version 1.0 (Bueler) June 2024

Exercises in Numerical Modeling of Glaciers

These exercises are aligned to my new slides (slides/slides-2024.pdf) *more than to the printed notes, but all questions are encouraged! Abbreviations: SKE = surface kinematical equation, SIA = shallow ice approximation.*

Paper exercises

1. Sketch a smooth ice surface $z = s(x)$ in the planar case. Convince yourself that the vectors $\mathbf{n}_s = (-\frac{\partial s}{\partial x}, 1)$ are normal to the surface and upward. What is the equivalent formula when $z = s(x, y)$, i.e. in 3D reality?

2. *(Do with a friend.)* For the ice thickness $H = s - b$, what inequality is equivalent to "$s \ge b$"? Which other geophysical fluid layer problems, in cryosphere contexts and outside, have the same inequalities? What (physically) determines the shapes of these fluid layers, and their margins and termini?

3. Assume $a = 0$ and $w = 0$ here, so that the SKE is simply $\frac{\partial s}{\partial t} + u\frac{\partial s}{\partial x} = 0$. For grid spacing $\Delta t, \Delta x$, write down the upwind scheme (see the slides) for both the $u_j \ge 0$ and $u_j \le 0$ cases. Show that if the CFL condition $|u_j|\frac{\Delta t}{\Delta x} \le 1$ applies then the new surface value is an average of the old values. (For instance, when $u_j \ge 0$ show that $s_j^{\text{new}} = c_{-1}s_{j-1} + c_0 s_j$ where c_{-1}, c_0 are nonnegative and sum to one.) What will happen if Δt is too large, that is, if the CFL condition is violated?

4. In the planar case, derive the mass continuity equation ($\frac{\partial H}{\partial t} + \frac{\partial}{\partial x}(\bar{u}H) = a$) from the SKE ($\frac{\partial s}{\partial t} + u\frac{\partial s}{\partial x} = a + w$), using the incompressibility of ice ($\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} = 0$). Here $\bar{u}(t,x) = H(t,x)^{-1}\int_{b(t,x)}^{s(t,x)} u(t,x,z)\,dz$ is the vertically-averaged ice velocity; note $\bar{u}H = \int_b^s u\,dz$ is called the *ice flux*. In the derivation, assume that the ice base is non-sliding ($u(t,x,b(x)) = 0$), non-penetrating ($w(t,x,b(x)) = 0$), and that the bedrock is not moving ($\frac{\partial b}{\partial t} = 0$). Also, use the Leibniz rule for differentiating integrals:

$$\frac{d}{dx}\left(\int_{g(x)}^{f(x)} k(x,y)\,dy\right) = f'(x)k(x,f(x)) - g'(x)k(x,g(x)) + \int_{g(x)}^{f(x)} \frac{\partial k}{\partial x}(x,y)\,dy.$$

5. Write out the details of the slab-on-a-slope calculation from the slides. Thereby derive the ($n = 3$) velocity formula $u(z) = u_0 + \frac{1}{2}A(\rho g \sin\alpha)^3\,(H^4 - (H-z)^4)$. Now add in x-dependence, to see the velocity formula for the SIA model velocity.

6. *(Do with a friend.)* Sketch a hypothetical planar glacier shape, with smooth surface and bed. Sketch what you think the non-sliding SIA formulas will generate for the surface values of the horizontal (u) and vertical (w) velocity components. How would the non-sliding Stokes model change your pictures? Repeat with a different hypothetical shape.

Computer exercises

> *These exercises are based on Python codes in the* `py/` *directory of* `github.com/bueler/mccarthy`*; see the* `README.md`*. Note that when you modify a code, first copy it to a new source file and then modify that. Keep the original, running code in pristine condition to check how things are supposed to work.*

7. Modify `surface1d.py` to systematically test its stability. That is, demonstrate conditional stability of the upwind numerical scheme for the SKE, by varying the grid spacings Δt and Δx, in the given-velocity case. Confirm that the CFL condition $|u_j| \frac{\Delta t}{\Delta x} \le 1$ predicts which cases destabilize and which remain stable.

8. Modify `surface1d.py` to show that the scheme can generate surfaces which fail $s \ge b$, i.e. which violate *admissibility*. (One can increase the ablation rate to cause this.) Modify the scheme so that it maintains admissibility via *truncation*. That is, when a preliminary surface elevation $\tilde{s}_j^{\text{new}}$ is calculated, apply $s_j^{\text{new}} = \max\{\tilde{s}_j^{\text{new}}, b_j\}$.

9. Construct a case where the horizontal velocity depends on time and space: $u = u(t,x)$. Modify `surface1d.py` to use this case. Modify the time-stepping scheme so that the time step is determined via the CFL condition. That is, at each time step, find the largest horizontal speed and adjust the next Δt accordingly.

Comment. Any serious numerical model for glacier geometry should at least have **8** and **9** capabilities above. The surface mass balance could then come from an energy balance submodel, and the surface velocities from any stress balance submodel. However, numerical stability will still not be guaranteed because ice flows downhill.

10. Modify `surface1d.py` so that the data a, u, w can be read from a file. Check that it runs as before, that is, for values which are the same as in the unmodified code.

11. In a suitable textbook, read about *implicit time-stepping*, and perhaps also about the *method of lines*. Modify `surface1d.py` to use an implicit scheme. Do you see benefits? What replaces truncation—see **7 c)**—in an implicit scheme?

12. Modify the ice geometry in `shallowuw.py` to a case where the glacier terminates. By trying different fractional power shapes for the terminus (margin), show that in some cases the SIA surface velocity goes to zero as one approaches the terminus, and in some cases not. (*Look up "Vialov profile" for a bad case.*) Explain what is going on in terms of the powers which appear in the SIA formulas for surface velocity.

13. Combine `surface1d.py` and `shallowuw.py` into one code which numerically solves the time-dependent SIA model via explicit time stepping. That is, alternately compute surface velocity from the current geometry (`shallowuw.py` does this), and then update the geometry using a time step of the SKE (`surface1d.py` does this). Now what is the time step condition for stability? How might one detect instability as the scheme runs?