The Big Thaw: Simulating Greenland's Future

John Burkardt
Department of Scientific Computing
Florida State University

....

Computational Mathematics Seminar
Mathematics Department
University of Pittsburgh

...

https://people.sc.fsu.edu/∼jburkardt/presentations/...
climate_2010_pitt.pdf

2pm, 10 December 2010

In recent decades, a slight but steady increase in average world-wide temperature was noticed. This was attributed to the effects of industrialization, reduction in tree cover, increased burning of fossil fuels, the role of carbon dioxide in absorbing greater amounts of solar energy.

Even if the mechanisms are still just plausible hypotheses, the slight but steady rise in temperature has continued. It is natural to ask the straightforward question:

Is anything important likely to happen to the Earth if the temperature continues to increase?

The question being asked is a *modeling question*. In other words, we are not asking *if* the temperature is rising a little each year, or by how much.

We are assuming a certain yearly rise in temperature, and asking if we can produce a plausible climate model which can exhibit the expected results.

Obviously, we would be interested in checking several different estimates for the increase, and if we model the climate, we'd like the actual yearly rise to vary in some statistical way about the estimated average rise.

Modeling the global climate is really hard. While we understand the properties of air, ocean, land and sunshine in general, we have a limited understanding of some common things (such as clouds) and some big, slowly moving things (glaciers and ice sheets).

Introduction - Calving from an Ice Sheet

The melting of a particular glacier might not seem important. But an ice sheet, on the other hand, can be the size of a small continent (Greenland and Antarctica, in particular); the ice can be as much as two miles in thickness. The amount of ice locked up in the Greenland ice sheet is enough, by itself, to raise the ocean level by 20 feet.

In other words, the status of the Greenland ice sheet is signficant.

The ice in Greenland moves slowly, but measurably. The movement of the icesheet is held back by a strong frictional force where the ice sheet rests on bedrock. This is usually a "dry" contact. However, in places where a significant amount of water percolates down to the bedrock, it has the effect of lubricating the interface and unlocking the ice sheet, which then can slide towards the sea.

Introduction - Surface Melt

Introduction - Simulating the Ice Sheets

The goal of a simulation is to take some model of the earth's temperature profile and to simulate the resulting climate over a period of 100 years.

There are already computer programs available which attempt this task, but in order to make any computations, they've had to make numerous short cuts and estimates, and to simply admit that certain parts of the model are not well understood.

The climate model programs have a particularly limited understanding of the behavior and influence of the global ice sheets of Greenland and Antarctica.

The Big Thaw: New Ice Sheet Model

The Fourth Report of the Intergovernmental Panel on Climate Change (2007) declared that the current models and programs for ice sheets did not provide credible predictions; for this reason, the panel was unable to estimate changes in sea level over the coming century.

One goal for the Fifth Report, to appear in 2014, was to substantially improve ice sheet models so that reliable sea level simulations could be produced.

One group working on improvements to the "CISM" (Community Ice Sheet Model) includes the US DOE, in particular groups at Oak Ridge and Los Alamos National Laboratories, headed by Kate Evans and Bill Lipscomb, respectively. Max Gunzburger's group at Florida State has been participating in this effort.

Introduction - Surface Melt

While some programs (for example, **GLIMMER**) are available to model the ice sheets in isolation, it has not been easy to integrate them into the global modeling system.

Moreover, current ice sheet models were designed using what now seem old-fashioned or limited approaches.

Our goal is to use better equations for the mathematical model, a more powerful and flexible approximation scheme for computation, and to add the ability for the ice sheet model to interact with a global climate model that is running simultaneously.

- Introduction
- The Big Picture
- Mathematical Modeling
- The Finite Volume Model
- Moving to Finite Elements
- Conclusion

BIG PICTURE: How to Find Greenland

Greenland has an area of about 2,000,000 km^2.

Ice covers 85% of the surface, to a maximum depth of 3 km.

BIG PICTURE: Greenland and Climate

The purpose of the investigation is to “plug” Greenland into programs that model global climate.

The primary effects that Greenland exerts on the external climate (ocean and atmosphere) include:

- the temperature of the surface of the ice sheet;
- the reflectance of solar radiation from surfaces covered by ice;
- the rate at which ice and meltwater flows into the ocean;
- where the land-based ice sheet extends into the ocean, the ocean and atmosphere become "decoupled";

The external climate’s effect on Greenland can be summarized by

- solar radiation, moderated by cloud cover;
- heat transfer with the atmosphere;
- snowfall, which builds the ice sheet.
- there is an essentially constant geothermal heat flux through the bottom of the ice sheet.

BIG PICTURE: State Variables

The behavior of the ice sheet can be understood in terms of the following state variables:

- the temperature $T(x, y, z)$;
- the thickness $H(x, y, z)$;
- the pressure $P(x, y, z)$;
- the velocity $\vec{V}(x, y, z)$.

We assume that density is essentially constant.

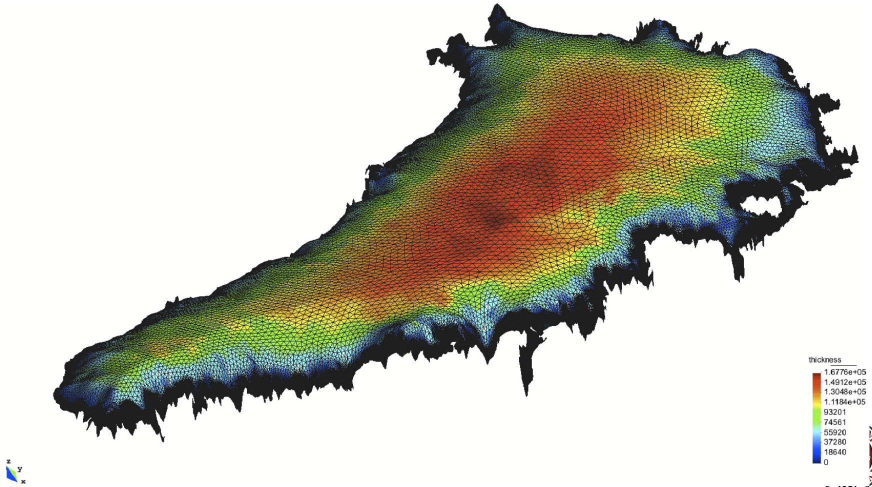
Because we are modeling such a large region, we do not try to study localized features such as cracks and faults in the ice.

BIG PICTURE: Current Resolution 5km x 5km

Current simulations produce values of the state variables on a spatial grid that is horizontally uniform (5km x 5km). This corresponds to a rectangular grid of 301 (East to West) by 561 (North to South) cells by 11 layers $\rightarrow$ 1.5 million nodes.

The needed resolution is 1km x 1km, at least in areas of high ice-sheet velocity; this could be 25 times as many nodes.

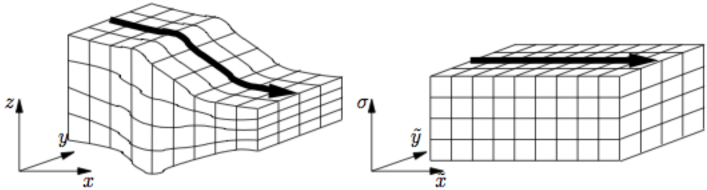
BIG PICTURE: Ice Thickness



BIG PICTURE: Normalizing the Ice Thickness

In the vertical direction, the grid might use 11 proportionally spaced levels from bedrock to ice surface; moreover, at each point (x, y) the vertical direction is scaled to the range [0,1].

Since the ice thickness varies, this means that the grid is *not* orthogonal in the **Z** direction.



This means that equations involving spatial derivatives will have extra terms derived from the geometric distortion.

BIG PICTURE: Long Time and Tight Accuracy

To perform simulations as desired, the model must be able to compute data over a time span of 100 years, using a time step that may be on the order of 1 year.

A continuity equation will be imposed on the state variables, and one requirement for the simulation is that the continuity equation be exactly satisfied, to machine roundoff; that is, the mass at the beginning, plus snowfall, minus calving, must equal the mass at the end of the century, to 16 decimal places, with a similar requirement on energy.

MODEL: The Thickness Equation

The thickness $H(x, y)$ of the ice changes with snowfall and motion of the ice sheet:

$$\frac{\partial H}{\partial t} = H_{\text{flux}} - \nabla \cdot \int_z \begin{bmatrix} u \\ v \end{bmatrix} dz$$

- H_{flux} is the flux due to snowfall;

Assuming the ice always flows outward, the boundary condition for thickness can be taken as a homogeneous Neumann condition.

MODEL: The Temperature Equation

The evolution equation for temperature T:

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial z}(\kappa \frac{\partial T}{\partial z}) - \rho c(\vec{u} \cdot \nabla T) + 2\dot{\varepsilon} :: \sigma$$

- c is the heat capacity;
- ρ the (constant) density;
- κ is the thermal conductance;
- $\dot{\varepsilon}$ is the strain rate tensor;
- σ is the stress tensor.

MODEL: The Temperature Boundary Conditions

Ice at the surface is assigned the mean annual surface temperature.

Ice at the bedrock boundary is subject to geothermal flux G and heat generated by sliding friction:

$$-k\nabla T \cdot \vec{n} = G + \vec{t} \cdot \vec{u}$$

MODEL: Stokes Equations for Velocity

The ice sheet can be regarded as a very viscous liquid with a tiny Reynolds number. A quasi-static assumption means that we can also drop the time derivative. What follows is a form of Stokes equation for velocity $\vec{U}$ with a variable viscosity:

$$\begin{aligned} -\nabla B(T)\,|\dot{\varepsilon_e}|\,(\nabla U + (\nabla U)'))/2 + \nabla P &= \begin{bmatrix} 0 \\ 0 \\ \rho g \end{bmatrix} \\ \nabla \cdot U &= 0 \end{aligned}$$

- ρ is the density, g the unit gravitational force;
- $|\dot{\varepsilon_e}|$ is the norm of the strain rate tensor;
- $B(T)$ is a constitutive coefficient.

MODEL: Simplified Velocity Equations

Because the ice flow tends to be predominantly in the horizontal plane, it is possible to simply the velocity state equations even further. Simplified models produce smaller sets of equations that can be solved faster.

Three levels of simplification include:

- *The 1st order model*: simplifying the momentum balance in the **Z** direction, and incorporating this assumption into the **X** and **Y** force balance equations;
- *"Shallow Ice Equations"*: assuming that pressure is strictly a function of height, and has the form $P(x, y, z) = \rho g(H - z)$ where $H(x, y)$ is the top of the ice sheet; it is suitable for slow-sliding regions;
- *"Shallow Shelf Equations"* are a simplification of the 1st order model suitable for fast-sliding regions.

- Introduction
- The Big Picture
- Mathematical Modeling
- **The Finite Volume Model**
- Moving to Finite Elements
- Conclusion

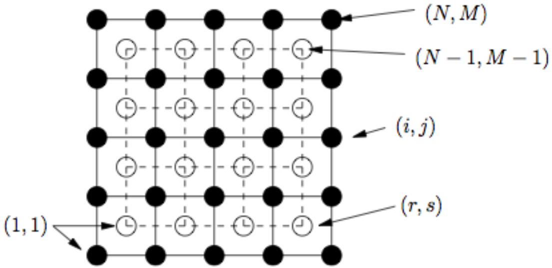
Finite Volumes: A Staggered Grid

The horizontal domain is approximated by a pair of grids, sometimes referred to as the (i,j) and (r,s) grids.

Horizontal velocities U and V are assigned to (r,s) nodes.

Vertical velocities W, ice thickness H, and temperature T, are assigned to (i,j) nodes.

Copies of these 2D grids are generated for each ice sheet layer.



Finite Volumes: Gradients

Gradients of (r,s) objects are assigned to (i,j) nodes:

$$(\frac{\partial u}{\partial x})_{i,j} \approx \frac{u_{r,s} + u_{r,s-1} - u_{r-1,s} - u_{r-1,s-1}}{2\Delta x}$$

A conservation law, written using this kind of scheme for gradients, will correctly conserve the quantity of interest.

Finite Volume: Forming Discrete System

Using the staggered meshes to form approximations to derivatives, and a constant stepsize in time, it is possible to discretize the equations for temperature, ice thickness, and velocity.

Starting from some initial condition, the solution is advanced by timesteps.

Because the number of variables is so large, and the system of equations includes nonlinearities, a direct solution is not attempted. Instead, an iterative scheme is employed. At each step of the nonlinear iteration, several linear systems must be solved. Each of these systems is also solved iteratively, using the **SLAP** sparse linear algebra package.

Finite Volume: Solving Discrete System

The system size is reduced by uncoupling the state variable equations, and by uncoupling the ice sheet layers.

In particular, the iteration proceeds as follows:

- update the horizontal velocities using the first order equations;
- back out the vertical velocities, to get $\vec{U}(x, y, z)$;
- for each point on the ice sheet bottom layer, integrate the ice thickness equation to update $H(x, y)$;
- by ignoring the effects of horizontal dissipation, integrate the temperature equation from bottom layer to top to get $T(x, y, z)$;

Although information seems to flow only upwards, there are coupling coefficients that relate adjacent ice sheet layers, so that information can also travel downwards as the iteration proceeds.

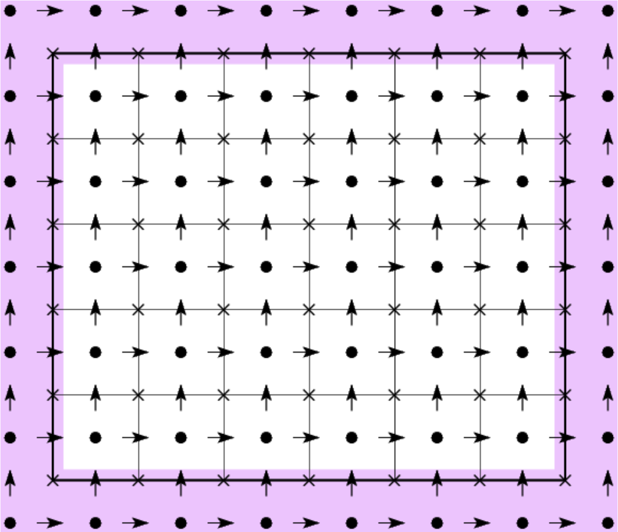
Finite Volume: Parallelism

For each subsystem being solved, the variables are laid out on a horizontal mesh.

To evaluate the equations defined at a node, it is typically necessary to access data at neighbor nodes to the east, west, north and south, as well as the lower and upper layers.

A parallel implementation which divides up the rectangular grid into subrectangles must enable each subrectangle to obtain some information from adjacent subrectangles. This is done by augmenting each partial grid with a layer of "ghost cells" or "halo cells", which are available as information, and do not need to be updated.

Finite Volume: Ghost Cells



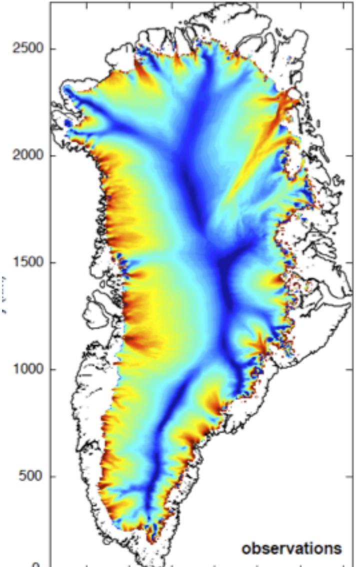
Finite Elements: Advantages

The contribution of our group has been to implement a revised treatment of the calculation using a grid adapted to the geometry and known ice behavior (Lili Ju), and a reformulation of the state equations using finite elements (Mauro Perego).

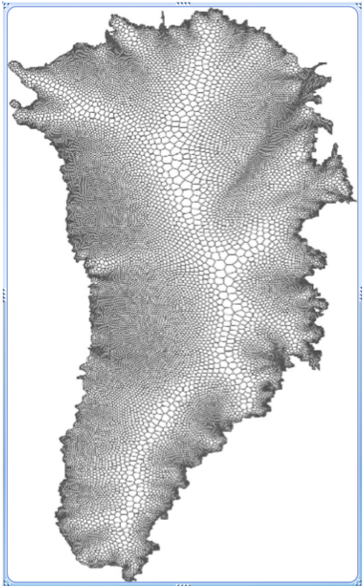
By abandoning the rectangular grid:

- we no longer waste time modeling bits of the ocean;
- we can more accurately follow the coastline of Greenland, and any other geometric objects;
- we can provide a refined mesh in areas of Greenland where the ice sheet velocity is known to be high;
- we can use a grid that smoothly interfaces with grids employed by global climate modeling programs, so that data from one program can be used by the other.

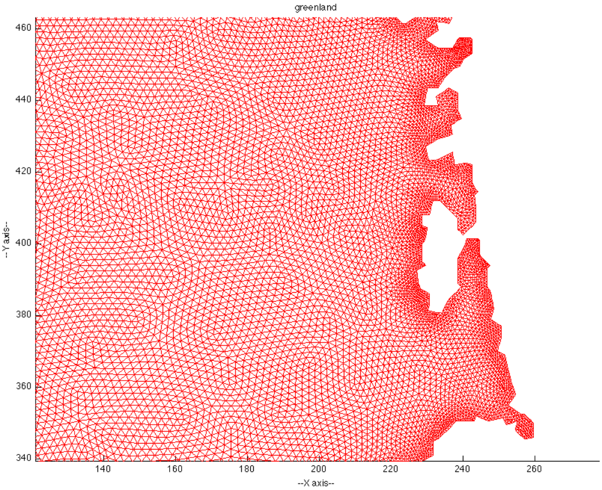
Finite Elements: Observed Ice Sheet Velocity



Finite Elements: a Sample Grid Adapted to Velocity

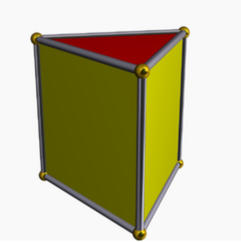


Finite Elements: Detail of Coastline Grid



Finite Elements: Using Triangular Prisms for 3D Elements

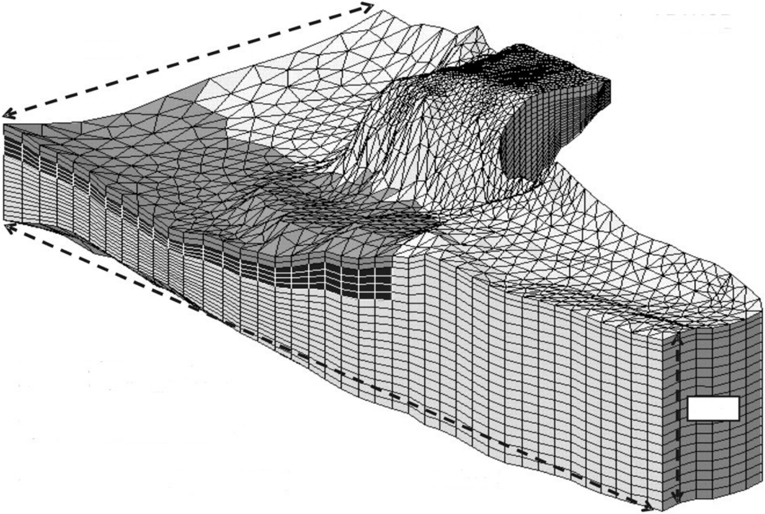
While tetrahedral elements can also be used, triangular prisms allow us to preserve the modeling of particular layers of the ice sheet.



An (essentially) linear element is shown here. To achieve quadratic approximation, midside nodes are added; higher order approximation is also easily achieved.

Finite Elements: Example of a Triangular Prism Grid

This is *not* the grid for an ice sheet, but it suggests the layered nature of the triangular prism grid.



Finite Elements: Example of a Triangular Prism Grid

All the state equations can be discretized using the same mesh.

The parallel assembly of the system matrix does not require any communication between processors at all.

The parallel solution of the linear systems arising in the nonlinear iteration at each time step only requires the use of an appropriate library solver.

In fact, the solution of the entire nonlinear system can also be done in parallel, using an off-the-shelf library solver.

For this implementation, we are using the **NOX** package from **Trilinos**;

NOX is an object-oriented C++ library for large nonlinear systems. It implements Newton-based globalization techniques including line search and trust region algortihms. NOX defines interfaces to user codes through the abstract group and vector pure virtual classes.

Thus, the user can supply the underluying linear algebra solver, or take advantage of the native **Epetra** package. With **Epetra** the user only needs to evaluate the residual equation $F(x) = 0$ for a given x;

To improve performance, the user can supply preconditioning or jacobian information.

Finite Elements: Trilinos EPETRA Parallel Linear Algebra

Epetra contains classes for distributed sparse and dense matrices and vectors;

It provides a flexible and powerful data redistribution capability for load balancing and scalability of linear algebra algorithms without the user needing any special knowledge about distributed object.

Epetra provides a parallel machine interface that allows users to write generate parallel functionality without specifically using any particular parallel library.

Finite Elements: LIFEV Finite Element Library

To increase the order of approximation of a finite difference or finite volume code can require an extensive rewrite.

For a finite element code, the order of approximation and even the form of the equations are easily changed in a way that does not obviously affect the main user code. Instead, these choices are implemented in separate code.

The **LifeV** library is our choice for implementing the finite element approximation to the state equations.

Conclusion: A Better Grid

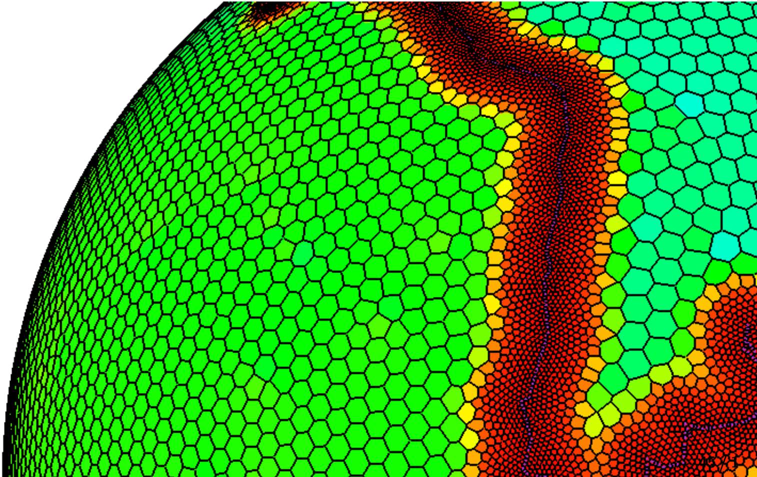
By dropping the uniform rectangular grid and moving to an adaptive mesh:

- we reduce the number of wasted cells;
- we can better conform to the geometry of the region;
- we can choose a weighting function (such as observed ice velocity) to vary the fineness of the mesh;
- we can refine the mesh near the coasts for better interaction with other simulation packages.

This allows us to achieve the desired 1 km x 1km resolution in areas of high ice sheet velocity.

Conclusion: Grid interfaces

The grid is flexible, and can be refined at the coastline so it can exchange more detailed information with a separate program modeling the ocean.

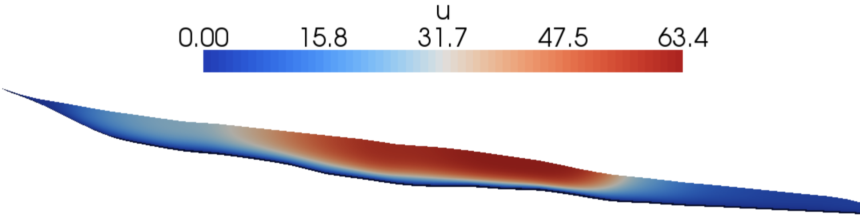


Conclusion: Using Finite Elements

By formulating the problem using finite elements:

- we are able to use the adaptive mesh;
- increasing the approximation power only requires changing a parameter;
- approximated state variables can be evaluated anywhere;
- the nonlinear solution can be handed off to an external library;
- the parallelism can be handed off to an external library.

Conclusion: Arolla Glacier Test Case



Conclusion: Using TRILINOS Package

By using TRILINOS and EPETRA for solving $F(X) = 0$ and $A * x = b$:

- we are no longer responsible for the housekeeping details required when implementing a parallel code;
- we guarantee good parallel performance on a wide range of configurations
- we have access to a variety of high-quality linear and nonlinear solvers with a uniform interface.

Conclusion

Our challenges include:

- implementing the temperature and thickness equations in finite element form (right now, we are only doing the velocities this way);
- converting the routines in LifeV from C++ to FORTRAN90, because the climate community insists on a uniform language;
- matching the very tight tolerances ($\sim$ machine precision) on conservation of mass and energy over the 100 year simulation cycles;
- finishing the process of verification, documentation, and publication before the end of 2012, after which no new input will be accepted for the Intergovernmental Panel on Climate Change (IPCC) Assessment Report 5 to be published in 2014.

Conclusion: Colleagues

- FSU: Max Gunzburger, Mauro Perego (finite elements);
- USC: Lili Ju (gridding);
- NYU: Jean-Francois Lemieux;
- Oak Ridge; Kate Evans, Jeff Nichols, Pat Worley;
- Los Alamos; Bill Lipscomb, Steve Price, Todd Ringler, Xyler Asay-Davis, Dana Knoll;
- Sandia: Andy Salinger (Trilinos);