Lab Week 5 – Heat Flow and Temperature in Ice

Heat Transfer in Ice

This week's lab is designed to give you a better intuition regarding the time scales of heat diffusion in ice. We use a few different solutions (numerical and analytical for some situations) of the heat equation

$$\frac{\partial T}{\partial t} = \alpha \nabla^2 T - u \cdot \nabla T + \frac{\varphi}{\rho C_p}$$

to test some simple scenarios. It is not important that you understand the details of the numerical implementation of the models, or that you have a background in coding with python. Rather, the intent here is that you adjust inputs to the models and think about the results.

To gain experience with solutions to the heat equation, we have prepared a Python notebook, which we hope will allow you to vary input parameters for a few scenarios and learn to better anticipate what the results might be.

Some comments on using this notebook:

- There are different cells for code and for 'markdown' which is the text. If you double click on the text you can change it, but there is no reason to do that (aside from the typos that inevitably exist somewhere).
- I have provided helper notes on locations where I think that you should change the code to play around with one of the models (### Please Change! ###), as well as notes in places where you probably shouldn't change the code (### Don't Change ###). Having said that, if you are familiar with python feel free to change whatever you want.
- The code can be run with the buttons at the top or from the keyboard. Press 'Shift+Enter' to run a cell and advance to the next cell or 'Ctrl+Enter' to run a cell and stay in it.

1) Surface Perturbation

First, we will examine a very simple scenario, where the entire domain begins at one temperature and then the surface is perturbed to a new temperature. Heat will begin to diffuse into the domain and warm up the ice below the surface. There is an analytical solution for this problem:

$$T = T_0 + \Delta T \, \mathrm{erfc}\left(\frac{z}{2\sqrt{\alpha t}}\right)$$

where erfc is the 'complementary error function'. This function is common in analytical solutions to the heat (and other diffusion) equation, so it is important to gain some experience with it.

Python Notebook – Try it out
Try running this function and making a figure! Everything is set up for you below, but try adjusting some of the parameters like T0 (initial temperature), dT (temperature change), t (time), and zs (distances from the surface) to get a feeling for the time scales for diffusion of this surface perturbation.

Questions:

1) How do the results compare to your intuition about how long it should take for heat to diffuse into the ice?

2) If the temperature change is larger, does the perturbation propagate faster? Or is it simply a scaling factor?

3) Did you try a different α (thermal diffusivity)? Does the rate of propagation scale linearly with α? Look back at the equation to help guide your answer.

2) Harmonic Surface Boundary
Now we are going to let the surface boundary condition change in time. If we chose to make the surface temperature a sinusoidal function, there is again an analytic solution to the heat equation.

Surface temperature in time:

$$T(0,t) = T_0 + T_a \sin(2\pi\omega t)$$

The analytic solution is:

$$T(z,t) = T_0 + T_a \exp\left(-z\sqrt{\frac{\pi\omega}{a}}\right) \sin\left(2\pi\omega t - z\sqrt{\frac{\pi\omega}{a}}\right)$$

Python Notebook - Try this new solution
Same as before, we want to play around with this model to get a feel for how it works. There are initial some initial values in the Python Notebook to try out, and a plotting script, but try different numbers to get a feel for the behavior.

*Please note that if you change the frequency you are probably going to have to change the times and the depths as well or you won't be able to see anything useful in the plot.

Questions:

In Cuffey and Paterson (2010) Chapter 9, they give us some helpful insight into the equation that we were using above:

- The amplitude of the wave decreases as $\exp\left(-z\sqrt{\frac{\pi\omega}{a}}\right)$. Thus, the higher the frequency, the more rapid the attenuation with depth.

- Temperature maxima and minima propagate at a velocity $2\sqrt{\pi\omega\alpha}$

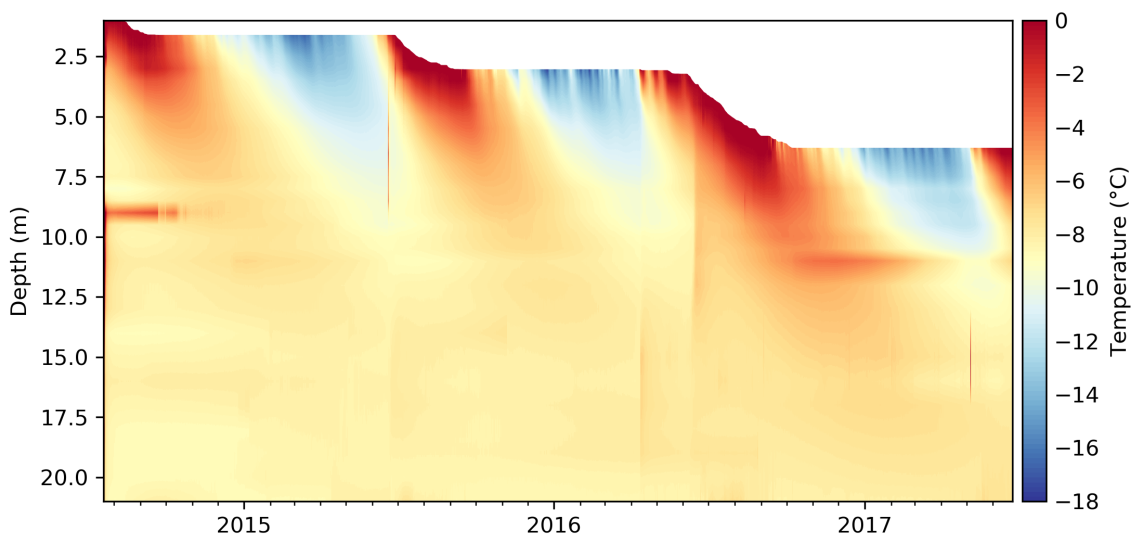
1) Use the points above to fill in the table from the book (reproduced below). Note that they use period P_ω (I use frequency $\omega = \frac{1}{P_\omega}$). You could complete the table with only the equations, but try playing with the model to better visualize the result.

Table 9.2: Propagation by conduction of a cyclical variation in surface temperature.

P_ω(yr)	z_5(m)	v(m yr^{-1})	Δt(yr)
1			
2500			
10^5			

P_ω = period = ω^{-1}.
z_5 = depth at which amplitude is 5% of surface value.
v = velocity of propagation of maxima and minima.
Δt = time lag = z_5/v.

2) In class on Wednesday we looked at some real data. I have plotted it again below. The 'winter cold wave' persists in the ice well into the summer months. Does this agree with your results above? What was the Δt that you got for 1yr? Is it about the same as in the data?



3) Can you derive the functions that Cuffey and Paterson (2010) give from the equation that we used above? Give it a try. The term for velocity might be a bit tricky. Ask yourself when is the sine wave at a maximum or minimum for a given depth, then solve for dz/dt.

4) The 'skin depth' is that at which the amplitude of variations are reduced to 1/e, or 0.367, of the amplitude at the surface. Use the first bulleted point from Cuffey and Paterson (2010) above to derive the skin depth in our case. (Hint: this is very similar to their value z_5).

3) Numerical Solution
Sometimes (in fact most of the time) physical problems are not as elegant as those we have posted above. Often, there is no analytic solution to a problem for the boundary conditions that describe that situation. When problems are more difficult to solve, we must solve use using numerical solutions rather than analytical ones Below, we will progress from these specialized cases where the surface temperature had to be either fixed or harmonic, to a numerical solution where we can tell the surface boundary to be any temperature and vary with time.

Note that the coding here is a more complex than for the previous cases, so feel free to run the code without paying attention to specific and focus on the results (varying input parameters and plotting the solutions). If you are keen to dig into the code, please do, and let us know if you find a more elegant way to implement a numerical discretization of the heat equation.

Python – Model is Written
Ok, if you ran the cell above, your computer knows the numerical model. Now we can run it!

Below I set up a sample problem again. For now, it is exactly the same sine wave as in problem (2). Try running this to see if the answer agrees with the analytical solution. Then try a more complicated surface temperature.

Questions:

1) How is the numerical result different from the analytical result? Pay attention to the 'initial condition' that you started the model at (if you didn't change anything this initial condition should be constant at the air temperature).

2) Try making the surface boundary condition something more complicated than a simple sine wave. How does the model respond?

3) Try overlaying several sine waves of different frequencies. Before you plot the model result, think about which frequencies will propagate further into the ice. Does the result agree with your intuition? (Hint: you will probably have to change the times and the depths in the plots as well to see the result in the figures).

Python – Animating the result
The cell below will animate the model result that you calculated above. So if you don’t find it helpful, don't use it.