Firn Densification Model

Cummings, Evan Davis, Tyler Brinkerhoff, Douglas

August 22, 2012

THE UNIVERSITY OF MONTANA SNOW, ICE, AND CLIMATE GROUP

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

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

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. We can descritize the temperature differential with the second-order accurate backward-difference formula

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

with superscripts referring to time index. To solve the total derivative dk/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

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

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

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

We again descritize the time partial with the backward-difference formula :

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

Arthern et. al, 2010 described the 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

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

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 :

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

Ligtenberg et. al 2011 developed a firn surface density expression from data :

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

3 Enthalpy Solution

$$\rho \frac{dH}{\partial t} = \frac{\partial}{\partial z} \left(\begin{Bmatrix} K_i(H) \\ K_0 \end{Bmatrix} \frac{\partial H}{\partial z} \right)$$

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

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

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

$$\frac{d\rho}{dt} = c(\rho_i - \rho)$$

$$c = M_O bg \frac{k_c}{k_g} \exp\left(\frac{E_c}{RT} + \frac{E_g}{RT_{avg}}\right)$$

$$k_c = \begin{cases} 3.7 \times 10^{-9}\,, & \rho > 550\; kg\; m^{-3} \\ 9.2 \times 10^{-9}\,, & \rho <= 550\; kg\; m^{-3} \end{cases}$$

$$M_O = \begin{cases} 2.366 - 0.293 \log A, & \rho > 550\; kg\; m^{-3} \\ 1.435 - 0.151 \log A, & \rho <= 550\; kg\; m^{-3} \end{cases}$$

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

$$c_i(T) = 146.3 + 7.253T$$

$$w(z) = -\frac{b}{\rho}$$

$$H = c_i(T_w - T_0) + \omega L_f, \text{where } H > H_{sp}$$

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

$$T = \frac{H}{c}$$

$$l_{n+1} = l_n \frac{\rho_{ini}}{\rho}$$

$$s_n = (z_s - z_b)\frac{s_{n-1} - z_b}{z_s - z_b} + w_s dt$$

4 Method

The implicit method used by the numerical method of lines to solve PDEs uses the future to predict the future. First the diffusion term:

$$\frac{k}{\rho C_p}\frac{\partial^2 \theta(x,z,t)}{\partial z^2}.$$

After discretization this becomes

$$\frac{k}{\rho C_p}\frac{u^n_{j+1} - 2u^n_j + u^n_{j-1}}{\Delta z^2},$$

where j is the z-position and n is the time-position.

This can be represented as a matrix as

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & \cdots & \cdots & 0 \\ \alpha & -2\alpha & \alpha & 0 & \cdots & \cdots & 0 \\ 0 & \alpha & -2\alpha & \alpha & \cdots & \cdots & 0 \\ & \ddots & \ddots & \ddots & & & \\ \vdots & \cdots & \alpha & -2\alpha & \alpha & \cdots & \vdots \\ & & & \ddots & \ddots & \ddots & \\ 0 & \cdots & \cdots & 0 & \alpha & -2\alpha & \alpha \\ 0 & \cdots & \cdots & 0 & 0 & 0 & 0 \end{bmatrix},$$

where $\alpha = \frac{k}{\rho C_p \Delta z^2}$.

Now for the vertical advection term :

$$w(z)\frac{\partial \theta}{\partial z}.$$

Taking the velocity to be positive and into the column of ice, this becomes

$$w(z)\frac{-3u^n_j + 4u^n_{j+1} - u^n_{j+2}}{2\Delta z},$$

which can be represented as a matrix as:

$$w(z)\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & \cdots & 0 \\ -3\beta & 4\beta & -\beta & 0 & \cdots & \cdots & 0 \\ 0 & -3\beta & 4\beta & -\beta & \cdots & \cdots & 0 \\ & \ddots & \ddots & \ddots & & & \\ \vdots & \cdots & -3\beta & 4\beta & -\beta & \cdots & \vdots \\ & & & \ddots & \ddots & \ddots & \\ 0 & \cdots & \cdots & 0 & -3\beta & 4\beta & -\beta \\ 0 & \cdots & \cdots & 0 & 0 & 0 & 1 \end{bmatrix},$$

where $\beta = \frac{1}{2\Delta z}$ and $w(z)$ is a vector with its length equal to the number of nodes.

The values for horizontal advection and heat sources from the deformation of ice can be calculated directly without the need for a matrix.

The surface temperature can be designated any value desired, here a function:

$$\theta(z_s) = -10 + 5\sin\left(2\pi\frac{t}{spy}\right),$$

where spy is the number of seconds in a year.

Geothermal heat flow can be described with

$$k\frac{\partial\theta}{\partial z}\bigg|_{z=z_b} = -Q_{geo},$$

where

$$\frac{\partial\theta}{\partial z} = \frac{u_{j-2}^n - 4u_{j-1}^n + 3u_j^n}{2\Delta z},$$

$$Q_{geo} = 42.0\times 10^{-3}\left(\frac{K}{m}\right),$$

and solving for u_j on the bottom results in

$$u_j^n = \frac{1}{3}\left(\frac{-2\Delta z Q_{geo}}{k} + 4u_{j-1}^n - u_{j-2}^n\right).$$

Under the pressure of the column of ice the melting point may drop below $0°C$. This pressure melting point of ice is defined as

$$\theta_{PMP} = \beta\rho g(z_s - z_b) \text{ in } °C,$$

where $g \approx 9.81\frac{m}{s^2}$ is the acceleration of gravity and $\beta = 9.8\times 19^{-8}\frac{K}{Pa}$ is the pressure dependence of the melting point. Because ice and water maintain the same temperature while undergoing a change of phase, θ_{PMP} is the maximum temperature that the ice can reach.

Utilizing Python's sparse matrix package simplifies the creation of the **A** and **B** matrices. Once these are created and the variables and constants defined, python's ODE integrator VODE in 'BDF' mode can be efficiently used to calculate θ for each time-step. VODE requires the right-hand-side function just described:

Algorithm 1 Right-hand-Side Function

```
$\theta_{z_b} := \frac{1}{3}\left(\frac{-2\Delta z Q_{geo}}{k} + 4\theta_{z_b-1} - \theta_{z_b-2}\right)$
$\vec{\theta} := \mathbf{A}\vec{\theta} - \vec{w}\mathbf{B}\vec{\theta} - \vec{u}\frac{\partial\theta}{\partial x} + \frac{\vec{\phi}}{\rho C_p}$
for $i := 0$ to $z_b$ do
  if $\theta_{z_i} \geq \theta_{PMP}$ then
    $\theta_{z_i} := \theta_{PMP}$
  end if
end for
return $\vec{\theta}$
```

Here is the equivalent of Algorithm 1 in python code:

```
def rhs(t, y):
  # heat flow from bed
  y[-1] = (-Qgeo*2*dz/k + 4*y[-2] - y[-3])/3.0
  # set max temp :
  for i in range(n):
    if y[i] >= Tpmp :
      y[i] = Tpmp
  y = A*y - w*(B * y) - u*dTdx + phi/(rho * cp)
  return y
```

Note: The extra variables used are defined globally, and therefore are not needed in the parameter list.

5 Verification of Program

Using the values given in the previous two sections results in these figures :

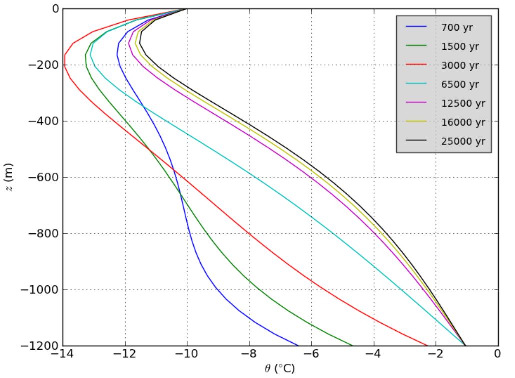

Figure 1: Temperatures of Ice in 30,000 Years.

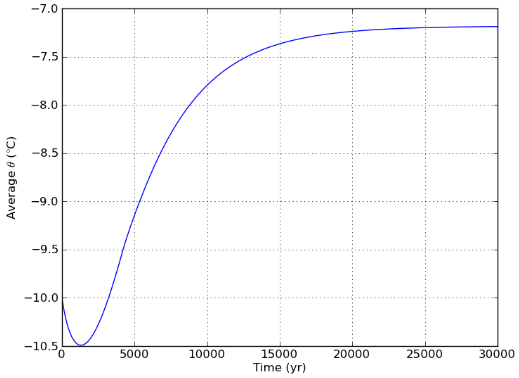

Figure 2: Average Temperature of Ice in 30,000 Years.

These data show that the temperature finds an equilibrium in around 25,000 years.

To find the maximum depth where melting occurs, it is necessary to zoom into the region on the surface :

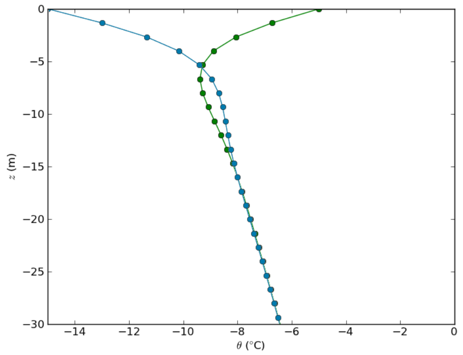

Figure 3: Annual Low and High Temperatures.

These data show that the temperature of the ice stays roughly constant around -16 meters. Figures 4 and 5 show the results after biasing the temperature by $+15\ ^\circ C$:

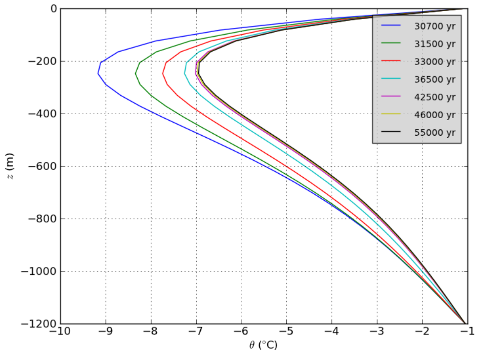

Figure 4: Temperature of Ice after Biasing the Temperature by $+15\ ^\circ C$

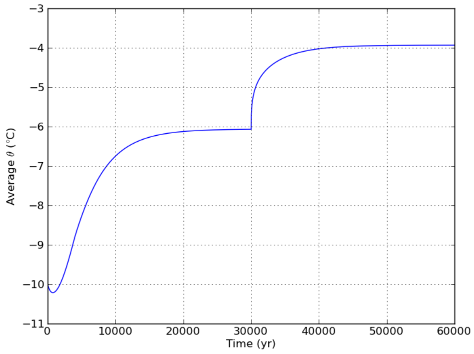

Figure 5: Average Temperature of Ice in 60,000 Years.

These data show that even though the surface temperature has increased by $15\ ^\circ C$, the average temperature of the ice has only increased $2\ ^\circ C$.

I made an investigation to find the precise value of Q_{geo} to create a change of phase on the bed of the ice sheet and found that with *any* value of Q_{geo} with the depth of ice around 50 meters, the bed would eventually reach θ_{PMP} and incur a change of phase. Once the ice temperature on the bottom gains any amount of heat, a chain reaction occurs which eventually brings the bottom temperature up to θ_{PMP}, which for this depth is close to 0. For increased ice thickness, the temperature on the bed could decrease without bound if Q_{geo} was insufficient to overtake the horizontal velocity of ice.

6 Interpretation

Creating this model provided a good deal of challenges. Carefull examination of unit calculations was necessary, and a good deal of the work was based on theoretical concepts. When plugging equations into a python script and testing results, errors would occur which were quite difficult to diagnose. Creating this system required a good amount of courage to keep going forward and not to stop to check and re-check what I had already checked.

This model can be improved in a number of ways. Accumulation rates on the surface and the rate of water runoff on the bed would provide better insight into the effects of increasing geothermal heat-flux. If the water were able to escape from underneath the ice and no accumulation were to occur, a simulation could be made to determine the length of time and amount of energy needed to melt the entire ice sheet.

Although computationally expensive it would seem feasible to simulate multiple columns of ice simultaneously. This would provide a view of the ice in a planar view and may be visualized as a vector-field. A three-dimensional view of the ice-sheet may be useful if color-coded to show different heat patterns on the surface; a planar view of the inside of the ice which can be adjusted for depth would also be insightful.

With the correct equations and data regarding volcanoes, quite an interesting simulation could be performed. Data could be obtained to determine the length of time needed preceding an eruption for an ice-sheet to repair itself. A similar simulation regarding atomic weapons would also provide insight into theoretical worst-case scenarios.

I feel that a simulation of a subject is best accomplished when you can do *anything* to the environment, and this project has brought me one step closer to understanding how that could be possible.