \nabla K \nabla h_{t+1} + W \quad (13)$$

Rearranging this equations gives:

$$-\nabla dtK \nabla h_{t+1} + S_{sh}h_{t+1} = S_{sh}h_t + dtW \quad (14)$$

and again in the same notation as the escript PDE:

$$-\nabla(A\nabla u + Bu) + C\nabla u + Du = -\nabla X + Y \quad (15)$$

$$-\nabla \underbrace{dtK}_A \nabla \underbrace{h_{t+1}}_u + \underbrace{S_{sh}}_D \underbrace{h_{t+1}}_u = \underbrace{S_{sh}h_t + dtW}_Y \quad (16)$$

$$u = h_{t+1}$$

$$A = dtK$$

$$D = S_{sh}$$

$$Y = S_{sh}h_t + dtW$$

### 1.2.4 Setting boundary conditions

The preceding equations solve hydraulic head in a given model domain. However, the solution always need so-called boundary conditions. The two main boundary conditions are a specified head or a specified flux at the boundary.

A specified flux or Neumann boundary condition is in escript defined as:

$$\vec{n}A\nabla u + \vec{n}Bu + du = \vec{n}\nabla X + y \quad (17)$$

Where  $u$  is again the variable we are trying to solve (hydraulic head).  $\vec{n}$  is a vector normal (perpendicular) to the model boundary.

The flux  $q$  is given by Darcy's law:  $q = K\nabla h$

If we want to define a fixed flux ( $q_b$  [ $ms^{-1}$ ]) at the boundary, the equation becomes:

$$\vec{n} dt K \nabla h = q dt \quad (18)$$

with  $y = q dt$

### 1.3 Density-dependent groundwater flow

The equation for variable density groundwater flow is given by:

$$S_s \frac{\partial P}{\partial t} = \nabla \frac{\rho_f k}{\mu} (\nabla P + \rho_f g \nabla z) + W \quad (19)$$

#### 1.3.1 Steady-state flow

Governing equation for steady-state variable density flow:

$$-\nabla \frac{\rho_f k}{\mu} (\nabla P + \rho_f g \nabla z) = W \quad (20)$$

Separate pressure and bouyancy term:

$$-\nabla \frac{\rho_f k}{\mu} (\nabla P) = \nabla \frac{\rho_f k}{\mu} \rho_f g \nabla z + W \quad (21)$$

Escript PDE:

$$-\nabla (A \nabla u + B u) + C \nabla u + D u = -\nabla X + Y \quad (22)$$

Reorganize steady state flow eq. and select coefficients to use in escript PDE solver:

$$\underbrace{-\nabla \frac{\rho_f k}{\mu}}_A \underbrace{\nabla P}_u = -\nabla \underbrace{-\frac{\rho_f k}{\mu} \rho_f g \nabla z}_X + \underbrace{W}_Y \quad (23)$$

Escript coefficients:

$$u = P$$

$$A = \frac{\rho_f k}{\mu}$$

$$X = -\frac{\rho_f k}{\mu} \rho_f g \nabla z$$

$$Y = W$$

#### 1.3.2 Transient flow, explicit formulation

$$S_s \frac{\partial P}{\partial t} = -\nabla \frac{\rho_f k}{\mu} (\nabla P + \rho_f g \nabla z) + W \quad (24)$$

with discretization in time following:

$$\frac{\partial P}{\partial t} \approx \frac{P^{t+1} - P^t}{dt} \quad (25)$$

gives:

$$S_s P^{t+1} = -dt \nabla \frac{\rho_f k}{\mu} (\nabla P^t + \rho_f g \nabla z) + W dt + S_s P^t \quad (26)$$

The Escript equation:

$$-\nabla(A\nabla u + Bu) + C\nabla u + Du = -\nabla X + Y \quad (27)$$

Casting the transient flow equation in the form of the escript equation results in:

$$\underbrace{S_s}_D \underbrace{P^{t+1}}_u = -\nabla \underbrace{dt \frac{\rho_f k}{\mu} (\nabla P^t + \rho_f g \nabla z)}_X + \underbrace{W dt + S_s P^t}_Y \quad (28)$$

with the following escript coefficients:

$$u = P^{t+1}$$

$$D = S_s$$

$$X = dt \frac{\rho_f k}{\mu} (\nabla P^t + \rho_f g \nabla z)$$

$$Y = W dt + S_s P^t$$

### 1.3.3 Transient flow, implicit formulation

$$S_s \frac{\partial P}{\partial t} = -\nabla \frac{\rho_f k}{\mu} \nabla (P + \rho_f g z) + W \quad (29)$$

with time discretization of pressure ( $P$ ):

$$\nabla \frac{\rho_f k}{\mu} (\nabla P^{t+1} + \rho_f g \nabla z) + S_s P^{t+1} = +dtW + S_s P^t \quad (30)$$

The Escrip equation:

$$-\nabla(A\nabla u + Bu) + C\nabla u + Du = -\nabla X + Y \quad (31)$$

$$-\nabla - dt \frac{\rho_f k}{\mu} (\nabla P^{t+1} + \rho_f g \nabla z) + S_s P^{t+1} = dtW + S_s P^t \quad (32)$$

$$\underbrace{-\nabla - dt \frac{\rho_f k}{\mu} \nabla P^{t+1}}_A + \underbrace{S_s}_D P^{t+1} = -\nabla \underbrace{dt \frac{\rho_f^2 g k}{\mu} \nabla z}_X + \underbrace{dtW + S_s P^t}_Y \quad (33)$$

Dependent variable:

$$u = P^{t+1}$$

Escrip PDE coefficients:

$$A = -dt \frac{\rho_f k}{\mu}$$

$$D = S_s$$

$$X = dt \frac{\rho_f^2 g k}{\mu} \nabla z$$

$$Y = dtW + S_s P^t$$

## 2 Solute transport

The solute mass balance is given by:

$$\frac{\partial(\phi\rho_f C)}{\partial t} = -\nabla(\rho_f q C) - \nabla\rho_f D \nabla C + W_s \quad (34)$$

(=eq. 38a, [Diersch and Kolditz, 2002])

Which is sometimes simplified by taking  $\rho_f$  and  $q$  out of the derivative on the right hand side of the equation:

$$\frac{\partial(\phi\rho_f C)}{\partial t} = -\rho_f q \nabla C - \nabla\rho_f D \nabla C + W_s \quad (35)$$

eq. 38b, [Diersch and Kolditz, 2002]

or

$$\frac{\partial(\phi\rho_f C)}{\partial t} = -\nabla(\phi_e \rho_f v C) + \nabla\phi_e \rho_f \overline{D_h} \nabla C + W_s \quad (36)$$

eq. 3.11 [Ingebritsen et al., 2006]

(? diffusive flux and advective flux should have same sign? -; no Rift2D uses same eq. (but with constant  $\phi$  and  $\rho_f$ ).

we simplify this eq. by assuming a constant porosity and equal effective and total porosity ( $\phi = \phi_e$ ):

$$\frac{\partial(\rho_f C)}{\partial t} = -\nabla(\rho_f \vec{v} C) + \nabla\rho_f \overline{D_h} \nabla C + \frac{W_s}{\phi} \quad (37)$$

and taking  $v$  out of the derivative of the advective term: (not sure why, but all eq. assume this. check Diersch and Kolditz [2002] paper for explanation)

$$\frac{\partial(\rho_f C)}{\partial t} = \nabla\rho_f D_h \nabla C - \vec{v} \nabla(\rho_f C) + \frac{W_s}{\phi} \quad (38)$$

taking  $\rho_f C$  as the dependent variable

$$\frac{\partial(\rho_f C)}{\partial t} = \nabla D_h \nabla(\rho_f C) - \vec{v} \nabla(\rho_f C) + \frac{W_s}{\phi} \quad (39)$$

or taking just  $C$  as the dependent variable

$$\rho_f \frac{\partial(C)}{\partial t} = \nabla D_h \rho_f \nabla(C) - \vec{v} \rho_f \nabla(C) + \frac{W_s}{\phi} \quad (40)$$

$$v = \frac{q}{\phi} \quad (41)$$

### 2.1 Steady state

The steady state solute balance equation reduces to:

$$-\nabla D_h \nabla(\rho_f C) + \vec{v} \nabla(\rho_f C) = -\frac{W_s}{\phi} \quad (42)$$

Esript equation:

$$-\nabla(A \nabla u + B u) + C \nabla u + D u = -\nabla X + Y \quad (43)$$

$$-\nabla \underbrace{D_h}_A \nabla(\rho_f C) + \underbrace{\vec{v}}_C \nabla(\rho_f C) = -\underbrace{\frac{W_s}{\phi}}_Y \quad (44)$$

$$\begin{aligned} u &= \rho_f C \\ A &= D_h \\ C &= \vec{v} \\ Y &= -\frac{W_s}{\phi} \end{aligned}$$

## 2.2 Explicit form

$$\frac{\partial(\rho_f C)}{\partial t} = \nabla D_h \nabla(\rho_f C) - \vec{v} \nabla(\rho_f C) + \frac{W_s}{\phi} \quad (45)$$

With  $u = \rho_f C$  the discretized form of this equation is:

$$u^{t+1} = u^t + dt \nabla D_h \nabla u^t - dt \vec{v} \nabla u^t + dt \frac{W_s}{\phi} \quad (46)$$

Esript equation:

$$-\nabla(A \nabla u + B u) + C \nabla u + D u = -\nabla X + Y \quad (47)$$

$$\underbrace{1}_D u^{t+1} = -\nabla \underbrace{-dt D_h \nabla u^t}_X + \underbrace{u^t - dt \vec{v} \nabla u^t + dt \frac{W_s}{\phi}}_Y \quad (48)$$

Dependent variable:

$$u = \rho_f C^{t+1}$$

Esript constants:

$$D = 1$$

$$X = -dt \overline{D_h} \nabla u^t$$

$$Y = u^t - dt \vec{v} \nabla u^t + dt \frac{W_s}{\phi}$$

## 2.3 Implicit form

$$\frac{\partial(\rho_f C)}{\partial t} = \nabla D_h \nabla(\rho_f C) - \vec{v} \nabla(\rho_f C) + \frac{W_s}{\phi} \quad (49)$$

Time discretization of derivative of  $u = \rho_f C$ :

$$\frac{\partial u}{\partial t} = \frac{u^{t+1} - u^t}{dt} \quad (50)$$

Insert into solute transport equation:

$$-\nabla dt \overline{D_h} \nabla u^{t+1} - dt \vec{v} \nabla u^{t+1} + u^{t+1} = u^t + dt W_s \quad (51)$$

Esript equation:

$$-\nabla(A \nabla u + B u) + C \nabla u + D u = -\nabla X + Y \quad (52)$$

Cast implicit solute transport equation into escript form:

$$-\nabla \underbrace{dt D_h}_A \nabla u^{t+1} + \underbrace{(dt \vec{v})}_C \nabla u^{t+1} + \underbrace{1}_D u^{t+1} = \underbrace{u^t + dt W_s}_Y \quad (53)$$

Dependent variable:

$$u = \rho_f C^{t+1}$$

Esript constants:

$$A = dt \overline{D_h}$$

$$C = dt \vec{v}$$

$$D = 1$$

$$Y = u^t + dt W_s$$



### 3 Coupled fluid flow and solute transport

coupled eq. by *Ingebritsen et al.* [2006], eq. 3.28:

$$\left( \rho_f \frac{\partial \phi}{\partial P} + \phi \frac{\partial \rho_f}{\partial P} \right) \frac{\partial P}{\partial t} + \gamma \phi \frac{\partial C}{\partial t} = \nabla \cdot \left( \frac{\rho_f k}{\mu} \nabla (P + \rho_f g z) \right) \quad (54)$$

Fluid flux is given by:

$$q = -\frac{k}{\mu} \nabla (P + \rho_f g z) \quad (55)$$

And solute transport is calculated as:

$$\frac{\partial C}{\partial t} = \nabla D \nabla C + v \nabla C \quad (56)$$

where  $D$  is the diffusion coefficient. TODO: use full eq. instead...

The equations are coupled through the fluid density term.

Fluid density depends on the solute concentration and temperature. For now we only take solute concentration into account. Fluid density is calculated as:

$$\rho_f = \rho_{f,0} + \gamma(C - C_0) \quad (57)$$

where  $\rho_{f,0}$  is the fluid density at a reference concentration  $C_0$  and  $\gamma$  is the ratio of change in fluid density to a change in concentration.

#### 3.1 Sequential coupling, explicit formulation (forward in time)

To solve the solute transport and fluid flow equations we use an iterative solution scheme proposed by *Ackerer* [2004].

The three equations are solved sequentially:

1. The flux is calculated using Darcy's law (eq. 55). For the first iteration we use the pressure field of the previous timestep or the initial pressure.
2. Solve the solute mass balance PDE (eq. 56) to get  $\rho_f C$
3. Calculate the solute concentration  $C$  using the existing fluid density.
4. Update the fluid density  $\rho_f$  using equation 57
5. Solve the groundwater flow equation (eq. 54) for pressure  $P$ .
6. Go back to step 1 and repeat until both pressure  $P$  and solute concentration  $C$  do not change significantly between iterations.

##### 3.1.1 Explicit form coupled gw flow eq:

$$S_s \frac{\partial P}{\partial t} + \gamma \phi \rho_f \frac{\partial C}{\partial t} + \alpha_f \phi \rho_f \frac{\partial T}{\partial t} = \nabla \cdot \frac{\rho_f k}{\mu} (\nabla P + \rho_f g \nabla z) + W_f \quad (58)$$

We use a discretization of the derivative of pressure over time  $\frac{\partial P}{\partial t}$ :

$$\frac{\partial P}{\partial t} = \frac{P^{t+1} - P^t}{dt} \quad (59)$$

We can now rewrite a discretized form of eq. 63 as:

$$S_s P^{t+1} = dt \nabla \frac{\rho_f k}{\mu} (\nabla P^t + \rho_f g \nabla z) + S_s P^t + dt W_f - dt \gamma \phi \rho_f \frac{\delta C}{\delta t} - dt \alpha_f \phi \rho_f \frac{\delta T}{\delta t} \quad (60)$$

The standard PDE that is solved by Escript/Finley is defined as:

$$-\nabla(A\nabla u + Bu) + C\nabla u + Du = -\nabla X + Y \quad (61)$$

$$\underbrace{S_s}_{D} \underbrace{P^{t+1}}_u = -\nabla \underbrace{-dt \frac{\rho_f k}{\mu} (\nabla P^t + \rho_f g \nabla z)}_X + \underbrace{S_s P^t + dt W_f - dt \gamma \phi \rho_f \frac{\delta C}{\delta t} - dt \alpha_f \phi \rho_f \frac{\delta T}{\delta t}}_Y \quad (62)$$

With the dependent variable:

$$u = P^{t+1}$$

and the Escript PDE coefficients:

$$D = S_s$$

$$X = -dt \frac{\rho_f k}{\mu} (\nabla P^t + \rho_f g \nabla z)$$

$$Y = S_s P^t + dt W_f - dt \gamma \phi \rho_f \frac{\delta C}{\delta t} - dt \alpha_f \phi \rho_f \frac{\delta T}{\delta t}$$

Translated into python code:

```
# construct new variables that contain the PDE coefficients:
```

```
D = Ss
```

```
X = -dt * rho_f * k / viscosity * ( es.grad(P) + rho_f * g * es.grad(z) )
```

```
Y = Ss * Pt + dt * Qf - dt * gamma * phi * rho_f * dCdt - dt * alpha_f * phi * rho_f * dTdt
```

```
# tell escript what the coefficients are:
```

```
gflow_pde.setValues(D=D, X=X, Y=Y)
```

```
# solve the partial differential equation to find P at t+1:
```

```
P = gflow_pde.getSolution()
```

### 3.1.2 Implicit form coupled gw flow eq:

Governing equation density dependent groundwater flow, with terms that account for mass changes with changes in solute concentration and temperature:

$$S_s \frac{\partial P}{\partial t} + \gamma \phi \rho_f \frac{\partial C}{\partial t} + \alpha_f \phi \rho_f \frac{\partial T}{\partial t} = \nabla \frac{\rho_f k}{\mu} (\nabla P + \rho_f g \nabla z) + W_f \quad (63)$$

We use a discretization of the derivative of pressure over time  $\frac{\partial P}{\partial t}$ :

$$\frac{\partial P}{\partial t} = \frac{P^{t+1} - P^t}{dt} \quad (64)$$

We can now rewrite a discretized form of eq. 63 as:

$$-\nabla dt \frac{\rho_f k}{\mu} \nabla P^{t+1} + S_s P^{t+1} = \nabla dt \frac{\rho_f^2 g k}{\mu} \nabla z + S_s P^t - dt \gamma \phi \frac{\partial C}{\partial t} - dt \alpha_f \phi \rho_f \frac{\partial T}{\partial t} + dt W_f \quad (65)$$

The standard PDE that is solved by Escript/Finley is defined as:

$$-\nabla(A\nabla u + Bu) + C\nabla u + Du = -\nabla X + Y \quad (66)$$

The parts that we do not need to solve the groundwater flow equation are greyed out. We rewrite the groundwater flow equation to conform with the Escript PDE:

$$-\nabla \underbrace{dt \frac{\rho_f k}{\mu}}_A \nabla P^{t+1} + \underbrace{S_s}_{D} P^{t+1} = -\nabla \underbrace{dt \frac{\rho_f^2 g k}{\mu}}_X \nabla z + \underbrace{S_s P^t - dt \gamma \phi \frac{\partial C}{\partial t} - dt \alpha_f \phi \rho_f \frac{\partial T}{\partial t} + dt W_f}_Y \quad (67)$$

With the dependent variable:

$$u = P^{t+1}$$

and the Escript PDE coefficients:

$$A = dt \frac{\rho_f k}{\mu}$$

$$D = S_s$$

$$X = -dt \frac{\rho_f^2 g k}{\mu} \nabla z$$

$$Y = S_s P^t - dt \gamma \phi \frac{\partial C}{\partial t} - dt \alpha_f \phi \rho_f \frac{\partial T}{\partial t} + dt W_f$$

Translated into python code:

```
# construct new variables that contain the PDE coefficients:
A = dt * rho_f * k / viscosity * es.kronecker(mydomain)
D = Ss
X = -dt * rho_f**2 * g * k / viscosity * es.grad(z)
Y = Ss * Pt - dt * gamma * phi * rho_f * dCdt - dt * alpha_f * phi * rho_f * dTdt + dt * Qf

# tell escript what the coefficients are:
gwflow_pde.setValues(A=A, D=D, X=X, Y=Y)

# solve the partial differential equation to find P at t+1:
P = gwflow_pde.getSolution()
```

## 4 Drain boundary

We implement a conditional boundary condition that simulates the transfer of groundwater to surface water bodies, streams or evapotranspiration. The simplest way to simulate groundwater-surface water exchange is by implementing a so called drain boundary function. A drain boundary function is a conditional boundary function that removes water from the boundary of a model domain if the pressure or hydraulic head exceed a certain value.

$$W_d = P_d C_d \quad (68)$$

where  $W_d$  is the drain flux  $S^{-1}$ ,  $P_d$  is the excess pressure at boundary nodes ( $P_a$ ), which is defined as  $P - P_s$  for  $P > P_s$  and 0 for  $P \leq P_s$ .  $P_s$  is the pressure at the surface boundary. To keep things simple we ignore atmospheric pressure and set  $P_s = 0$ .

This formulation is highly non-linear, and requires to check which parts of the model boundary contribute to the drain flux before solving the groundwater flow equation. An alternative is to find a smooth function that reduces to 0 for  $P < P_s$ . One possibility would be using the error function. The error function is included in escript. The error function varies between  $y=-1$  for  $x=-6$  to  $y=1$  for  $x=6$ .  $erf - 2 = -0.99$ . Therefore if we want to start generating a drain flux at  $P_d = 0$  we can reformulate.

$$W_d = C_d \frac{(erf((P_d * 4) - 2) + 1)}{2} \quad (69)$$

$P_{\eta}d$  is the noremalized drain pressure, which is the drain pressure divided by the pressure at which the drain flux reaches a maximum  $P_d - P_{dmax}$ . We set the maximum drain pressure to a hydraulic head of 10 m above the surface level  $P_{dmax} = 10 * 1000.0 * 9.81 \approx 1.0 \times 10^5$

## 5 Analytical equations

### 5.1 One-dimensional flow with replenishment

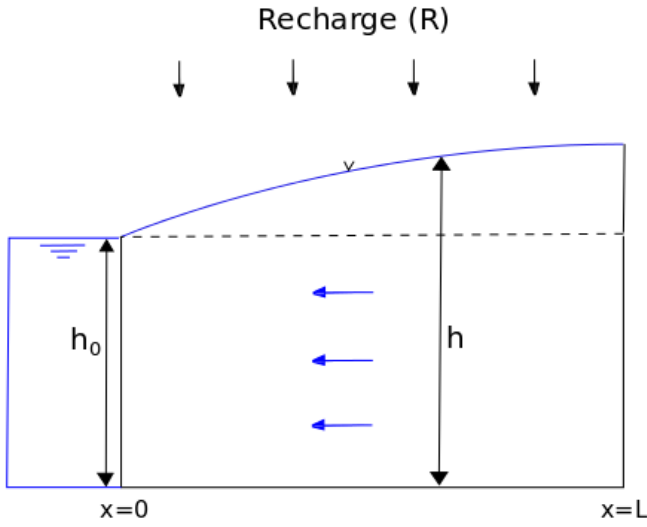


Figure 1: Groundwater flow to a channel with uniform recharge

#### 5.1.1 Unconfined aquifer / free water table

The flow equation with Dupuit assumptions:

$$q = -Kh \frac{\partial h}{\partial x} \quad (70)$$

Continuity

$$q = -R(L - x) \quad (71)$$

combining these two equations gives

$$-R(L - x)\partial x = -Kh\partial h \quad (72)$$

which after integration gives:

$$-R(Lx - \frac{1}{2}x^2) = -\frac{1}{2}Kh^2 + C \quad (73)$$

for the boundary condition  $h = h_0$  for  $x = 0$ :

$$C = \frac{1}{2}Kh_0^2 \quad (74)$$

thus:

$$-R(Lx - \frac{1}{2}x^2) = -\frac{1}{2}Kh^2 + \frac{1}{2}Kh_0^2 \quad (75)$$

rewriting this eq. to find  $h$

$$\frac{1}{2}Kh^2 = \frac{1}{2}Kh_0^2 + R(Lx - \frac{1}{2}x^2) \quad (76)$$

$$h = \sqrt{h_0^2 + 2R/K(Lx - \frac{1}{2}x^2)} \quad (77)$$

### 5.1.2 Confined aquifer

Similarly, the solution for a confined aquifer can be derived by taking

$$q = -KD \frac{\partial h}{\partial x} \quad (78)$$

where  $D$  is the thickness of the aquifer

$$-R(L - x)\partial x = -KD\partial h \quad (79)$$

which after integration gives:

$$-R(Lx - \frac{1}{2}x^2) = -KDh + C \quad (80)$$

for the boundary condition  $h = h_0$  for  $x = 0$ :

$$-R(Lx - \frac{1}{2}x^2) = -KDh + KDh_0 \quad (81)$$

$$h = h_0 + R/(KD)(Lx - \frac{1}{2}x^2) \quad (82)$$

## References

- Ackerer, P., A new coupling algorithm for density-driven flow in porous media, *Geophysical Research Letters*, 31(12), L12,506, doi:10.1029/2004GL019496, 2004.
- Diersch, H.-J., and O. Kolditz, Variable-density flow and transport in porous media: approaches and challenges, *Advances in Water Resources*, 25(812), 899–944, 2002.
- Ingebritsen, S. E., W. E. Sanford, and C. E. Neuzil, *Groundwater in geologic processes*, Cambridge Univ Pr, 2006.