Interactive Transient and Steady-state Analysis of Regional Ice Flow C30-W65A

Jed Brown jedbrown@mcs.anl.gov, Iulian Grindeanu, Dmitry Karpeev, Barry Smith, Tim Tautges

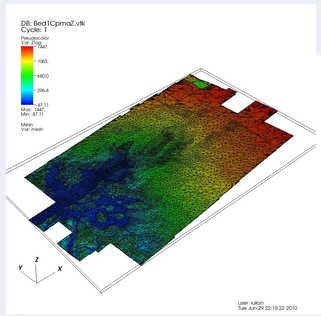
Download this poster from http://59A2.org/files/WCRP2011-RegionalIceFlow.pdf

Preprocessing

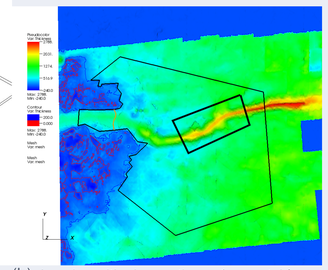
Model data for a regional ice flow models often comes from many different sources, each with its own gridding, projection, and geoid/ellipsoid choices. Manipulating these data into a common form suitable for a numerical model takes a great deal of time and is error-prone. Although often not present in the raw data files due to lack of standardized formats (or not adhering to standards), someone has usually provided the semantic information to GIS software in order to visualize each data set. To reduce the number of intermediate steps, our software reads input in any coordinate system from a variety of GIS formats (any supported by the GDAL package) and converts on the fly to a chosen model coordinate system. Each input data set can use a different format and coordinate system, but any necessary conversions are transparent to the user. In parallel, each process reads only the necessary part of each input file.

Geometry and Mesh Generation

Constructing an unstructured hexahedral mesh with refinement in a region of interest typically involves several steps, due to the general-purpose nature of most meshing packages. We automate the steps by simplifying the input to bed and surface in any format/coordinate system, a closed polygon defining the domain (usually traced by picking corners in GIS program), an optional refinement indicator (as an expression or bitmap), and optional manifold surfaces to be meshed. From these inputs, universal kriging is used to produce an initial fine triangular mesh, this mesh is decimated to produce a variable-resolution representation (see 1a). A smooth geometric (CAD-like) model is created from this triangulation, which combined with the exterior boundary, defines the volume to be meshed. The meshing in the horizontal is done by either CAMALPaver (from CUBIT) or the new open source library Jaal [4] and then swept and smoothed in the vertical. The bottom, top, and side surfaces are tagged so that boundary conditions can be chosen by the analysis software.



(a) Surface mesh with 20k triangles after decimating (edge contraction, qslim).



(b) A region with observations of acceptable accuracy and a more localized region.

Why implicit?

The evolution of glaciers and ice sheets occurs on multiple time scales and frequently the slower of these scales are of great scientific importance. Most models of ice sheets and other climate systems are based on methods in which crucial components of the physics are treated explicitly. In addition to reducing the accuracy, time splitting errors produced by such methods may radically change steady states or mispredict hysteresis. With no measure of coupled residual, it is difficult to determine when a system has reached steady state rather than just a period of slow evolution. Furthermore, explicit methods must satisfy stability constraints such that the maximum stable time step is mesh- and parameter-dependent, preventing weak scalability. If the resolution is increased, it is not sufficient to simply run on a larger number of processors since more time steps will be required. For non-stiff hyperbolic equations, it is often desirable to maintain time-accuracy of transport phenomena in which case the CFL condition cannot be circumvented and explicit methods are highly appropriate. Stiff hyperbolic, parabolic, and elliptically constrained equations contain time scales that are not of physical interest and the necessity of explicit methods to resolve these scales prevents scalability.

Implicit methods offer the ability to take time steps independent of mesh resolution, only tracking the time scales of interest. Additionally, bifurcation analysis is most effective when the Jacobian evaluated at a steady state is available, allowing, for example, efficient exploration of a branch jump in multi-dimensional parameter space.

Conservative polythermal ice formulation

We solve field equations for total momentum, pressure, and total energy density $(\rho \boldsymbol{u}, p, E) \in W^{1,\mathfrak{p}} \times L^2 \times H^1$:

$$(\rho \boldsymbol{u})_t + \nabla \cdot (\rho \boldsymbol{u} \otimes \boldsymbol{u} - \eta D\boldsymbol{u}_i + p\mathbf{1}) - \rho \boldsymbol{g} = 0 \quad (1a)$$

$$\rho_t + \nabla \cdot \rho \boldsymbol{u} = 0 \quad (1b)$$

$$E_t + \nabla \cdot \left((E+p)\boldsymbol{u} - k_T \nabla T - L(1-\omega)\frac{\rho_i}{\rho}\kappa_\omega \nabla \omega \right) - \eta D\boldsymbol{u}_i : D\boldsymbol{u}_i - \rho \boldsymbol{u} \cdot \boldsymbol{g} = 0 \quad (1c)$$

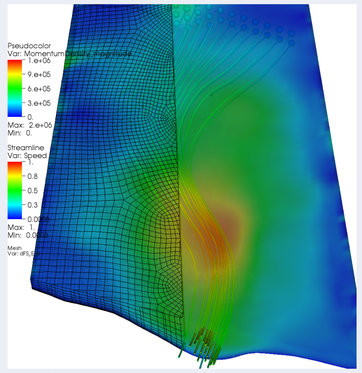


Figure: Computed momentum density field for the ice stream region at Jakobshavn Isbræ.

Constitutive relations are needed for total density ρ ($\mathrm{kg\,m^{-1}}$), ice velocity $\boldsymbol{u}_i$ ($\mathrm{m\,s^{-1}}$), temperature T (K), volumetric moisture fraction (porosity) ω (nondimensional), and viscosity η ($\mathrm{Pa\,s} = \mathrm{kg\,m^{-1}\,s^{-1}}$). The internal energy is decomposed into a contribution toward temperature and a contribution toward moisture fraction using the error function to ensure smooth derivatives near the melting point. Due to this decomposition and the full conservative form, the constitutive relation is generally implicit. If assumptions about small melt fraction are made, the constitutive relation becomes explicit and this formulation reduces to those used by others, e.g. [1].

Newton linearization produces linear systems with the block structure

$$J = \begin{pmatrix} J_{uu} & J_{up} & J_{uE} \\ J_{pu} & 0 & 0 \\ J_{Eu} & J_{Ep} & J_{EE} \end{pmatrix}. \quad (2)$$

The relative stiffness of the blocks must be taken into consideration when designing preconditioners for Krylov methods used to solve with (2) when computing each Newton step or time step. The linear stiffness is essentially contained in the elliptic operator J_{uu} (viscous effects), the velocity-pressure coupling in J_{up}, J_{pu} (made worse by the presence of the hydrostatic mode), the coupling to the energy equation J_{Eu}, and the advection-dominated J_{EE}. Although important to the flow in some regimes, the coupling J_{uE} contributes very little linear stiffness and in practice, can be neglected by the preconditioner without causing significant changes to the spectrum or introduction of non-normality.

Tightly coupled solvers, loosely coupled software

It is desirable to reuse exactly the same "physics" code to run single-physics models, coupled models using semi-implicit methods, fully implicit coupled models using split preconditioners, and fully implicit coupled models using monolithic preconditioners. A generic interface has been added to PETSc [2] and the results on this poster are early client applications. It provides efficient assembly with arbitrary subphysics nesting and parallel decomposition, using a natural interface based on "local submatrices". When the local submatrix interface is used for assembly, subphysics modules can be composed without recompilation into arbitrarily deep hierarchies, the hierarchy is flattened by the library so that performance is not affected. An example use case is the preconditioner for (2),

$$P = \begin{bmatrix} \begin{pmatrix} J_{uu} & J_{up} \\ J_{pu} & 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \begin{pmatrix} J_{Eu} & J_{Ep} \end{pmatrix} & J_{EE} \end{bmatrix} \quad (3)$$

where the inner saddle point problem is a variable-viscosity Stokes problem that is preconditioned using another split.

The matrix format behind the interface can be chosen at run-time and includes a single monolithic matrix or nested pieces intended for use with field-split preconditioning with no memory or scalability penalty. Each nested piece can take take advantage of efficient blocked and symmetric storage formats, offering performance gains of a factor of 2 for sparse matrix kernels and assembly. Matrix-free methods are fully supported and can be used for some or all physics components and inter-physics couplings. When a geometric multigrid hierarchy is available, field-split can be done inside or outside the multigrid cycles.

If the differential algebraic equations for the global transient problem are written in the semidiscrete form

$$g(t, x, \dot{x}) = f(t, x)$$

where x is a vector containing all the discretized state variables, g represents the stiff part of the equation, and f represents the non-stiff part, then a variety of implicit and IMEX time integration methods become available. For example, the same physics code can be used for steady-state analysis using Newton or pseudotransient continuation and for transient analysis using Rosenbrock-W, Runge-Kutta IMEX, or linear multistep methods, all with adaptive error control, with no modification to the source code.

Coupled velocity, surface evolution, and erosion

Here, we solve the hydrostatic equations coupled to surface evolution and erosion,

$$-\nabla \cdot \left[\eta \begin{pmatrix} 4u_x + 2v_y & u_y + v_x & u_z \\ u_y + v_x & 2u_x + 4v_y & v_z \end{pmatrix} \right] + \rho g \nabla s = 0$$

$$h_t + \nabla \cdot \int_b^s (u, v) = 0$$

$$b_t + k_e |(u, v)|^p = 0$$

where $\eta(\gamma) = \frac{B}{2}(\frac{1}{2}\epsilon^2 + \gamma)^{\frac{1-n}{2n}}$ is nonlinear effective viscosity with regularization ϵ and p

$$\gamma = u_x^2 + v_y^2 + u_x v_y + \frac{1}{4}(u_y + v_x)^2 + \frac{1}{4}u_z^2 + \frac{1}{4}v_z^2$$

is the second invariant. The momentum equations are discretized using Q_1 finite elements and the surface evolution equation uses a node-centered upwind finite volume scheme with coupling slopes computed using centered reconstruction.

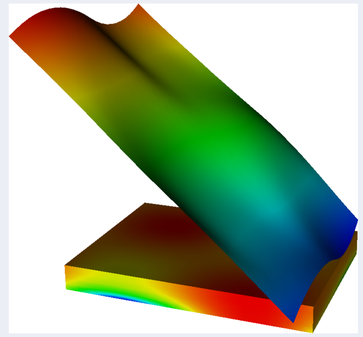


Figure: A steady-state solution for ISMIP-HOM test C at 10km computed in 19 iterations. The elevated surface is exaggerated surface height.

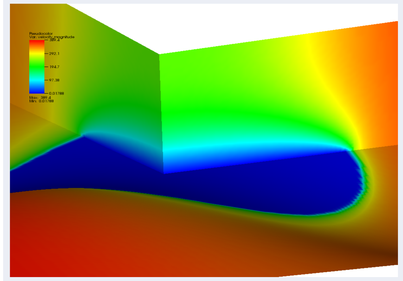


Figure: Shear margin for flow over a smooth bumpy bed with discontinuous sliding parameters and $m = 1/10$ nearly plastic yield model.

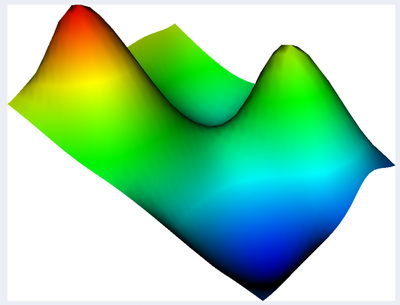
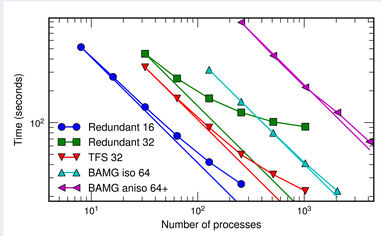
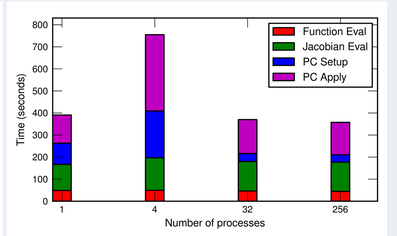


Figure: Bed profile eroded from a flat bed after 300 ka with test C slipperiness perturbation. Time steps are 30 ka at this point in the simulation.

Parallel Scalability on Blue Gene/P



(a) Strong scalability on Shaheen for different problem sizes with different coarse level solvers.



(b) Weak scalability for an entire grid-sequenced solve of a problem with $m = 1/5$ nonlinear sliding with discontinuous coefficients over a bumpy bed. Subdomains of size 64^3.

References

[1] Andy Aschwanden, Ed Bueler, Constantine Khroulev, and Heinz Blatter. An enthalpy formulation for glaciers and ice sheets. *Journal of Glaciology, submitted* 2011.

[2] Satish Balay, Jed Brown, Kris Buschelman, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Barry F. Smith, and Hong Zhang. PETSc Web page. mcs.anl.gov/petsc, 2011.

[3] Jed Brown, Barry F Smith, and Aron Ahmadia. Achieving textbook multigrid efficiency for hydrostatic ice flow. *SIAM Journal on Scientific Computing, submitted* 2011.

[4] Timothy J. Tautges. MeshKit. https://trac.mcs.anl.gov/projects/fathom/wiki/MeshKit, 2011.

This work was supported by the Department of Energy Office of Advanced Scientific Computing Research.