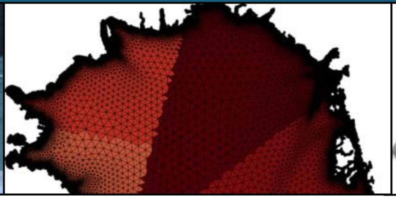
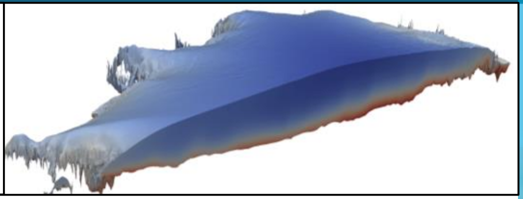
Sandia National Laboratories

SAND2021-2328C

Improving Large-Scale PDE-Constrained Optimization for Ice Sheet Initialization

CCR Center for Computing Research





Mauro Perego

Collaborators:

L. Bertagna, M. Carlson, K. Liégeois S. Rajamanickam,
I. Tezaur, J. Watkins (SNL)
T. Hillebrand, M. Hoffman, S. Price (LANL)
A. Heinlein (U. Stuttgart, Germany)

SciDAC Scientific Discovery through Advanced Computing

U.S. Department of Energy ENERGY NNSA National Nuclear Security Administration

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Talk Outline

- Overview of DOE programs impacting Land Ice modeling
- Brief motivation and introduction to ice sheet models
- Ice sheet initialization
- Improvements enabled by FASTMath institute:
  - Performance speed-up in assembly phase
  - Novel preconditioner for thermo-mechanical problem
  - Newton-Krylov approaches for PDE-constrained optimization

DOE Land Ice Modeling Efforts

ASCR
Advanced Scientific Computing Research

BER
Biological and Environmental Research

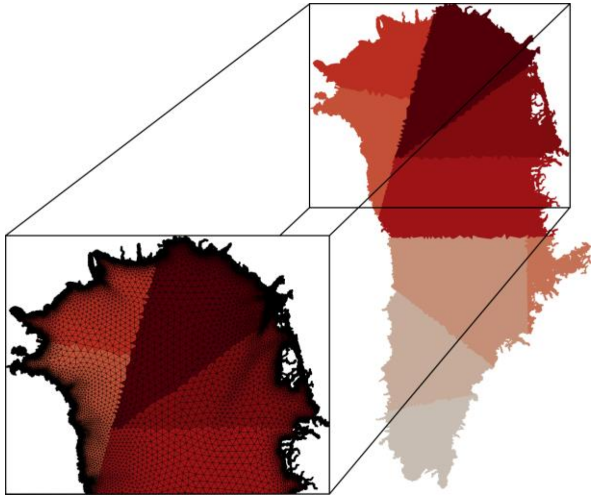
FASTMath
(applied and computational mathematics)

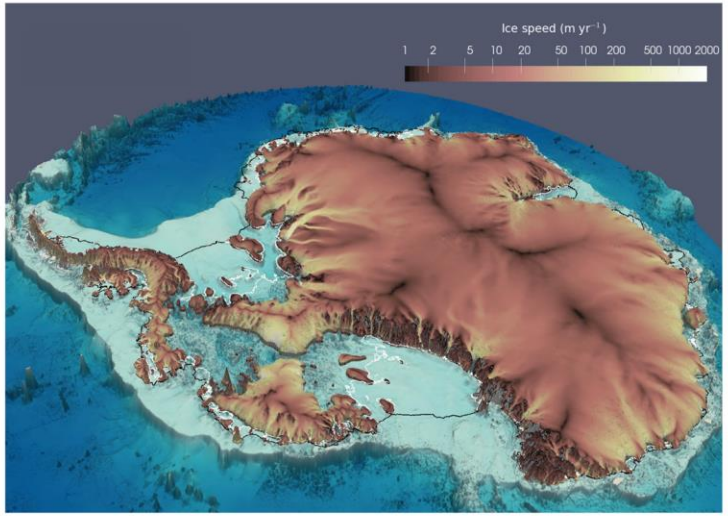
RAPIDS
(computer science and data challenges)

ProSPect
Probabilistic Sea level Projection from Ice Sheet and Earth System Models

E3SM
Energy Exascale Earth System Model

SciDAC
Scientific Discovery through Advanced Computing



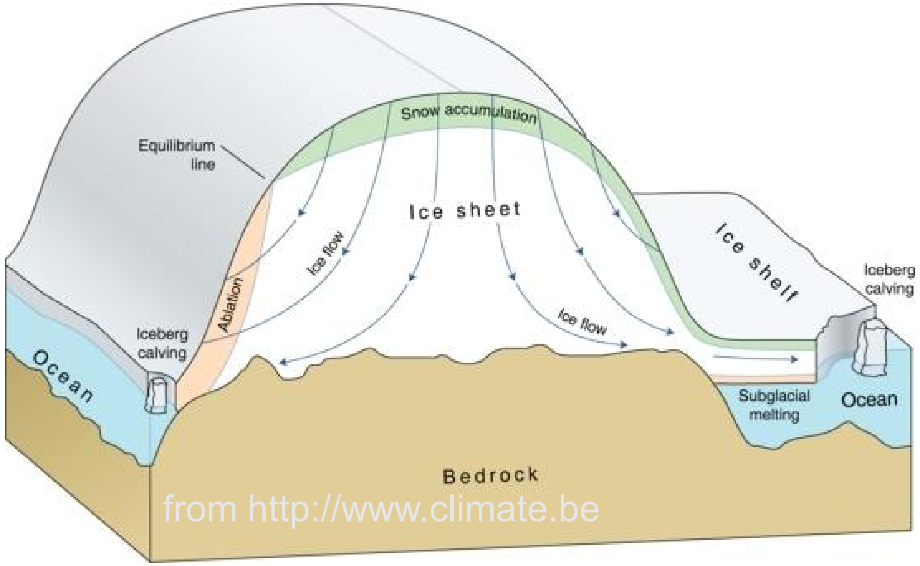


Brief Motivation an basic physics

- Modeling ice sheets (Greenland and Antarctica) dynamics is essential to provide estimates for sea-level rise in next decades to centuries.

- Ice behaves like a very viscous shear-thinning fluid (similar to lava flow) driven by gravity.

- Several unknown or poorly known parameters (e.g. basal friction, bed topography) and processes (calving laws, basal hydrology)



Model: Ice velocity equations

Stokes equations:

$$\begin{cases} -\nabla \cdot \sigma = \rho \mathbf{g} \\ \nabla \cdot \mathbf{u} = 0 \end{cases}$$

Stress tensor:

$$\sigma = 2\mu \mathbf{D} - pI, \qquad \mathbf{D}_{ij}(\mathbf{u}) = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$

Ice viscosity (dependent on temperature):

$$\mu = \frac{1}{2} A(T) \, |\mathbf{D}(\mathbf{u})|^{\frac{1}{n}-1}, \quad n \geq 1, \quad (\text{tipically } n \simeq 3)$$

In this work we use a simplification of Stokes equations, called **First Order** equations, obtained by scaling arguments given the shallow nature of the ice sheets and using hydrostatic pressure.

Model: Ice velocity equations

Stokes equations:

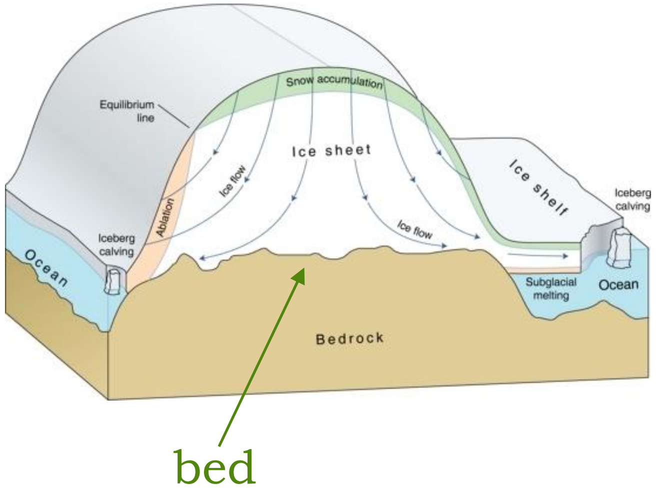
$$\begin{cases} -\nabla \cdot \sigma = \rho \mathbf{g} \\ \nabla \cdot \mathbf{u} = 0 \end{cases}$$

Sliding boundary condition at ice bed:

$$\begin{cases} \mathbf{u} \cdot \mathbf{n} = 0, \quad \text{(impenetrablity)} \\ (\sigma \mathbf{n})_{||} = \beta \mathbf{u} \end{cases}$$

Free slip: $\beta = 0$

No slip: $\beta = \infty$



Model: Temperature equation

Heat equation (for cold ice):

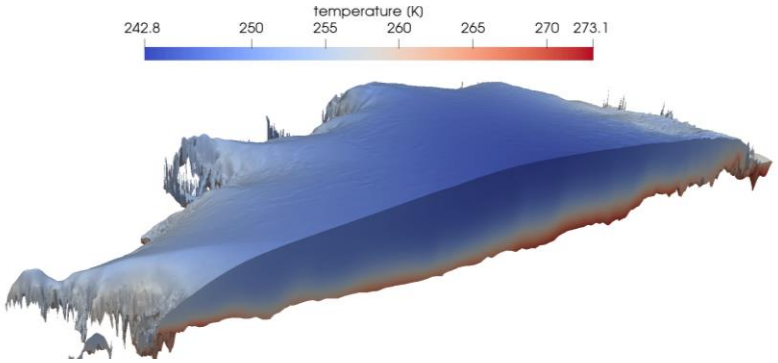
$$\rho c\, \partial_t T + \nabla \cdot (k \nabla T) + \rho c\, \mathbf{u} \cdot \nabla T = 4\mu |D(\mathbf{u})|^2$$

conductivity (k), heat capacity (c), dissipation heating ($4\mu |D(\mathbf{u})|^2$)

Boundary condition at the ice bed (includes melting and refreezing):

$$\mathrm{m} = \mathrm{G} + \beta |\mathbf{u}|^2 - k \nabla T \cdot \mathbf{n}$$

melting rate (m), geothermal heat flux (G), frictional heating ($\beta |\mathbf{u}|^2$), temperature flux ($k \nabla T \cdot \mathbf{n}$)



In this work we use a enthalpy formulation that accounts for temperate ice as well.

Ice sheet initialization

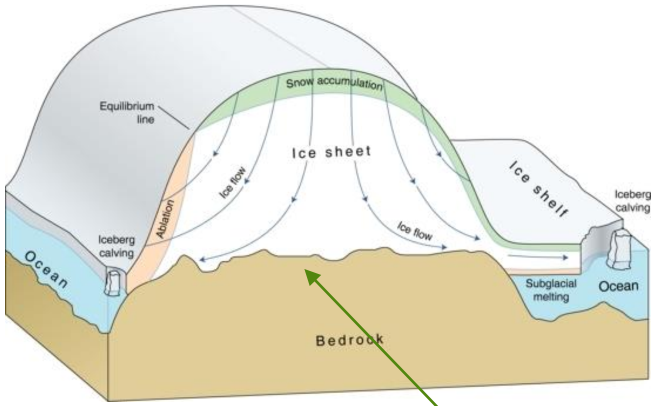
Goal: Find the initial/present-day thermo-mechanical state of the ice sheet and estimate the unknown/poorly known model parameters, by matching observations

Approach: **PDE-constrained optimization**

Find basal friction coefficient β that minimizes the mismatch with surface velocity:

$$\min_{\beta} J(\beta) = \int_{\Omega} \frac{|u - u_{obs}|^2}{\sigma^2} + R(\beta)$$

Subject to the coupled velocity/temperature problem



unknown sliding parameter β

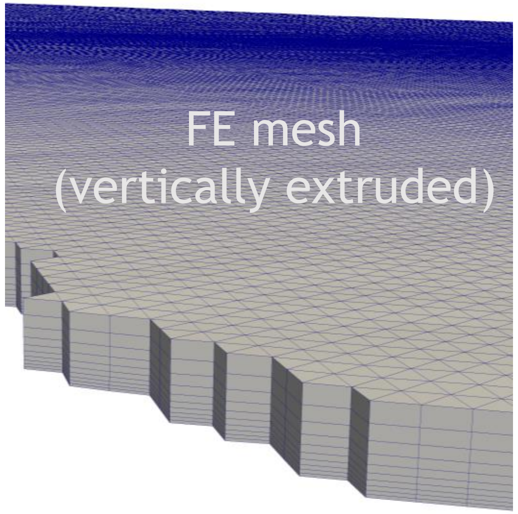
Software Requirements

- Large Scale optimization library (ROL), featuring gradient-based methods (ROL)
- Computation of gradients of the PDE residual and the loss functional w.r.t. the solution and the parameters. **Automatic Differentiation** is crucial for complex physics
- Faster, more robust methods available using **Hessian** (second derivatives)

RAPID OPTIMIZATION LIBRARY

Software: MPAS-Albany Land Ice model (MALI)

ALGORITHM	SOFTWARE TOOLS
Linear Finite Elements on tets/prisms	Albany Land Ice
Optimization	ROL
Nonlinear solver (Newton method)	NOX
Krylov linear solvers/Prec	Belos/MueLu, Belos/FROSch
Automatic differentiation	Sacado



MPAS (Model for Prediction Across Scales): *Fortran*, **finite volumes** library, conservative Lagrangian schemes for advecting tracers (evolution of ice thickness)
Albany Land Ice: *C++* finite element library built on top of **Trilinos** achieving performance portability through ***Kokkos*** programming model. Provides large scale PDE constrained optimization capabilities

References:

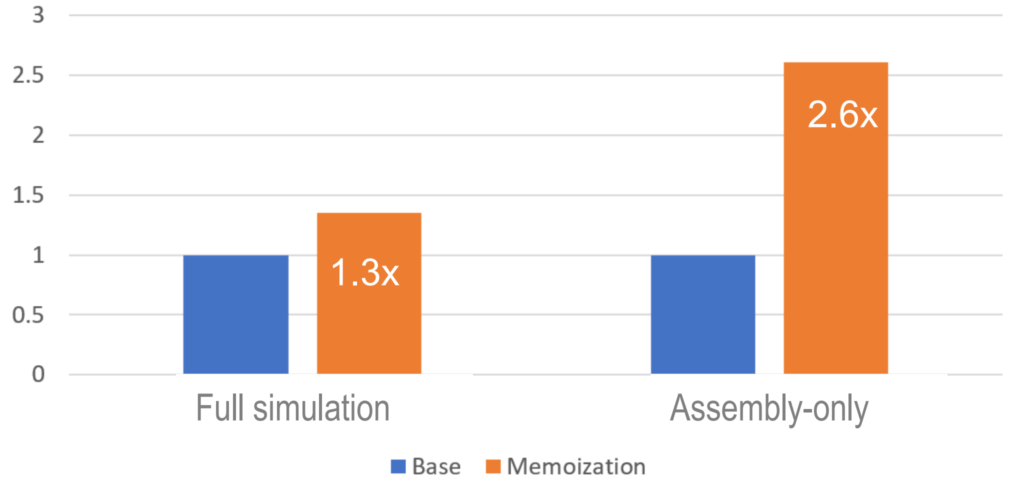
Hoffman, et al. GMD, 2018
Tuminaro, Perego, Tezaur, Salinger, Price, SISC, 2016.
Tezaur, Perego, Salinger, Tuminaro, Price, Hoffman, GMD, 2015
Perego, Price, Stadler, JGR, 2014

MPAS
Model for Prediction Across Scales

Albany

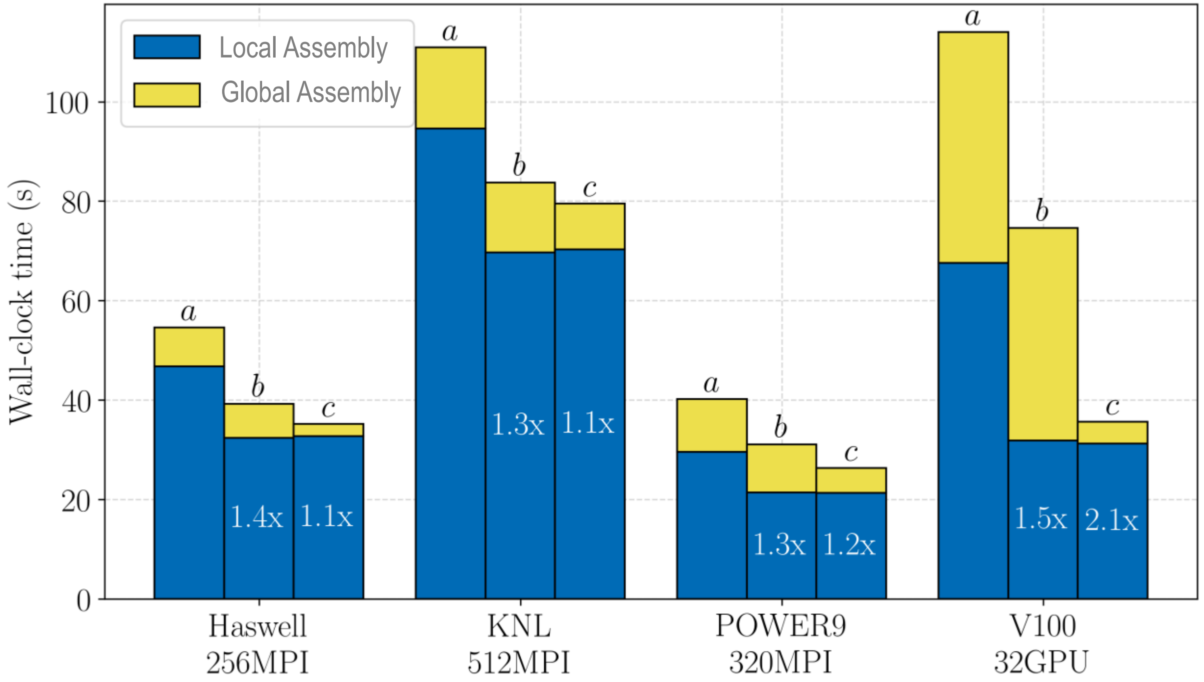
Performance improvements on CPUs
(Greenland glacier, velocity solver)

Improvements to initialization problem due to **memoization**:



AIS initialization 10 iterations
(HSW, 128 nodes, 4096 procs)

Combined improvements due to **memoization** and **improved storage of FE matrix** (Tpetra FE Crs Matrix)

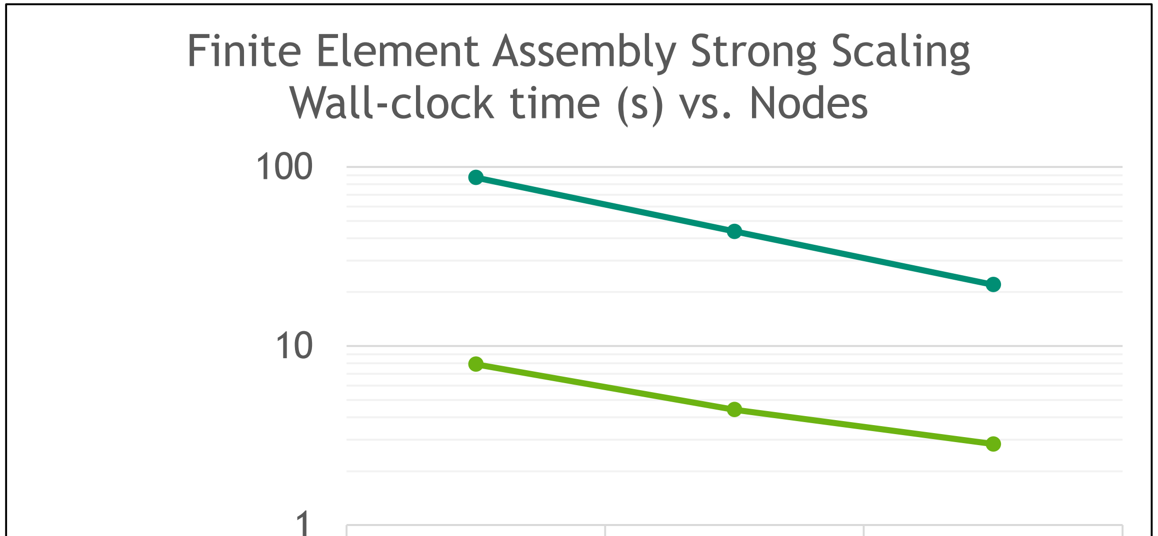


Work/Simulations **by J. Watkins** (see MS235 on Thursday, 10:25am) and **L. Bertagna**
Support: CMDV-SM/FASTMath/ProSPect

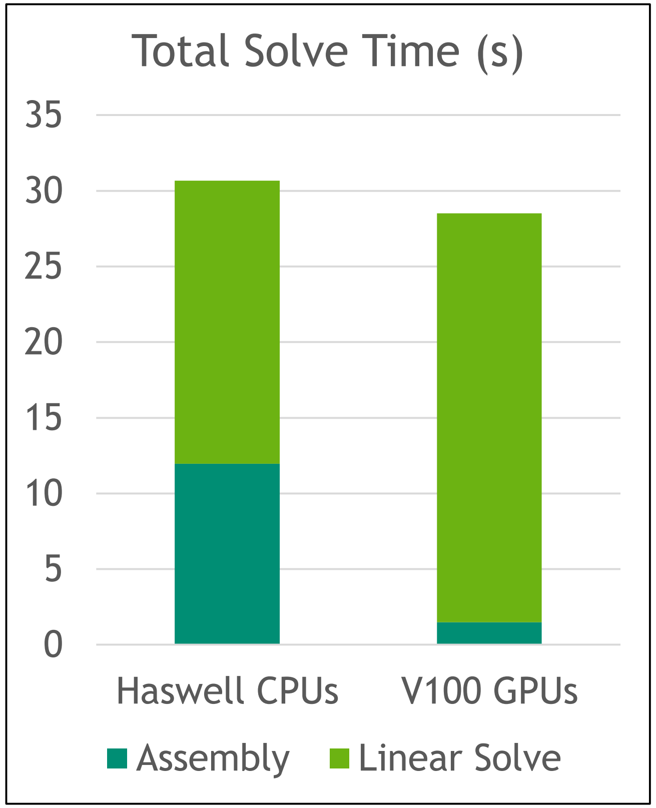
Assembly speed up on GPUs
(Thwaites glacier, velocity solver)

Haswell Node: 32 CPU cores/node
V100 Node: 4 GPUs/node

Up to **11x** speedup for assembly on GPUS.
Time to work on linear solvers!



	4	8	16
Haswell CPUs	87.2051	43.6626	22.0101
V100 GPUs	7.89909	4.41412	2.8328
Speedup	11.03989194	9.891575218	7.769733126
DoFs/GPU	87230	43615	21807



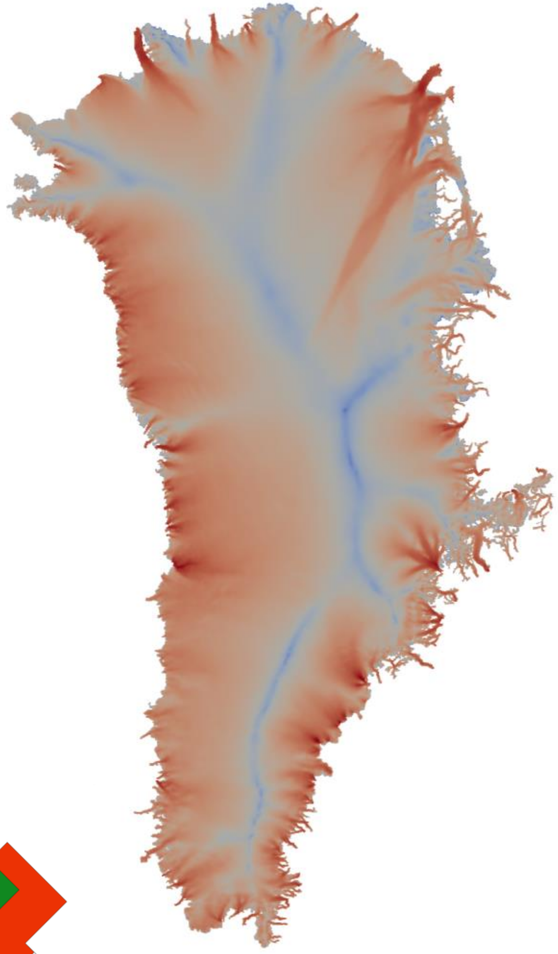
Work/Simulations **by J. Watkins** (see MS235 on Thursday, 10:25am) and **M.Carlson**
Support: CMDV-SM/FASTMath/ProSPect

Two-level Schwarz preconditioner (FROSch)
(Greenland Ice sheet, coupled velocity/temperature problem)

Challenge: our "workhorse" multigrid preconditioner does not work out of the box for the coupled problem.
Solution: FROSch (Fast and Robust Overlapping Schwarz)

Single-level Schwarz preconditioner vs Two-levels Schwarz preconditioner

MPI ranks	avg. #iters		avg. setup time		avg. solve time	
512	48.7	45.0	11.3s	10.53s	5.41 s	5.36s
1024	61.9	54.3	5.29s	4.59s	4.75s	4.31s
2048	89.9	59.1	2.52s	2.32s	5.70s	3.99
4096	116.1	78.7	1.17s	1.37s	3.68	3.30



- The two-level preconditioner does better in terms of number of iterations but it is almost equivalent in term of CPU time to the one-level one.
- Time is dominated by direct solves of the sub-problems and the coarse solve.

Simulations by **A. Heinlein**
Support: CMDV-SM/FASTMath/ProSPect
Resources: Cori (NERSC)
Reference: Heinlein, Perego, Rajamanickam, arXiv, 2021

Ice sheet initialization

Hessian computation using automatic differentiation (using Sacado package)

Capabilities required for Newton-Krylov optimization methods:

Hessian of residual $\boldsymbol{f}$ dotted with the Lagrange multiplier $\boldsymbol{\lambda}$ in the direction $\boldsymbol{v}$:

$$\partial_{\boldsymbol{uu}}(\boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{u},\boldsymbol{p}))\,\boldsymbol{v}, \quad \partial_{\boldsymbol{up}}(\boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{u},\boldsymbol{p}))\,\boldsymbol{v},$$
$$\partial_{\boldsymbol{pu}}(\boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{u},\boldsymbol{p}))\,\boldsymbol{v}, \quad \partial_{\boldsymbol{pp}}(\boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{u},\boldsymbol{p}))\,\boldsymbol{v}$$

Hessian of loss functional J in the direction $\boldsymbol{v}$:

$$\partial_{\boldsymbol{uu}}\, J(\boldsymbol{u},\boldsymbol{p})\,\boldsymbol{v}, \quad \partial_{\boldsymbol{up}}\, J(\boldsymbol{u},\boldsymbol{p})\,\boldsymbol{v},$$
$$\partial_{\boldsymbol{pu}}\, J(\boldsymbol{u},\boldsymbol{p})\,\boldsymbol{v}, \quad \partial_{\boldsymbol{pp}}\, J(\boldsymbol{u},\boldsymbol{p})\,\boldsymbol{v},$$

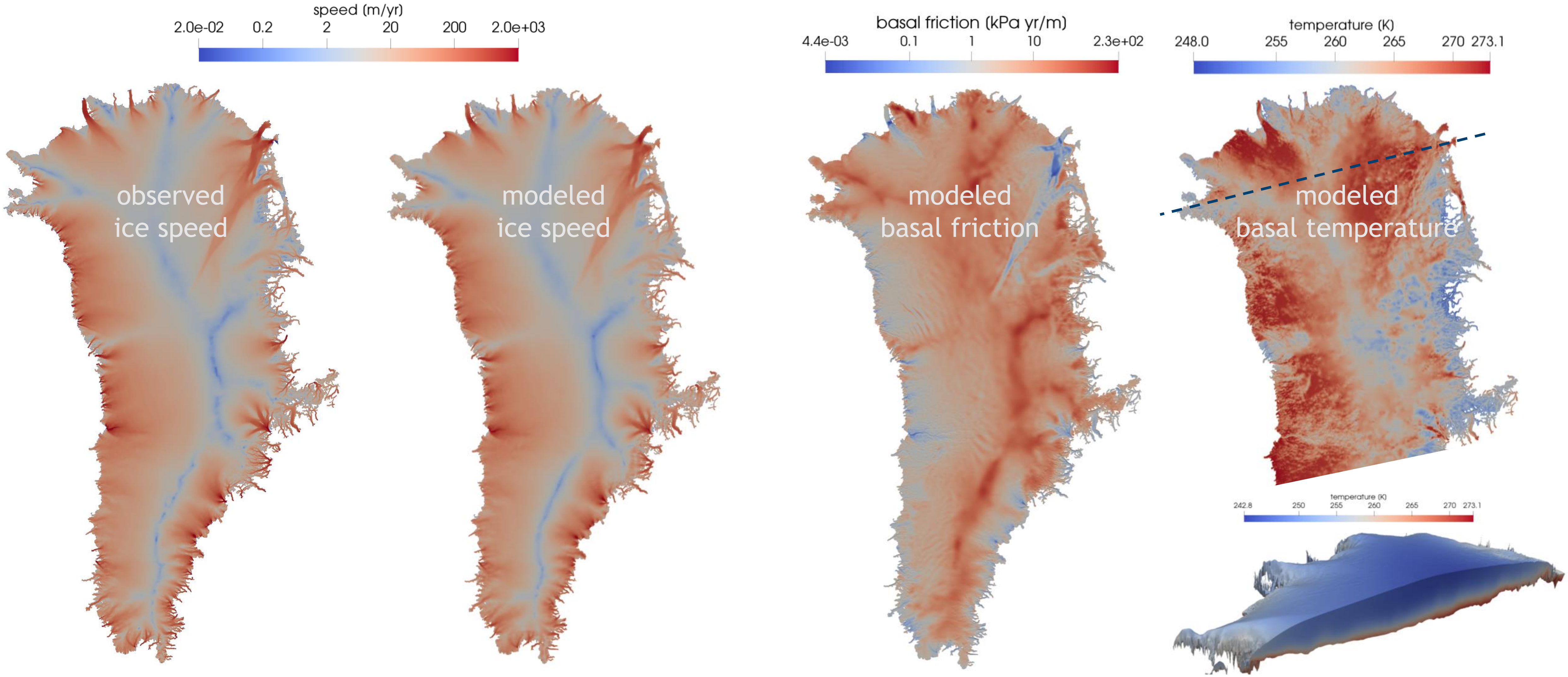
Computed w/ **automatic differentiation**, based on the formula: $\partial_{\boldsymbol{pp}}\, J(\boldsymbol{p})\,\boldsymbol{v} = \partial_r \left(\partial_{\boldsymbol{p}}\, J(\boldsymbol{p} + r\,\boldsymbol{v})\right)\Big|_{r=0}$

We also provide the full matrix $\partial_{\boldsymbol{pp}}\, J$, that can be used to initialize reduced-space optimization methods like BFGS, and the *reduced* Hessian-vector product.

Work by **K. Liégeois**
Support: FASTMath/ProSPect

See T. Hartland talk on Thursday, MS270 at 2:55pm for different application of the reduced Hessian

Thermo-mechanical initialization of Greenland ice sheet



300K parameters, 14M unknowns. Initialization: ~10 hours on 8k nodes on NERSC Cori (Haswell)

Support: FASTMath/E3SM/ProSPect

MS270 on Thursday targets ice sheet modeling