From: Archie Paulson <archie.paulson@colorado.edu>
Subject: **glacier model improvement**
Date: March 2, 2008 7:58:25 PM MST
To: Chris Malley <cmalley@pixelzoom.com>
Cc: Wendy Kristine Adams <Wendy.Adams@colorado.edu>, Kathy Perkins <Katherine.Perkins@colorado.edu>
2 Attachments, 16.2 KB Save

Chris,

So here's the new model. I expect this one to be good enough to not change any more. A working implementation of this model is in the attached Python code.

It works like this:
- the climate model (unchanged) is used to compute an ELA
- the entire equilibrium glacier shape is determined from the ELA
- if the climate is changed, the ELA also changes (immediately); then a "psuedo-ELA" is computed that exponentially evolves from the former ELA to the new (current) ELA, and the glacier shape is then governed by this psuedo-ELA

The equilibrium shape is calculated as follows:

1. given an ELA (stored in variable ela), compute
```
x_terminus = 170.5e3-41.8*ela
H_max = 400.-(1.04e-2*ela-23)**2
x_peak = 0.5 * x_terminus
```
2. compute the glacier height (H) at each x-value (x), where both H and x arrays are indexed by i, as follows:
```
for each i:
    if x[i] < x_peak:
        p = 42-0.01*ela
        f = 1.5
        r =f*x_peak
        H[i] = sqrt(r**2 - (x[i]-x_peak)**2) * H_max/r
        H[i] *= ( x_peak**p - (abs(x[i]-x_peak)**p))/ x_peak**p
    elif x[i] < x_terminus:
        H[i] = sqrt(x_peak**2 - (x[i]-x_peak)**2) * H_max/x_peak
    else:
        H[i] = 0.0
```

Obviously the syntax there is pretty much Python. Let me know if anything is ambiguous. Please see the attached python code for a running example.

The ice velocities are computed the same way as before, using the current glacier height.

The three controls govern three variables that I've called t0, p0 and pmax. These are what we've called "sea level temperature", "snowfall transition elevation" and "max snowfall", respectively. Note that pmax is twice the value of the snowfall at elevation p0, which is on the current control.

The ranges for these values should be:
```
t0_min,t0_max = 13., 20.
p0_min,p0_max = 2.2e3, 6e3
pmax_min,pmax_max = 0.0, 4.0
```

With these ranges, the longest glacier you get is just shy of 80 km.

Let me know if there's anything not clear.

-Archie


model.py (13.5 KB)

lib.py (2.7 KB)