Firn Densification Model

Cummings, Evan Davis, Tyler Brinkerhoff, Douglas

August 24, 2012

1 Introduction

The top layer of snow on a glacier or ice sheet increases in density as depth increases. Several models have been created to simulate this process, some based on temperature and others based on enthalpy. I have re-created these models with the finite-element software package FEniCS.

2 Temperature Solution

We begin with the standard heat-transport equation (Patterson, 2001

$$\rho c_i \frac{\partial T}{\partial t} = k_i \frac{\partial^2 T}{\partial z^2} + \left(\frac{dk_i}{dt} - \rho c_i w \right) \frac{\partial T}{\partial z}$$

with heat sources from the deformation of ice ommitted, ρ density, c_i heat capacity, k_i thermal conductivity, w vertical velocity, and T temperature of firn. To solve the total derivative dk_i/dt we must apply the chain rule

$$\frac{dk_i}{dz} = \frac{\partial k_i}{\partial \rho}\frac{\partial \rho}{\partial z} + \frac{\partial k_i}{\partial T}\frac{\partial T}{\partial z}.$$

The thermal conductiviy of ice is defined by Arthern et. al, 1998 as

$$k_i = 2.1 \left(\frac{\rho}{\rho_i} \right)^2,$$

and gives

$$\frac{\partial k_i}{\partial \rho} = 4.2 \frac{\rho}{\rho_i^2}$$

and

$$\frac{\partial k_i}{\partial T} = \frac{4.2}{\rho_i^2} \left(\frac{\partial \rho}{\partial T} \right).$$

Patterson, 2001 defined ρ in terms of T and from this can be derived

$$\frac{\partial \rho}{\partial T} = 5.6 \times 10^{-2} \exp((-5.7 \times 10^{-3})T).$$

The vertical velocity of ice, w, is directly proportional to the accumulation, b,

$$w = -\frac{b}{\rho}$$

with b in units of kg m^{-2} s^{-1}. The densification process is defined with the material derivative

$$\frac{d\rho}{dt} = \frac{\partial \rho}{\partial t} + w\frac{\partial \rho}{\partial z}.$$

Arthern et. al, 2010 described this total derivative with the formula

$$\frac{d\rho}{dt} = \begin{cases} c_0(\rho_i - \rho), & \rho <= 550\ kg\ m^{-3} \\ c_1(\rho_i - \rho), & \rho > 550\ kg\ m^{-3} \end{cases}.$$

Zwally and Li, 2002 defined the multiplying constant c with an arrhenius-type relation

$$c_0 = c_1 = b\beta(T) \left(\frac{\rho_i}{\rho_w} \right) K_{0G}(T) \exp\left(-\frac{E(T)}{RT} \right),$$

with $K_{0G}(T)\exp(-E(T)/(RT)) = 8.36T^{-2.061}$, and $\beta(T)$ a smoothing function to match a desired density rate. Arthern et. al 2010 developed a semi-empirical formula by coupling the rate equations for Nabarro-Herring creep and normal grain-growth :

$$\begin{cases} c_0 = M_0 bg \frac{k_{c0}}{k_g} \exp\left(-\frac{E_c}{RT} + \frac{E_g}{RT_{avg}} \right) \\ c_1 = M_1 bg \frac{k_{c1}}{k_g} \exp\left(-\frac{E_c}{RT} + \frac{E_g}{RT_{avg}} \right) \end{cases},$$

with the creep coefficients defined as

$$\begin{cases} k_{c0} = 9.2 \times 10^{-9}\,\mathrm{m}^3\ s\ kg^{-1} \\ k_{c1} = 3.7 \times 10^{-9}\,\mathrm{m}^3\ s\ kg^{-1} \end{cases}$$

and M defined by Ligtenberg et. al 2011 to better fit with observed densification rates in higher-temperature environments :

$$\begin{cases} M_0 = 2.366 - 0.293 \ln(b * 1 \times 10^3) \\ M_1 = 1.435 - 0.151 \ln(b * 1 \times 10^3) \end{cases}.$$

Within the same paper a firn surface density expression from data was given :

$$\rho_s = -151.94 + 1.4266(73.6 + 1.06T_s + 0.0669A + 4.77V_a).$$

3 Enthalpy Solution

As stated in Aschwanden et. al 2012, we take 'enthalpy' to be synonymous with 'internal energy' due to the exclusion of work done with changing volume. The equation used here is the shallow-enthalpy :

$$\rho\frac{\partial H}{\partial t} = \frac{\partial}{\partial z}\left(\begin{Bmatrix} K_i(H), & \text{Temperate} \\ K_0, & \text{Cold} \end{Bmatrix} \frac{\partial H}{\partial z} \right) + w\rho\frac{\partial H}{\partial z}.$$

Strain heating has been neglected and the advective term $w\rho\,\partial H/\partial z$ has been added. The coefficient for temperate ice is

$$K_i(H) = \frac{k_i}{c_i}.$$

The coefficient for cold ice is

$$K_0 = \frac{1}{10}K_i(H).$$

Temperate firn is defined as firn with $H > H_s$ and cold firn $H <= H_s$, with

$$H_s = \int_{T_0}^{T_m} c_i(T)dT$$

where $T_m = 273.15$ K and $T_0 = 0.0$ K. The enthalpy can be found with a constant heat capacity of 2009 J kg^{-1} K^{-1} with the linear equation

$$H = \begin{cases} c_i(T - T_0), & \text{where } T <= T_m \\ c_i(T_w - T_0) + \omega L_f, & \text{where } T > T_m \end{cases}$$

where L_f is the latent heat of fusion and ω represents the water content percentage of firn given by

$$\omega L_f = H - c_i(T_w - T_0).$$

It is conveinent to find the temperature :

$$T = \frac{H}{c_i}.$$

The density equations remain the same as the temperature solution, and the surface density at time index $n+1$ can be described as :

$$\rho_s^{n+1} = \rho_b^{n+1} d_p + \rho_s^n (1 - d_p),$$

where

$$\rho_b^{n+1} = \rho_b^{n-1} + \Delta\omega_s \rho_b^n,$$
$$\Delta\omega_s = \omega_s^n - \omega_s^{n-1},$$
$$d_p = \frac{d_n}{l_s},$$
$$d_n = w_s dt,$$

l_s is the length of the surface node, and

$$\rho_b^n = \begin{cases} \rho_w - \rho_i \text{ kg m}^{-3}, & T <= T_w \\ \rho_w \text{ kg m}^{-3}, & T > T_w \end{cases}.$$

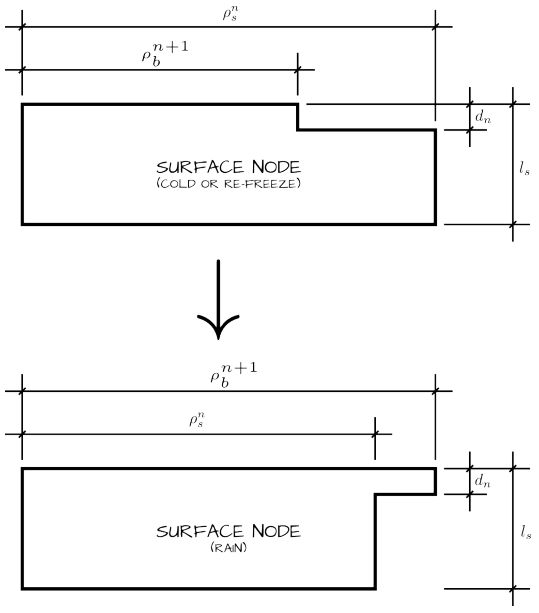

Figure 1: Evolution of surface node density

4 Finite Element Method

This section will focus on the enthalpy solution; solving these equations with FENicS is done with Galerkin's method which requires finding the weak formulation :

$$0 = \int_\Omega \begin{Bmatrix} K_i(H) \\ K_0 \end{Bmatrix} \nabla^2 H\psi \, d\Omega + \int_\Omega w\rho\nabla H\psi \, d\Omega - \int_\Omega \rho\frac{\partial H}{\partial t}\psi \, d\Omega.$$

Here we have integrated over the entire domain, Ω, and multiplied by a test function ψ. After integrating the diffusive term by parts this becomes

$$f_H = -\int_\Omega \begin{Bmatrix} K_i(H) \\ K_0 \end{Bmatrix} \nabla H\nabla\psi \, d\Omega + \int_\Omega w\rho\nabla H\psi \, d\Omega - \int_\Omega \rho\frac{\partial H}{\partial t}\psi \, d\Omega.$$

We can descritize the enthalpy time-differential with the second-order accurate backward-difference formula

$$\frac{\partial H}{\partial t} = \frac{H^{k-2} - 4H^{k-1} + 3H}{2dt},$$

with superscripts referring to time index. This equation can be represented in FENicS as :

```
f_H = (rho*(H_2 - 4*H_1 + 3*H)/(2*dt)*psi
      + k/c*Kcoef*inner(grad(H), grad(psi))
      + rho*w*grad(H)*psi)*dx
```

The variable `Kcoef` is a coefficient vector which will be updated dynamically depending on the temperature of firn (either 1.0 or 0.1).

The weak form for density is found similarly :

$$f_\rho = \int_\Omega \frac{\partial\rho}{\partial t}\phi \, d\Omega + \int_\Omega w\nabla\rho\phi \, d\Omega - \int_\Omega \frac{d\rho}{dt}\phi \, d\Omega.$$

With the partial-time-differential of density defined indentically to the enthalpy equation, this can be represented in FENicS with the Arthern et. all 2010 densification equation as :

```
c       = b*g*rhoCoef/kg *
          exp(-Ec/(R*T) + Eg/(R*Tavg))
drhodt  = c*(rhoi - rho)
f_rho   = ((rho_2 - 4*rho_1 + 3*rho)/(2*dt)
          - drhodt + w*grad(rho))*phi*dx
```

The variable `rhoCoef` is another dynamically-updated-coefficient vector and is either k_{c0} or k_{c1} depending upon the density at the node.

We can define the function space for the entire nonlinear problem as

$$U = \Omega \times \Omega,$$

with corresponding trial and test functions respectively defined as

$$v, u \subset U.$$

The test functions for each function can now be described as

$$\psi, \phi \subset u.$$

In FENicS these spaces can be defined by this :

```
V          = FunctionSpace(mesh,
                           'Lagrange',
                           1)
MV         = V*V
h          = Function(MV)
H,rho      = split(h)
dh         = TrialFunction(MV)
dH, drho   = split(dh)
j          = TestFunction(MV)
psi, phi   = split(j)
```

We define the entire function as

$$f = f_H + f_\rho.$$

Solving this system can be accomplished with *Newton's Method* which requires derivation of the Jacobian :

$$J = \frac{\partial f}{\partial v}.$$

In FENicS this becomes :

```
f  = f_H + f_rho
J  = derivative(f, h, dh)
```

5 Boundary Conditions

The cyclical enthalpy boundary condition for the surface can be simulated with

$$H_s = c_i(T_s - T_0),$$

$$T_s = T_{avg} + \alpha \sin(\omega t),$$

where α is the amplitude of temperature variation and $\omega = 2\pi/spy$ is the frequency. The surface-density boundary condtion can be likewise described as (see figure 1) :

$$\rho_s^{n+1} = \rho_b^{n+1} d_p + \rho_s^n (1 - d_p).$$

Both of these can be created with FENicS with

```
code = 'c*(Tavg + 9.9*sin(omega*t) - T0)'
Hs   = Expression(code, c=cp, Tavg=Tavg,
                  omega=freq, t=t0, T0=T0)

code = 'dp*rhon + (1 - dp)*rhoi'
rhoS = Expression(code, rhon=rhosi,
                  rhoi=rhosi, dp=1.0)
```

```
def surface(x, on_boundary):
  return on_boundary and x[0] == zs

Hbc  = DirichletBC(MV.sub(0), Hs, surface)
Dbc  = DirichletBC(MV.sub(1), rhoS, surface)
```

Within the time-loop the variables `t`, `rhon`, `rhoi`, and `dp` can be updated as needed.

Now all that is left is to iterate through time and call the `solve` method at each step :

```
solve(f == 0, h, [Hbc, Dbc], J=J)
```

The use of the `solve` function in this way chooses *Newton's Method* by default and minimizes the residual of `f`. The boundary conditions are updated by specifing the list `[Hbc, Dbc]`.

6 Model Parameters

Within the time-loop there are a number of parameters which need to be updated. Taking into account conservation of mass, the height l of each node must be recalculated :

$$l^{n+1} = l^n \frac{\rho_{ini}}{\rho},$$

where ρ_{ini} is the density of the firn column when the system was initialized. With the height of the nodes calculated, the z-positions may be found by iterating through the heights and setting the z vector's correspoding cell equal to the current sum. After sucessfull calculation the FENicS `mesh` object must have its coordinates refreshed. These tasks may be completed with the following code :

```
lnew     = l*rhoin[index] / firn.rho[index]
zSum     = zb
zTemp    = zeros(n)
for i in range(n)[1:]:
  zTemp[i] = zSum + lnew[i]
  zSum    += lnew[i]
firn.z[index] = zTemp
mesh.coordinates()[:,0] = firn.z
```

The variable `index` is an array of positions corresponding to the correct ordering of the nodes, necessarry after mesh refinement; `zb` is the z-position of the base of the firn column which does not change and `n` is the nuber of nodes.

The height s at time index n of the original surface may be calculated as follows :

$$s^n = (z_s - z_b)\frac{s^{n-1} - z_b}{z_s - z_b} + w_s dt.$$

This maintains the relative location of the original surface to the current surface and moves downward proportional to w.

For all operations it is conveinent to store all the state data from the simulation in a class object for ease of access. For this purpose the `firn` class was created and contains the signature

```
firn(self, H, T, rho, omega, w,
     k, c, z, index)
```

with `z` the node z-positions as defined above, `index` index of re-ordered mesh locations, and other variables as described earlier in the paper.

7 Variable Definitions

Many variable are used in this simulation and many do not change. These are defined below :

Constants :

Var.	Value	Units	Description
g	9.81	m s^{-2}	gravitational acceleration
R	8.3144621	J mol^{-1} K^{-1}	gas constant
spy	31556926	s	seconds per year
ρ_i	917	kg m^{-3}	density of ice
ρ_w	1000	kg m^{-3}	density of water
ρ_m	550	kg m^{-3}	critical density value
k_i	2.1	W $m^{-1}K^{-1}$	thermal conductivity of ice
c_i	2009	J $kg^{-1}K^{-1}$	heat capacity of ice
T_w	273.15	K	triple point of water
k_g	1.3×10^{-7}	m^2s^{-1}	grain growth coefficient
E_c	60×10^3	J mol^{-1}	act. energy for water in ice
E_g	42.4×10^3	J mol^{-1}	act. energy for grain growth

Model :

Var.	Units	Description
ρ_{si}	kg m^{-3}	initial density at surface
b	kg $m^{-2}s^{-1}$	surface accumulation
A	mm a^{-1}	surface accumulation
V_a	m s^{-1}	mean annual wind speed
T_{avg}	K	average annual temperature
T_s	K	firn surface temperature
z_s	m	surface start z-location
z_b	m	firn base z-location
z_{s0}	m	previous time-step's surface
dz	m	initial z-spacing
l	m	vector of node heights
dt	s	time-step
t_0	s	begin time
t_f	s	end-time

Enthalpy-specific :

Var.	Value	Units	Description
T_0	0.0	K	reference temperature
L_f	3.34×10^5	J kg^{-1}	latent heat of fusion
H_s	$c_i(T_w - T_0)$	J kg^{-1}	Enthalpy of ice at T_w

8 Verification of Program

9 Interpretation