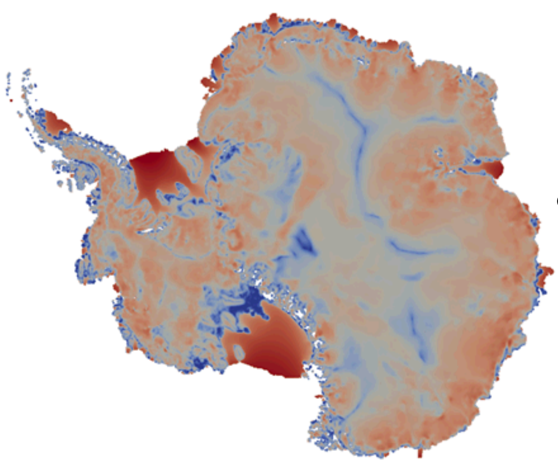
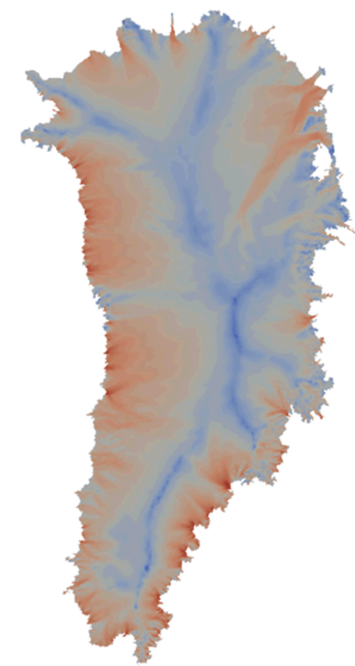
SAND2015-6419C

Albany/FELIX: A Robust & Scalable *Trilinos*-Based Finite-Element Ice Flow Dycore Built for Advanced Architectures & Analysis

I. Tezaur[1], A. Salinger[1], M. Perego[1], J. Jakeman[1], M. Eldred[1], I. Demeshko[1], R. Tuminaro[1], S. Price[2]

[1] Sandia National Laboratories
Livermore, CA and Albuquerque, NM, USA
[2] Los Alamos National Laboratory
Los Alamos, NM, USA





8th International Congress on Industrial and Applied Mathematics (ICIAM 2015)
August 10-14, 2015
Beijing, China

SAND2015-XXXXX

Sandia National Laboratories

Outline

Albany/FELIX = new land-ice solver with ***next-generation*** capabilities.

- **Overview:** the PISCEES project and the First Order (FO) Stokes model.
- The ***Albany/FELIX*** First-Order (FO) Stokes diagnostic solver.
- ***Coupling*** of *Albany/FELIX* to the ***CISM*** and ***MPAS*** codes for prognostic simulations of the ice sheet evolution.
- **Uncertainty Quantification (UQ):** Bayesian calibration and forward propagation of uncertainty.
- **Performance portability.**
- **Summary** and ongoing work.
- **Questions?**

Outline

Albany/FELIX = new land-ice solver with ***next-generation*** capabilities.

- **Overview:** the PISCEES project and the First Order (FO) Stokes model.
- The ***Albany/FELIX*** First-Order (FO) Stokes diagnostic solver.
- **Coupling** of *Albany/FELIX* to the ***CISM*** and ***MPAS*** codes for prognostic simulations of the ice sheet evolution.
- **Uncertainty Quantification (UQ):** Bayesian calibration and forward propagation of uncertainty.
- **Performance portability.**
- **Summary** and ongoing work.
- **Questions?**

The PISCEES Project and the *Albany/FELIX* Solver

SciDAC Scientific Discovery through Advanced Computing

"PISCEES" = Predicting Ice Sheet Climate & Evolution at Extreme Scales
5 year project funded by SciDAC, which began in June 2012

Sandia's Role in the PISCEES Project: to **develop** and **support** a robust and scalable land ice solver based on the "First-Order" (FO) Stokes physics

- Steady-state stress-velocity solver based on FO Stokes physics is known as ***Albany/FELIX**.***
- **Requirements for *Albany/FELIX*:**
  - Scalable, fast, robust.
  - Dynamical core (dycore) when coupled to codes that solve thickness and temperature evolution equations (*CISM/MPAS LI* codes).
  - Advanced analysis capabilities (adjoint-based deterministic inversion, Bayesian calibration, UQ, sensitivity analysis).
  - Performance-portability.

This talk (brace covering the last two items)

Dycore will provide actionable predictions of 21st century sea-level rise (including uncertainty).

**FELIX*="Finite Elements for Land Ice eXperiments"

Albany/FELIX Solver (steady):
Ice Sheet PDEs (First Order Stokes)
(stress-velocity solve)

⇄

***CISM/MPAS* Land Ice Codes (dynamic):**
Ice Sheet Evolution PDEs
(thickness, temperature evolution)

Sandia National Laboratories

The First-Order Stokes Model for Ice Sheets & Glaciers

- Ice sheet dynamics are given by the ***"First-Order" Stokes PDEs***: approximation* to viscous incompressible ***quasi-static*** Stokes flow with power-law viscosity.

$$\begin{cases} -\nabla \cdot (2\mu \dot{\boldsymbol{\epsilon}}_1) = -\rho g \dfrac{\partial s}{\partial x} \\ -\nabla \cdot (2\mu \dot{\boldsymbol{\epsilon}}_2) = -\rho g \dfrac{\partial s}{\partial y} \end{cases}, \quad \text{in } \Omega$$

$$\dot{\boldsymbol{\epsilon}}_1^T = (2\dot{\epsilon}_{11} + \dot{\epsilon}_{22}, \dot{\epsilon}_{12}, \dot{\epsilon}_{13})$$
$$\dot{\boldsymbol{\epsilon}}_2^T = (2\dot{\epsilon}_{12}, \dot{\epsilon}_{11} + 2\dot{\epsilon}_{22}, \dot{\epsilon}_{23})$$
$$\dot{\epsilon}_{\text{ij}} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial uj}{\partial xi}\right)$$

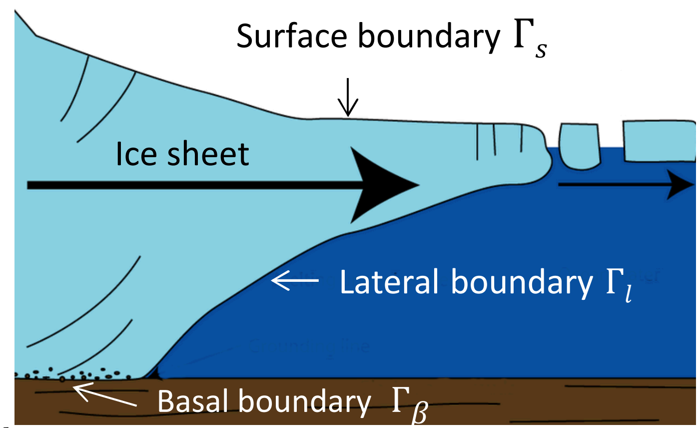
- Viscosity μ is nonlinear function given by "***Glen's law***":

$$\mu = \frac{1}{2} A^{-\frac{1}{n}} \left(\frac{1}{2}\sum_{ij} \dot{\boldsymbol{\epsilon}}_{ij}^{\,2}\right)^{\left(\frac{1}{2n}-\frac{1}{2}\right)} \qquad (n = 3)$$

- Relevant boundary conditions:
  - ***Stress-free BC:*** $2\mu\dot{\boldsymbol{\epsilon}}_i \cdot \boldsymbol{n} = 0$, on Γ_s
  - ***Floating ice BC:***

    $$2\mu\dot{\boldsymbol{\epsilon}}_i \cdot \boldsymbol{n} = \begin{cases} \rho g z \boldsymbol{n}, & \text{if } z > 0 \\ 0, & \text{if } z \le 0 \end{cases}, \text{ on } \Gamma_l$$

  - ***Basal sliding BC:*** $2\mu\dot{\boldsymbol{\epsilon}}_i \cdot \boldsymbol{n} + \beta ui = 0$, on Γ_β

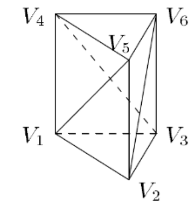


$$\beta = \text{sliding coefficient} \ge 0$$

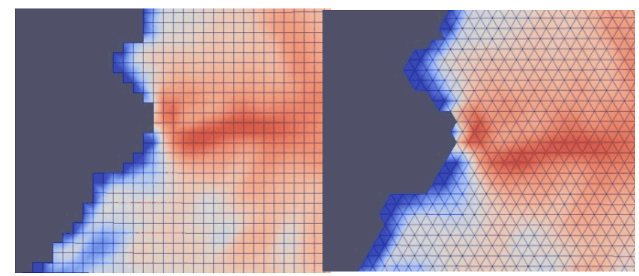
***Assumption:** aspect ratio δ is small and normals to upper/lower surfaces are almost vertical.

Sandia National Laboratories

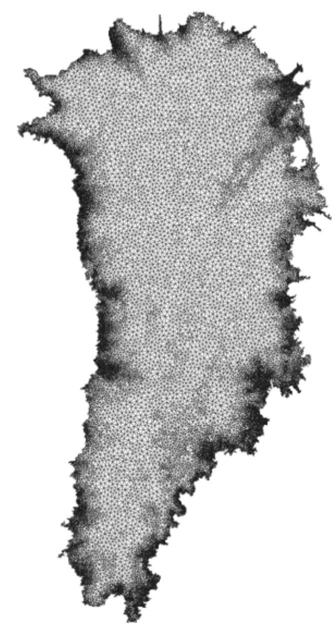
Algorithmic Choices for *Albany/FELIX*: Discretization & Meshes



- **<u>Discretization:</u>** unstructured grid finite element method (FEM)
    - Can handle readily complex geometries.
    - Natural treatment of stress boundary conditions.
    - Enables regional refinement/unstructured meshes.
    - Wealth of software and algorithms.

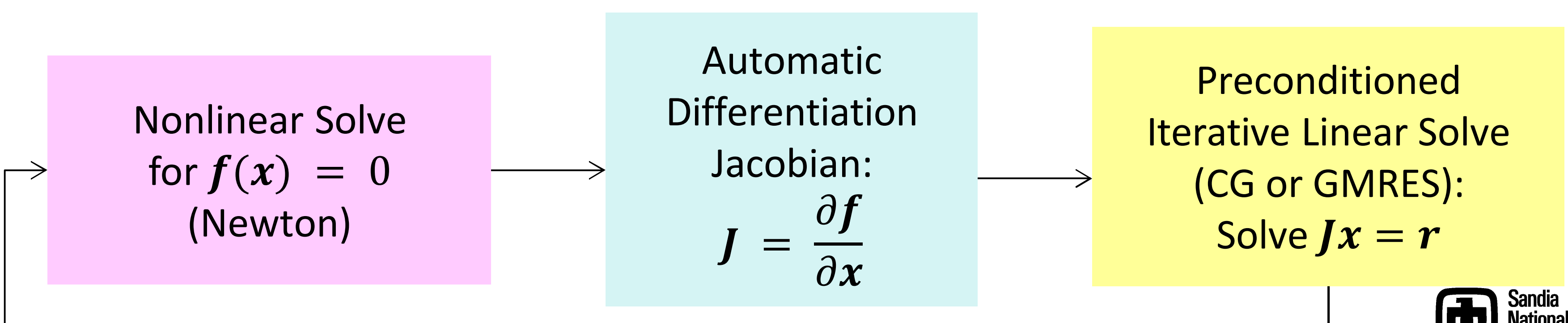


- **<u>Meshes:</u>** can use any mesh but interested specifically in
    - ***Structured hexahedral*** meshes (compatible with *CISM*).
    - ***Tetrahedral*** meshes (compatible with *MPAS LI*)
    - ***Unstructured Delaunay triangle*** meshes with regional refinement based on gradient of surface velocity.
    - All meshes are extruded (structured) in vertical direction as tetrahedra or hexahedra.



Algorithmic Choices for *Albany/FELIX*: Nonlinear & Linear Solver

- **Nonlinear solver:** full Newton with analytic (automatic differentiation) derivatives and homotopy continuation
  - Most robust and efficient for steady-state solves.
  - Jacobian available for preconditioners and matrix-vector products.
  - Analytic sensitivity analysis.
  - Analytic gradients for inversion.
- **Linear solver:** preconditioned iterative method
  - **Solvers:** Conjugate Gradient (CG) or GMRES
  - **Preconditioners:** ILU or algebraic multi-grid (AMG)



The *Albany/FELIX* Solver: Implementation in *Albany* using *Trilinos*

The ***Albany/FELIX*** First Order Stokes solver is implemented in a Sandia (open-source*) parallel C++ finite element code called…

"Agile Components"

- Discretizations/meshes
- Solver libraries
- Preconditioners
- Automatic differentiation
- Many others!

Trilinos

- Configure/build/test/documentation

Started by A. Salinger

Albany

Land Ice Physics Set (***Albany/FELIX code***)

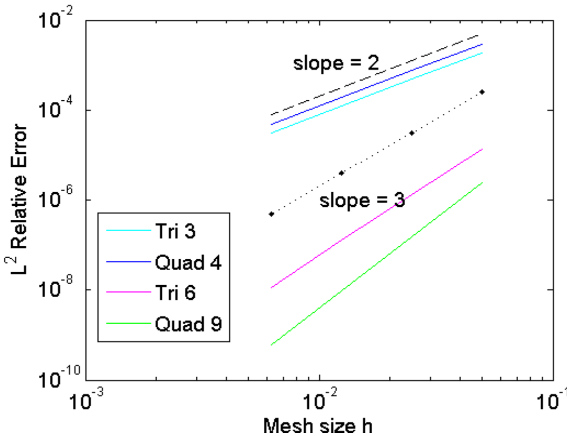
Other Albany Physics Sets

*Available on `github`: https://github.com/gahansen/Albany (Salinger *et al.*, 2015).

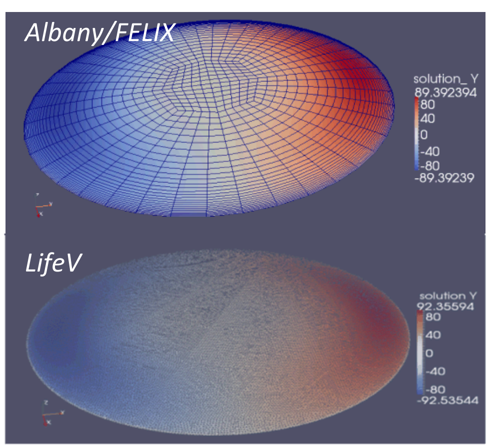
*Use of **Trilinos** components has enabled the **rapid** development of the **Albany/FELIX** First Order Stokes dycore!*

Sandia National Laboratories

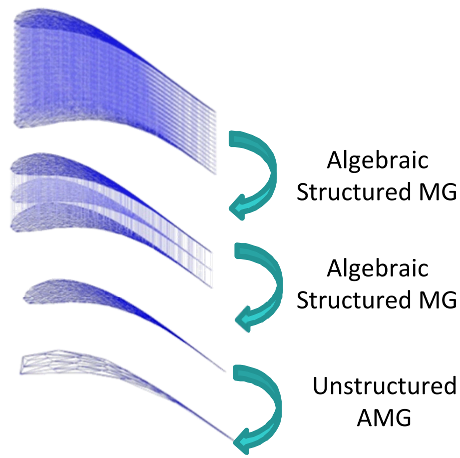
The *Albany/FELIX* Solver is Verified, Scalable, Fast and Robust!

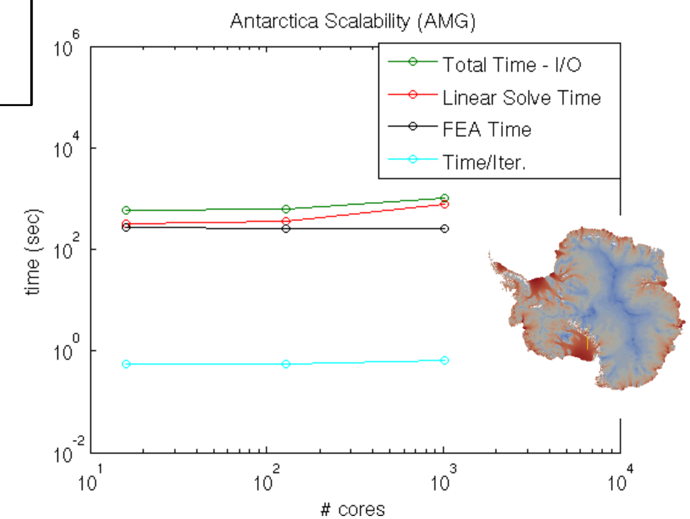


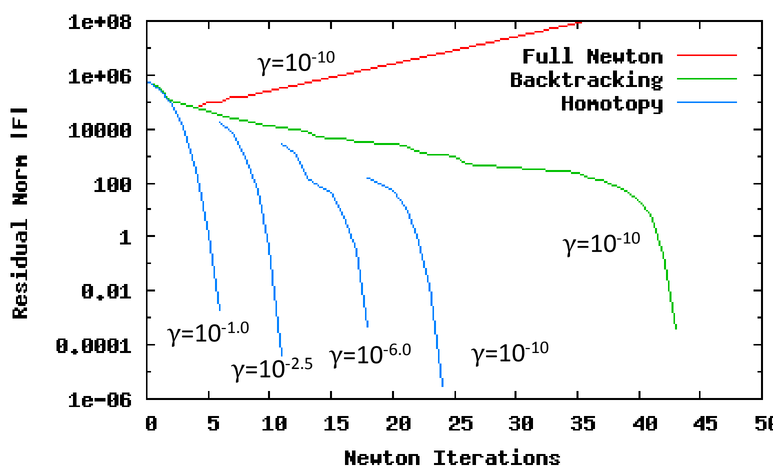
Verified via MMS and code-to-code comparisons.



Scalable via algebraic multi-grid (AMG) preconditioning







$$\mu = \frac{1}{2} A^{-\frac{1}{n}} \left(\frac{1}{2} \sum_{ij} \dot{\epsilon}_{ij}^{\,2} + \gamma \right)^{\left(\frac{1}{2n} - \frac{1}{2}\right)}$$

Robust via homotopy continuation w.r.t. γ.

Sandia National Laboratories

Next Generation Capabilities in *Albany*

- **Uncertainty Quantification**
  - Leverages *DAKOTA* toolkit.
  - Sampling, sensitivity analysis, parameter studies, calibration.
  - Embedded techniques (*Stokhos* + *DAKOTA*).

- **Performance Portability**
  - Leverages C++ *Kokkos* package from *Trilinos.*
  - A *programming model* as much as a software library.
  - Provides automatic access to OpenMP, CUDA, Pthreads, etc.
  - Templated meta-programming: parallel_for, parallel_reduce (templates describe an *execution space*).
  - Memory layout abstraction (“array of structs” vs. “struct of arrays”, locality).

Ice Sheet Evolution Models

- Model for ***evolution of the boundaries*** (thickness evolution equation):

$$\frac{\partial H}{\partial t} = Hflux - \nabla \cdot \int_z \boldsymbol{u}\, dz$$

(conservation of mass).

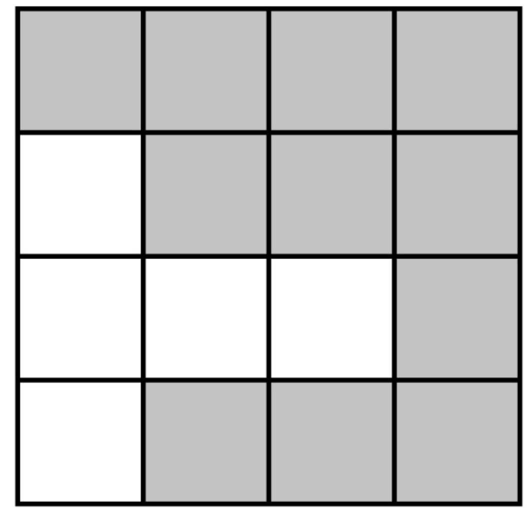
- ***Temperature equation*** (advection-diffusion):

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial z}\left(k \frac{\partial T}{\partial z}\right) - \rho c \boldsymbol{u} \cdot \nabla T + 2\dot{\boldsymbol{\epsilon}}\boldsymbol{\sigma}$$

(energy balance).

- ***Flow factor*** A in Glen's law depends on temperature T: $A = A(T)$.

- Ice sheet ***grows/retreats*** depending on thickness H.



time t_0

Ice-covered ("active") cells shaded in gray ($H > Hmin$)

Ice Sheet Evolution Models

- Model for ***evolution of the boundaries*** (thickness evolution equation):

$$\frac{\partial H}{\partial t} = Hflux - \nabla \cdot \int_z \boldsymbol{u}\, dz$$

(conservation of mass).

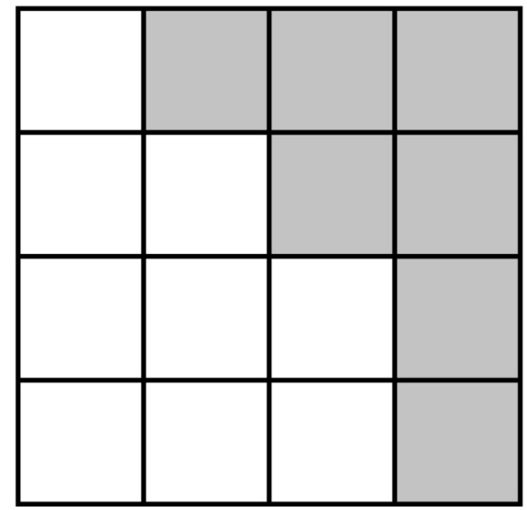
- ***Temperature equation*** (advection-diffusion):

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial z}\left(k \frac{\partial T}{\partial z}\right) - \rho c \boldsymbol{u} \cdot \nabla T + 2\dot{\boldsymbol{\epsilon}}\boldsymbol{\sigma}$$

(energy balance).

- ***Flow factor*** A in Glen's law depends on temperature T: $A = A(T)$.

- Ice sheet ***grows/retreats*** depending on thickness H.



time t1

Ice-covered ("active") cells shaded in gray ($H > Hmin$)

Ice Sheet Evolution Models

- Model for ***evolution of the boundaries*** (thickness evolution equation):

$$\frac{\partial H}{\partial t} = Hflux - \nabla \cdot \int_z \boldsymbol{u}\, dz$$

(conservation of mass).

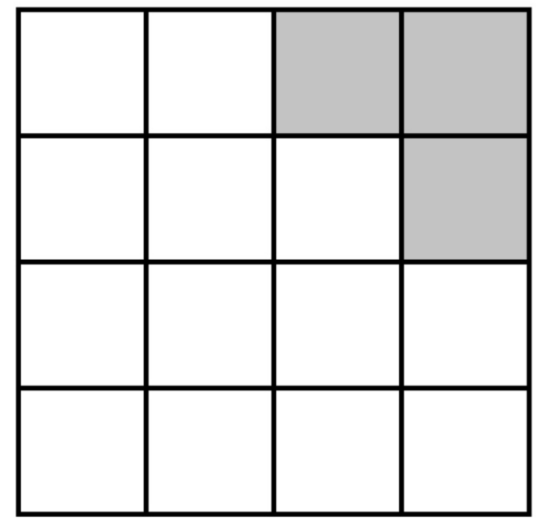
- ***Temperature equation*** (advection-diffusion):

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial z}\left(k \frac{\partial T}{\partial z}\right) - \rho c \boldsymbol{u} \cdot \nabla T + 2\dot{\boldsymbol{\epsilon}}\boldsymbol{\sigma}$$

(energy balance).

- ***Flow factor*** A in Glen's law depends on temperature T: $A = A(T)$.

- Ice sheet ***grows/retreats*** depending on thickness H.



time t_2

Ice-covered ("active") cells shaded in gray ($H > Hmin$)

Sandia National Laboratories

Interfaces to *CISM* and *MPAS LI* for Transient Simulations

CISM-Albany

simple_glide

CISM (Fortran)
Thickness evolution, temperature solve, coupling to CESM

C++/Fortran Interface, Mesh Conversion

output file

Albany/FELIX (C++)
velocity solve

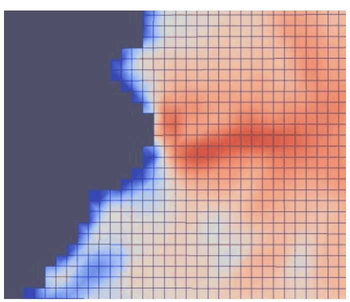
MPAS LI-Albany

LandIce_model

MPAS Land-Ice (Fortran)
Thickness evolution, temperature solve, coupling to DOE-ESM

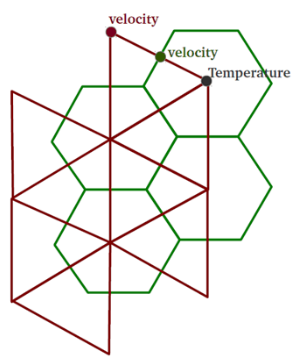
C++/Fortran Interface, Mesh Conversion

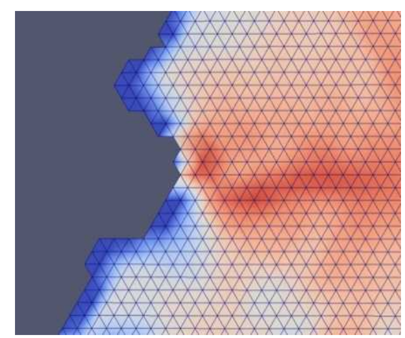
output file



- Structured hexahedral meshes (rectangles extruded to hexes).

- Tetrahedral meshes (dual of hexaganonal mesh, extruded to tets).





Albany/FELIX has been coupled to two land ice dycores: **C**ommunity **I**ce **S**heet **M**odel (***CISM***) and **M**odel for **P**rediction **A**cross **S**cales for **L**and **I**ce (***MPAS LI***)

Sandia National Laboratories

Outline

Albany/FELIX = new land-ice solver with ***next-generation*** capabilities.

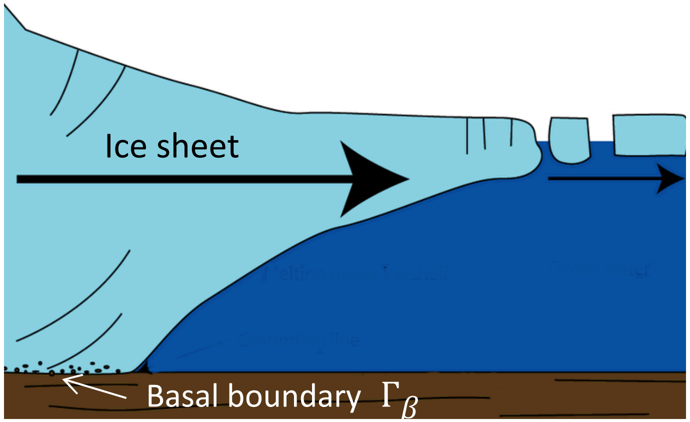
- **Overview:** the PISCEES project and the First Order (FO) Stokes model.
- The ***Albany/FELIX*** First-Order (FO) Stokes diagnostic solver.
- ***Coupling*** of *Albany/FELIX* to the ***CISM*** and ***MPAS*** codes for prognostic simulations of the ice sheet evolution.
- **Uncertainty Quantification (UQ):** Bayesian calibration and forward propagation of uncertainty.
- **Performance portability.**
- **Summary** and ongoing work.
- **Questions?**

Uncertainty Quantification (UQ) Problem Definition

Quantity of Interest (QoI) in Ice Sheet Modeling:
total ice mass loss/gain during 21st century
→ ***sea level rise prediction.***

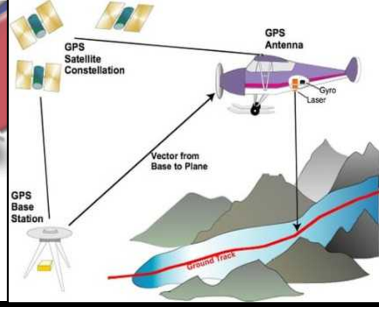
There are several sources of uncertainty, most notably:

- Climate forcings (e.g., surface mass balance).
- **Basal friction (β).**
- Bedrock topography.
- Geothermal heat flux.
- Model parameters (e.g., Glen's flow law exponent).



As a first step, we focus on effect of uncertainty in **basal friction**(β) only.

Basal sliding BC:
$2\mu\dot{\boldsymbol{\epsilon}}_i \cdot \boldsymbol{n} + \beta ui = 0$, on Γ_β



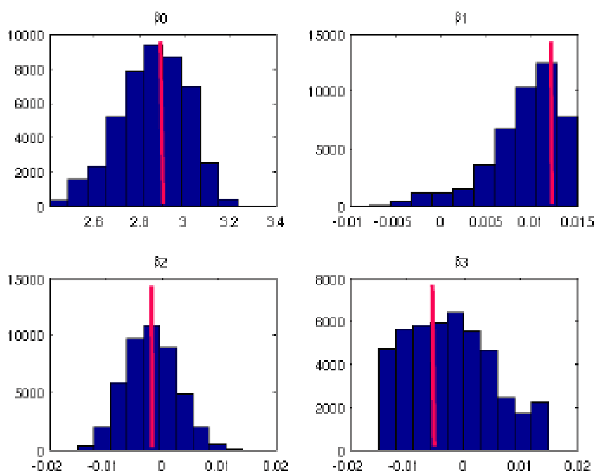
Uncertainty Quantification Workflow

Step 1: Model Initialization through Bayesian Calibration

What are the model parameters that render a given set of observations?

Observations *(known)*: measurements of ice velocity at top surface ($\boldsymbol{u}^{obs}$).

Parameters *(unknown)*: basal sliding field $\beta(x, y)$ at basal surface.



$$\beta(x, y) = \beta_0 + \beta_1 x + \beta_2 y + \beta_3 r$$

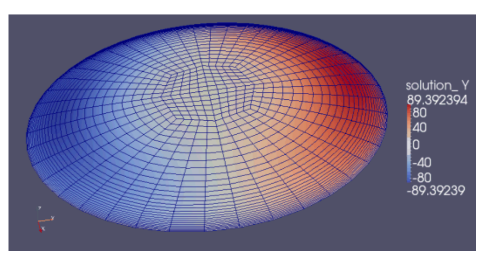
Step 2: Uncertainty Propagation

What is the impact of uncertain parameters in the model on quantities of interest (QOI)?

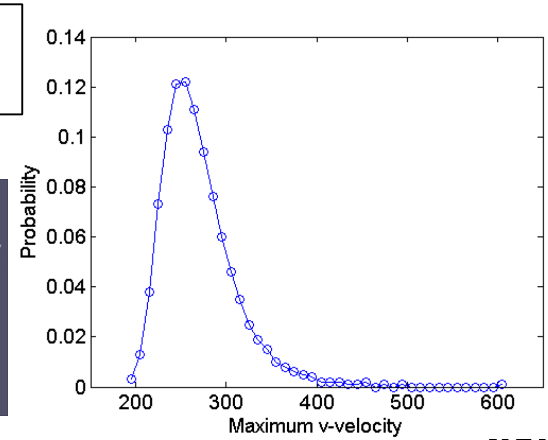
Parameters *(known from Step 1)*: PDF of $\beta(x, y)$.

QOI *(unknown)*: sea-level rise during 21st century.

Basal sliding BC:
$2\mu\dot{\boldsymbol{\epsilon}}_i \cdot \boldsymbol{n} + \beta u i = 0$, on Γ_β



$\beta(x, y) \sim Normal(1,0.2)$



Uncertainty Quantification Workflow (cont'd)

Goal: Uncertainty Quantification in 21st century sea level (QoI)

- **Deterministic inversion:** perform adjoint-based deterministic inversion to estimate initial ice sheet state (i.e., characterize the present state of the ice sheet to be used for performing prediction runs).

- Use deterministic inversion to build a **Gaussian posterior** for the inverse problem (based on recovered fields and Hessian) [Future work].

- **Bayesian calibration:** construct the posterior distribution using Markov Chain Monte Carlo (MCMC) run on an emulator of the forward model.

- **Forward propagation:** sample the obtained distribution and perform ensemble of forward propagation runs to compute the uncertainty in the QoI.

Deterministic Inversion: Estimation of Ice Sheet Initial State

Objective: find ice sheet initial state that

- Matches observations (e.g., surface velocity, temperature, etc.)
- Matches present-day geometry (elevation, thickness).
- Is in "equilibrium" with climate forcings (SMB).

Approach: invert for unknown/uncertain ice sheet model parameters.

- Significantly reduces non-physical transients without model spin-up.

Available data/measurements:

- (Ice extent and surface topography.)
- Surface velocity.
- Surface mass balance (SMB).
- Ice thickness H (sparse measurements).

Field to be estimated:

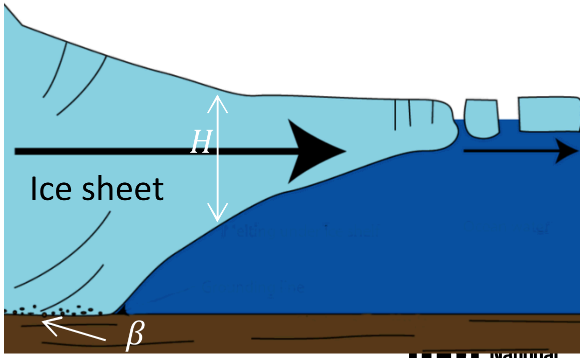
- Ice thickness H (allowed to be weighted by observational uncertainties).
- Basal friction β (spatially variable proxy for all basal processes)

Assumptions:

- Ice flow described by FO Stokes equations.
- Ice is close to mechanical equilibrium.
- Temperature field is given.

Basal sliding BC:

$2\mu\dot{\boldsymbol{\epsilon}}_i \cdot \boldsymbol{n} + \beta u i = 0$, on Γ_β



Deterministic Inversion: Estimation of Ice Sheet Initial State (cont'd)

First Order Stokes PDE Constrained Optimization Problem:

$$J(\beta, H) = \frac{1}{2}\alpha_v \int_{\Gamma_{top}} |\boldsymbol{u} - \boldsymbol{u}^{obs}|^2 ds + \frac{1}{2}\alpha \int_{\Gamma} |div(\boldsymbol{U}H) - SMB|^2 ds + \frac{1}{2}\alpha_H \int_{\Gamma_{top}} |H - H^{obs}|^2 ds + \mathcal{R}(\beta) + \mathcal{R}(H)$$

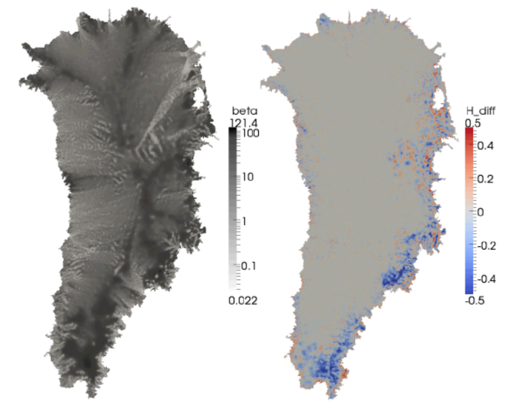
- Minimize difference between:
  - Computed and measured ***surface velocity*** **($\boldsymbol{u}^{obs}$)** → *common*
  - Computed divergence flux and measured ***surface mass balance (SMB)*** → *novel*
  - Computed and ***reference thickness*** **(*H^{obs}*)** → *novel*

- Control variables:
  - ***Basal friction*** **($\boldsymbol{\beta}$)**.
  - ***Thickness (H)***.

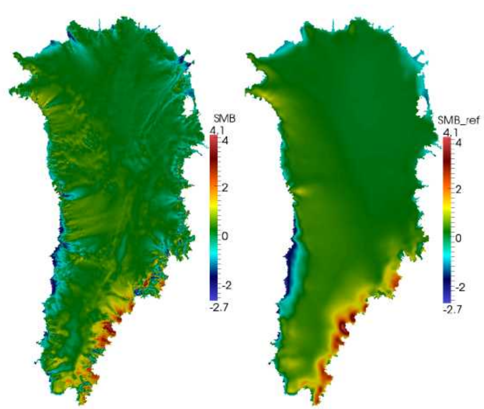
<u>*Software tools for adjoint-based inversion:*</u>

- ***Albany*** (assembly)
- ***Trilinos*** (linear/nonlinear solvers)
- ***ROL*** (gradient-based optimization).

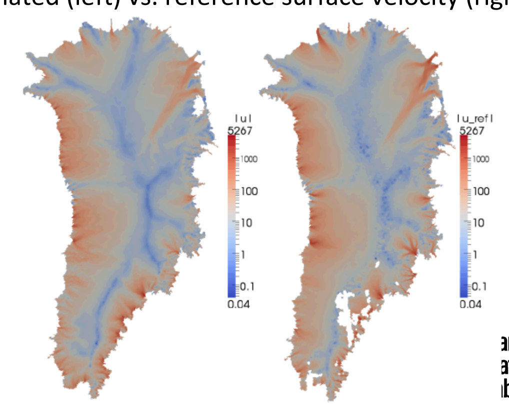
Estimated β $\quad$ $H - Hobs$



Estimated divergence (left) vs. reference SMB (right)



Estimated (left) vs. reference surface velocity (right)



Bayesian Calibration: Proof-of-Concept using KLE

Albany/FELIX has been hooked up to ***DAKOTA/QUESO*** (in "black-box" mode) for ***UQ/Bayesian calibration.***

Difficulty in UQ: "Curse of Dimensionality"
The β-field inversion problem has $O(100K)$ dimensions!

Approach: Reduce $O(100K)$ dimensional problem to $O(10)$ dimensional problem.

- For initial proof-of-concept, we use the ***Karhunen-Loeve Expansion (KLE)***:

Offline

1. Assume analytic covariance kernel $C(r_1, r_2) = exp\left(-\frac{(r_1-r_2)^2}{L^2}\right)$.
2. Perform eigenvalue decomposition of C.

Online

3. Expand* $\beta - \bar{\beta}$ in basis of eigenvectors $\{\boldsymbol{\phi}_k\}$ of C, with random variables $\{\xi_k\}$:

$$\beta(\omega) = \bar{\beta} + \sum_{k=1}^{K} \sqrt{\lambda_k} \boldsymbol{\phi}_k \xi_k(\omega)$$

$\bar{\beta}$ = solution to deterministic inversion problem (previous slide)

Inference/calibration is for coefficients of KLE
$\Rightarrow$ ***significant dimension reduction***.

*In practice, expansion is done on $\log(\beta)$ to avoid negative values of β.

Bayesian Calibration: Proof-of-Concept Using KLE (cont'd)

- **<u>Step 1 (*Trilinos*):</u>** Reduce $O(100K)$ dimensional problem to $O(10)$ dimensional problem using ***Karhunen-Loeve Expansion (KLE)***:

Offline

1. Assume analytic covariance kernel $C(r_1, r_2) = exp\left(-\frac{(r_1 - r_2)^2}{L^2}\right)$.
2. Perform eigenvalue decomposition of C.

Online

3. Expand* $\beta - \bar{\beta}$ in basis of eigenvectors $\{\boldsymbol{\phi}_k\}$ of C, with random variables $\{\xi_k\}$:

$$\beta(\omega) = \bar{\beta} + \sum_{k=1}^{K} \sqrt{\lambda_k}\boldsymbol{\phi}_k \xi_k(\omega)$$

$\bar{\beta}$ = solution to deterministic inversion problem

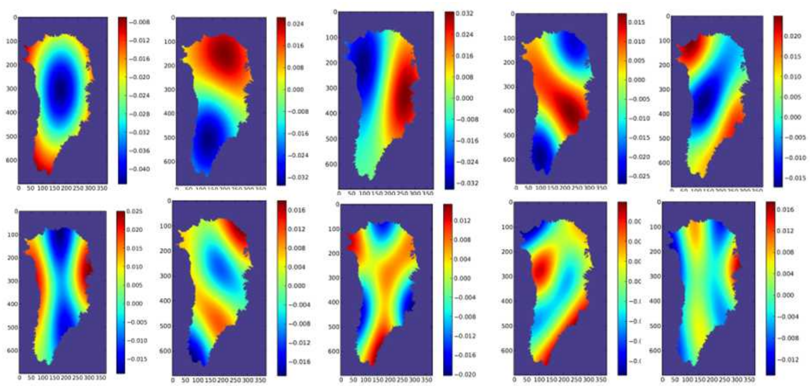
- **<u>Step 2 *(DAKOTA)*:</u>** ***Polynomial Chaos Expansion (PCE)*** emulator for mismatch over surface velocity discrepancy.

- **<u>Step 3 *(QUESO)*:</u>** ***Markov Chain Monte Carlo (MCMC)*** calibration using PCE emulator.
  →can obtain posterior distributions on KLE coefficients.

*In practice, expansion is done on $\log(\beta)$ to avoid negative values of β.

Bayesian Calibration: Illustration on 4km GIS Problem

- Mean $\bar{\beta}$ field obtained XXXX *(right).*
- 10 KLE modes capture ??% of covariance energy (due to correlation length; only spatial correlation has been considered).



- Mismatch function (calculated in *Albany/FELIX*):

$$J(\beta, H) = \int_{\Gamma_{top}} \frac{1}{{\sigma_u}^2} |\boldsymbol{u} - \boldsymbol{u}^{obs}|^2 ds$$

- PCE emulator was formed for the mismatch $J(\beta, H)$ using uniform prior distributions.
- For calibration, MCMC was performed on the PCE with ??K samples.

Bayesian Calibration: Illustration on 4km GIS Problem (cont'd)

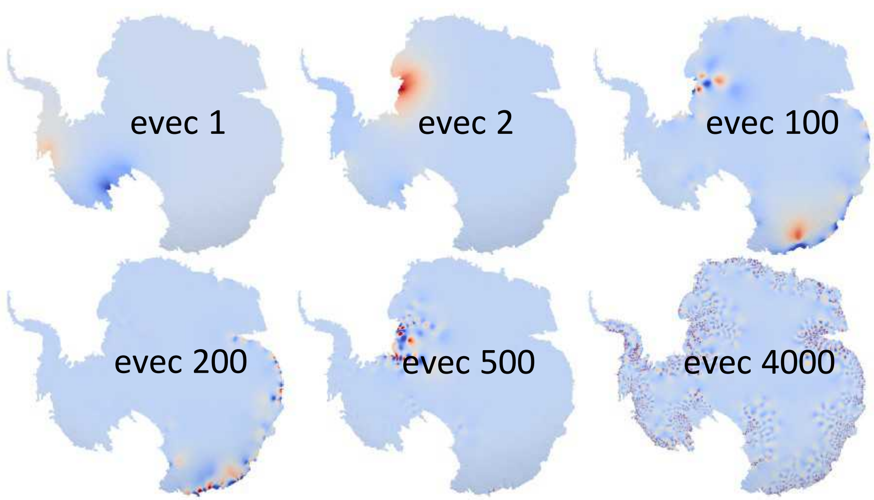
- Insert pictures of distributions; discussion.

Bayesian Calibration: Building Gaussian Distribution using Hessian

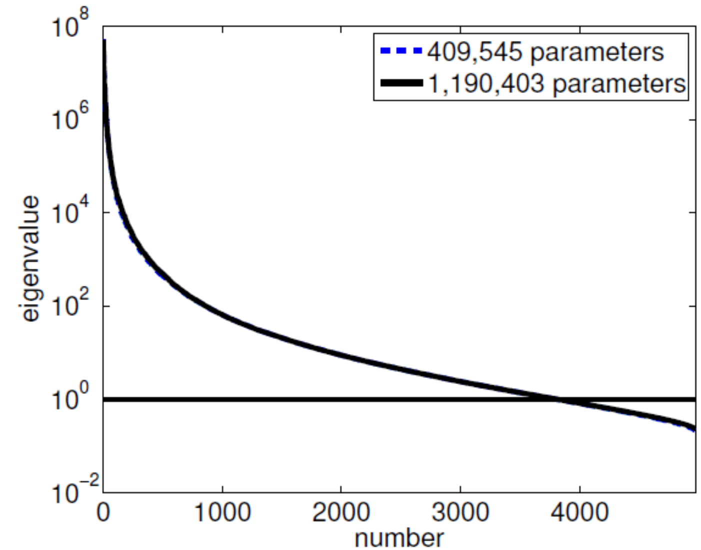
- Hessian of the merit functional (velocity mismatch) can provide a way to compute the covariance of a Gaussian posterior:

$$\boldsymbol{C}_{post} = (\boldsymbol{C}_{prior}\boldsymbol{H}_{misfit} + \boldsymbol{I})^{-1}\boldsymbol{C}_{prior}$$

- We want to limit only the most important directions (eigenvectors) of $\boldsymbol{C}_{post}$.



Figures courtesy of O. Ghattas' group (Isaac et al, 2004)



- **Issue:** there are still too many (~1000) significant eigenvalues (*right*: log-linear plot of spectra for 2 sample Hessians).

Forward Propagation

PCE Emulator

$$\beta(\omega) = \bar{\beta} + \sum_{k=1}^{K} \sqrt{\lambda_k}\boldsymbol{\phi}_k \xi_k(\omega)$$

→ Model realizations
Forward propagation
(e.g., 2000-2050)

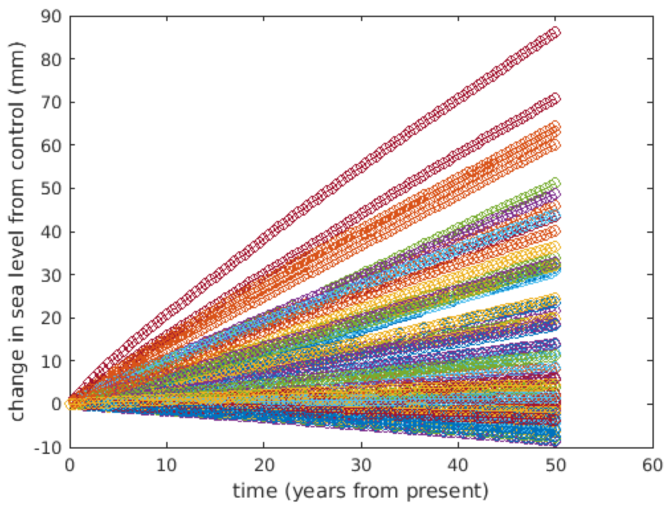
DAKOTA, Albany/FELIX
QoI(β)
(total ice mass loss)

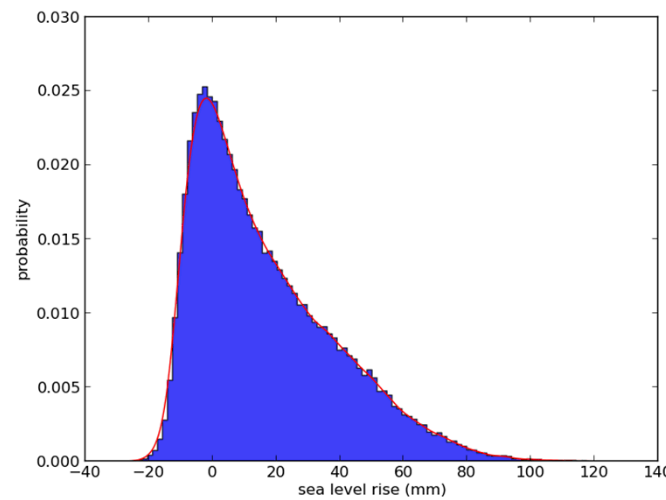
- Parameter (β) distribution can either be assumed to be Gaussian (based on Hessian information) or can be the result of Bayesian calibration.
- Emulator is built using *DAKOTA* coupled with *Albany/FELIX* for forward runs.
  - Use **compressed sensing** to adaptively select significant modes and basis from parameter space. The hope is that only a few modes affect the QoI.
  - Could use cheaper physical models to reduce computational time of forward model.
- MCMC (*QUESO)* used to perform uncertainty propagation.
- More details: priors, etc?

Sandia National Laboratories

Forward Propagation: Illustration on 4km GIS Problem

- Explain procedure!





- *Left:* SLR distribution from an ensemble of 66 high-fidelity simulations (ensemble members were differenced against a control run using the $\bar{\beta}$ distribution. ***All 66 runs ran to completion out-of-the-box on Hopper!***

- *Right:* PDF of SLR from the PCE build using compressed sensing with the 66 high-fidelity simulations. The PDF was generated by sampling the 10D KLE modes.

Sandia National Laboratories

Forward Propagation: Illustration on 4km GIS Problem (cont'd)

Explanations for long tail in sea level PDF:

- Rapid sliding is confined to an area that is a small fraction of the overall ice sheet.
- A perturbation to the initial beta field that further increases β in areas where there is already very little sliding won't affect the output much.
- Decreasing β in areas where there is currently little sliding has a very large effect, since velocity in those regions will change significantly from the initial condition.

Forward Propagation: Illustration on 4km GIS Problem (cont'd)

Outline

Albany/FELIX = new land-ice solver with ***next-generation*** capabilities.

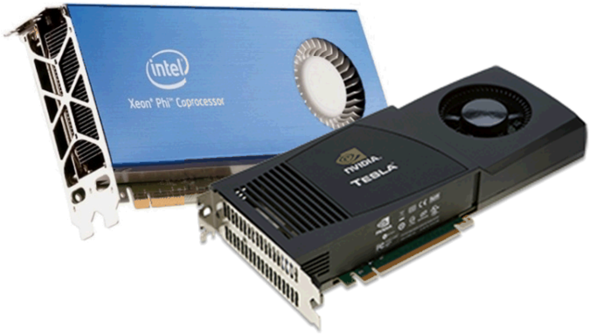
- **Overview:** the PISCEES project and the First Order (FO) Stokes model.
- The ***Albany/FELIX*** First-Order (FO) Stokes diagnostic solver.
- ***Coupling*** of *Albany/FELIX* to the ***CISM*** and ***MPAS*** codes for prognostic simulations of the ice sheet evolution.
- **Uncertainty Quantification (UQ):** Bayesian calibration and forward propagation of uncertainty.
- **Performance portability.**
- **Summary** and ongoing work.
- **Questions?**

Performance-Portability via *Kokkos*

With I. Demeshko (SNL)

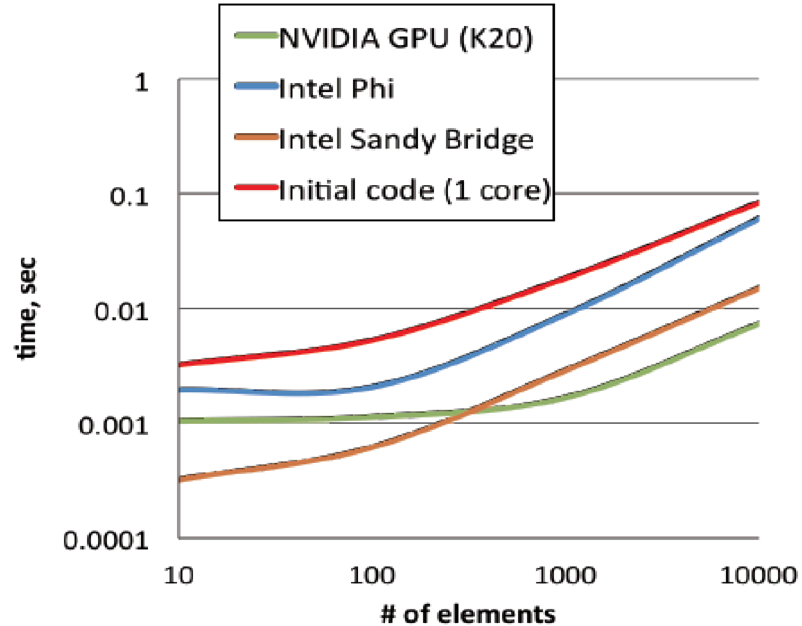
We need to be able to run *Albany/FELIX* on ***new architecture machines*** (hybrid systems) and ***manycore devices*** (multi-core CPU, NVIDIA GPU, Intel Xeon Phi, etc.) .

- ***Kokkos***: *Trilinos* library and programming model that provides performance portability across diverse devises with different memory models.
- With *Kokkos*, you write an algorithm once, and just change a template parameter to get the optimal data layout for your hardware.



Performance-Portability via *Kokkos* (continued)

- ***Right:*** results for a **mini-app** that uses finite element kernels from *Albany/FELIX* but none of the surrounding infrastructure.
  - "# of elements" = threading index (allows for on-node parallelism).
  - # of threads required before the Phi and GPU accelerators start to get enough work to warrant overhead: ~100 for the Phi and ~1000 for the GPU.



- ***Below:*** preliminary results for 3 of the finite element assembly kernels, as part of **full *Albany/FELIX*** code run.

Kernel	Serial	16 OpenMP Threads	GPU
Viscosity Jacobian	20.39 s	2.06 s	0.54 s
Basis Functions w/ FE Transforms	8.75 s	0.94 s	1.23 s
Gather Coordinates	0.097 s	0.107 s	5.77 s

Note: *Gather Coordinates routine requires copying data from host to GPU.*

Sandia National Laboratories

Summary and Ongoing Work

Summary: this talk described…

- The development of a finite element land ice solver known as *Albany/FELIX* written using the libraries of the *Trilinos* libraries.
- Coupling of *Albany/FELIX* to the *CISM* and *MPAS* codes for transient simulations of ice sheet evolution.
- Advanced, next generation capabilities (UQ, performance portability) were highlighted.

Verification, science simulations, scalability, robustness, UQ, advanced analysis: all attained in **~1.5 FTE of effort**!

Ongoing/future work:

- Science runs using *CISM-Albany* and *MPAS-Albany*.
- Bayesian calibration using better bases than KLE (e.g., Hessian eigenvectors).
- Continued porting of code to new architecture supercomputers (*Titan*, *Cori Phase I*).
- Delivering code to climate community and coupling to earth system models.

Funding/Acknowledgements

Support for this work was provided through Scientific Discovery through Advanced Computing (**SciDAC**) projects funded by the U.S. Department of Energy, Office of Science (**OSCR**), Advanced Scientific Computing Research and Biological and Environmental Research (**BER**) → **PISCEES SciDAC Application Partnership.**

PISCEES team members: *W. Lipscomb, S. Price, M. Hoffman, A. Salinger, M. Perego, I. Tezaur, R. Tuminaro, P. Jones, K. Evans, P. Worley, M. Gunzburger, C. Jackson;*

Trilinos/DAKOTA collaborators*: E. Phipps, M. Eldred, J. Jakeman, L. Swiler.*

Thank you! Questions?

References

[1] M.A. Heroux *et al.* “An overview of the Trilinos project.” *ACM Trans. Math. Softw.* **31**(3) (2005).

[2] A.G. Salinger *et al*. "Albany: Using Agile Components to Develop a Flexible, Generic Multiphysics Analysis Code", *Comput. Sci. Disc.* (submitted, 2015).

[3] **I. Tezaur**, M. Perego, A. Salinger, R. Tuminaro, S. Price. "*Albany/FELIX*: A Parallel, Scalable and Robust Finite Element Higher-Order Stokes Ice Sheet Solver Built for Advanced Analysis", *Geosci. Model Develop.* 8 (2015) 1-24.

[4] **I. Tezaur**, R. Tuminaro, M. Perego, A. Salinger, S. Price. "On the scalability of the *Albany/FELIX* first-order Stokes approximation ice sheet solver for large-scale simulations of the Greenland and Antarctic ice sheets", *MSESM/ICCS15*, Reykjavik, Iceland (June 2014).

[5] R.S. Tuminaro, **I. Tezaur**, M. Perego, A.G. Salinger. "A Hybrid Operator Dependent Multi-Grid/Algebraic Multi-Grid Approach: Application to Ice Sheet Modeling", *SIAM J. Sci. Comput.* (in prep).

[6] R. Tuminaro. “ML’s SemiCoarsening Feature, Addition to ML 5.0 Smoothed Aggregation User’s Guide” , Sandia National Laboratories Report, SAND2006-2649, Sandia National Laboratories, Albuquerque, NM, 2014.

References (cont'd)

[7] S. Shannon, *et al.* "Enhanced basal lubrication and the contribution of the Greenland ice sheet to future sea-level rise", *P. Natl. Acad. Sci.,* 110 (2013) 14156-14161.

[8] P. Fretwell, *et al.* "BEDMAP2: Improved ice bed, surface, and thickness datasets for Antarctica", *The Cryosphere* 7(1) (2013) 375-393.

[9] F. Pattyn. "Antarctic subglacial conditions inferred from a hybrid ice sheet/ice stream model", *Earth and Planetary Science Letters 295 (2010).*

[10] M. Perego, S. Price, G. Stadler. "Optimal Initial Conditions for Coupling Ice Sheet Models to Earth System Models", *J. Geophys. Res.* 119 (2014) 1894-1917.

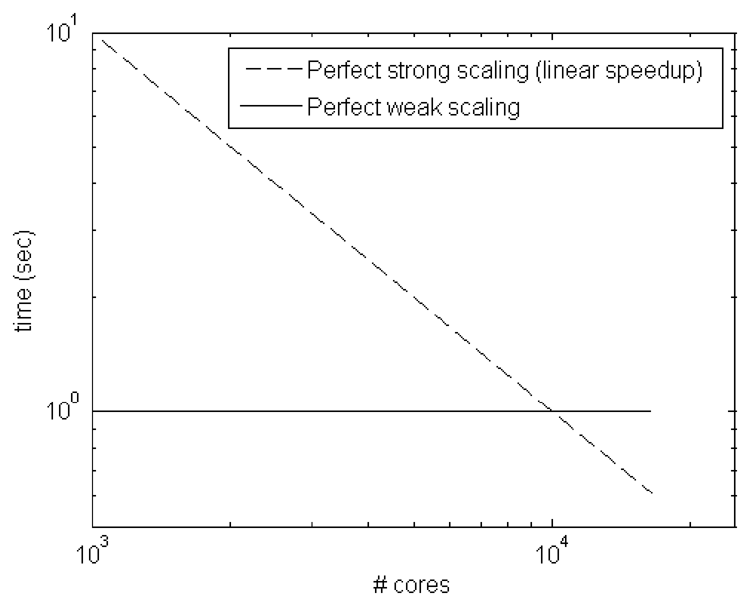
Sandia National Laboratories

Definitions: Strong vs. Weak Scaling

Scalability (a.k.a. **Scaling Efficiency**) = measure of the efficiency of a code when increasing numbers of parallel processing elements (CPUs, cores, processes, threads, etc.).

- ***Strong scaling:*** how the solution time varies with the number of cores for a fixed total problem size.
  - ⇒ Fix problem size, increase # cores.
  - **<u>Ideal:</u>** linear speed-up with increase in # cores.

- ***Weak scaling:*** how the solution time varies with the number of cores for a fixed problem size per core.
  - ⇒ Increase problem size and # cores s.t. # dofs/core is approximately constant.
  - **<u>Ideal:</u>** solution time remains constant as problem size and # cores increases.



Sandia National Laboratories

Scalability via Algebraic Multi-Grid Preconditioning with Semi-Coarsening

Bad aspect ratios ruin classical AMG convergence rates!

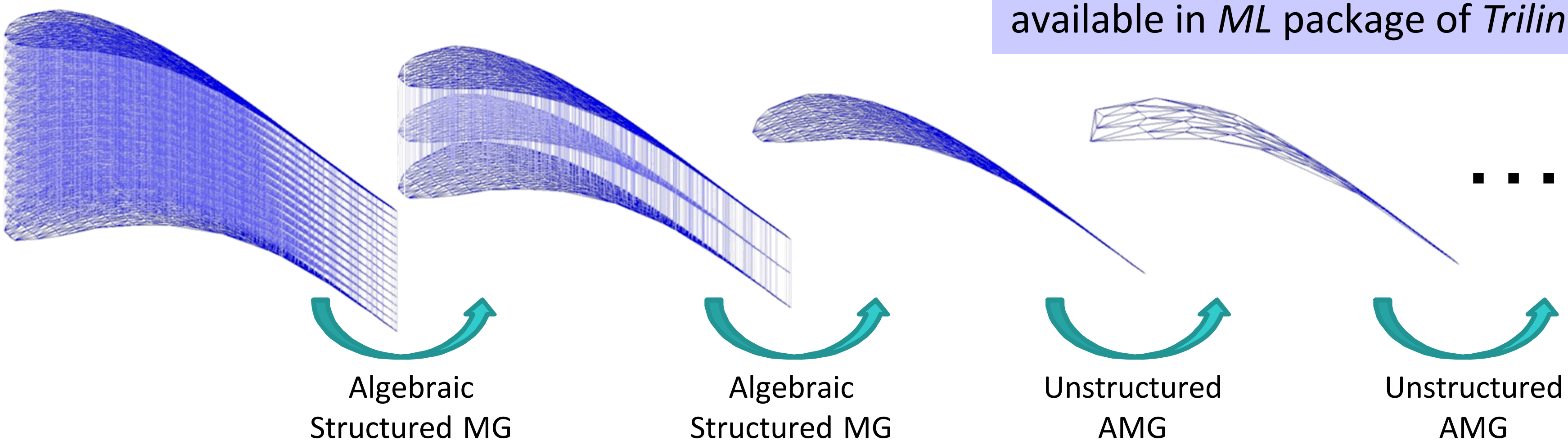
- relatively small horizontal coupling terms, hard to smooth horizontal errors

⇒ Solvers (AMG and ILU) must take aspect ratios into account

We developed a **new AMG solver** based on aggressive **semi-coarsening** *(figure below)*

- Algebraic Structured MG (≡ matrix depend. MG) used with vertical line relaxation on finest levels + traditional AMG on 1 layer problem

New AMG preconditioner is available in *ML* package of *Trilinos*!

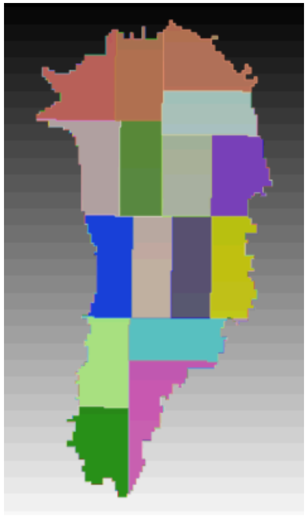


Scaling studies (next slides):
New AMG preconditioner vs. ILU

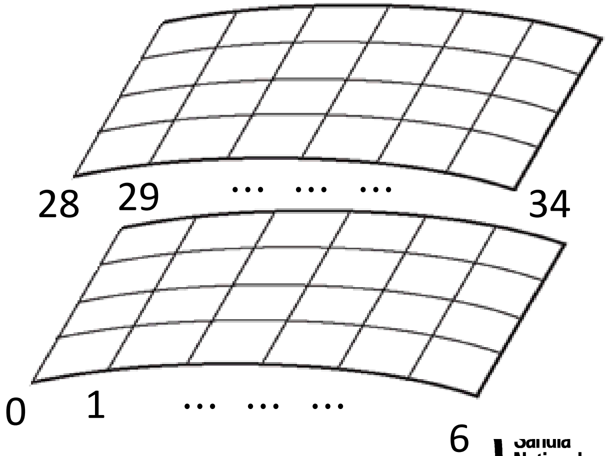
See (Tuminaro, 2014), (Tezaur *et al.*, 2015), (Tuminaro *et al.*, 2015).

Importance of Node Ordering & Mesh Partitioning

Our studies revealed that **node ordering** and **mesh partitioning** matters for linear solver performance, especially for the ILU preconditioner!



- It is essential that incomplete factorization accurately captures vertical coupling, which is dominant due to anisotropic mesh.
- This is accomplished by:
  - Ensuring all points along a vertically extruded grid line reside within a single processor ("**2D mesh partitioning**"; top right).
  - Ordering the equations such that grid layer k's nodes are ordered before all dofs associated with grid layer $k+1$ ("**row-wise ordering**"; bottom right).



Strong Scaling Study for a Fine-Resolution GIS Problem

- Uniform quadrilateral mesh with 1 km horizontal resolution, extruded vertically using 40 layers (69.8M hex elements, 143M dofs).

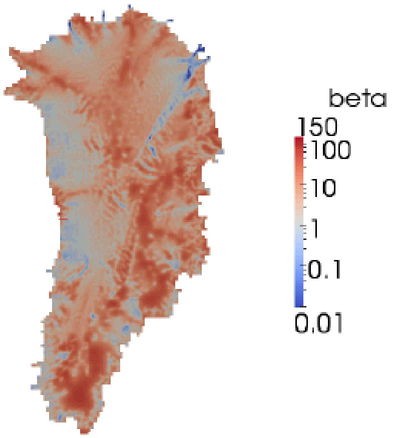
- Run on 1024→16,384 cores of *Hopper* (16-fold increase).

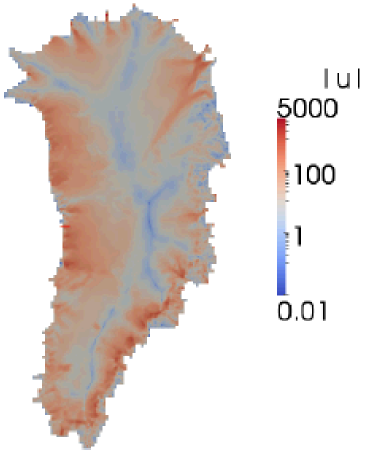
- Realistic basal friction coefficient and bed topographies calculated by solving a deterministic inversion problem that minimized modeled and observed surface velocity mismatch (Perego *et al.*, 2014; top right).

- Realistic 3D temperature field calculated in *CISM* (Shannon *et al.*)

- **Preconditioner:** ILU vs. new AMG (with aggressive semi-coarsening).

- **Iterative linear solver:** Conjugate Gradient (CG).

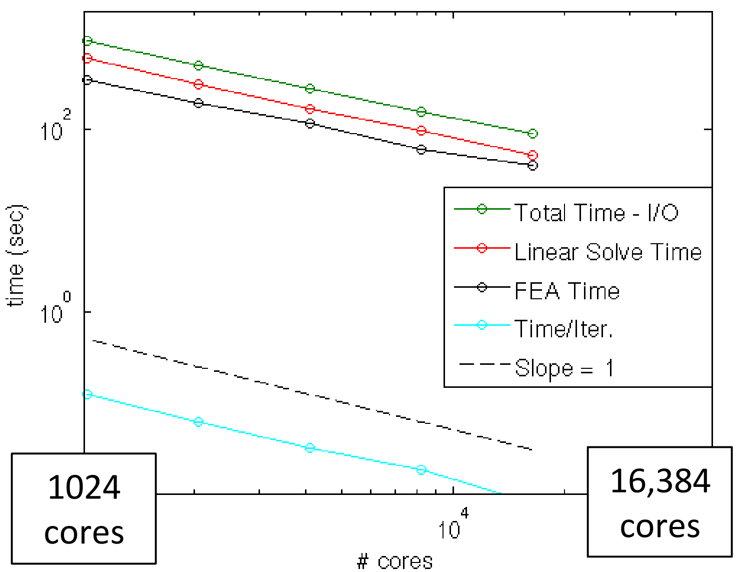


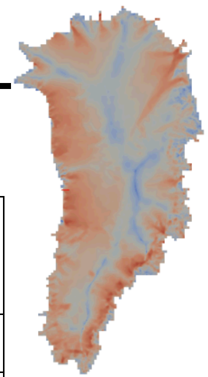


Sandia National Laboratories

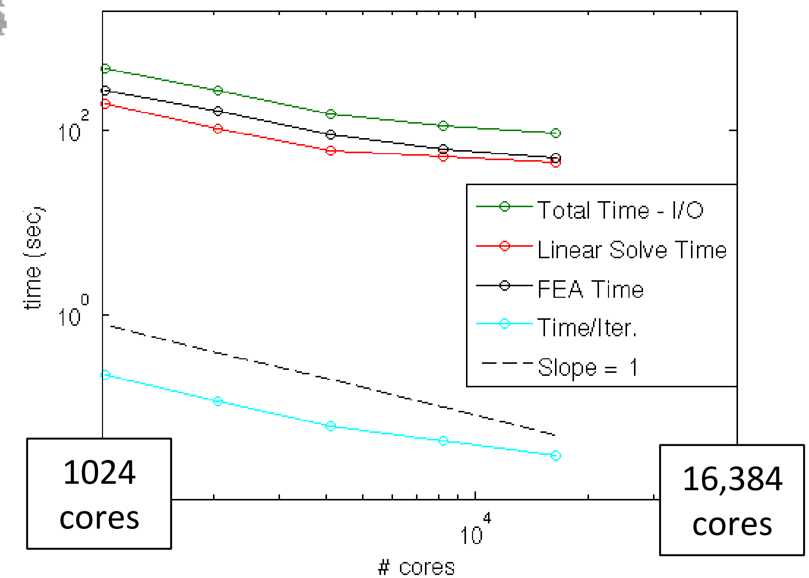
Strong Scaling Study for a Fine-Resolution GIS Problem (cont'd)

ILU





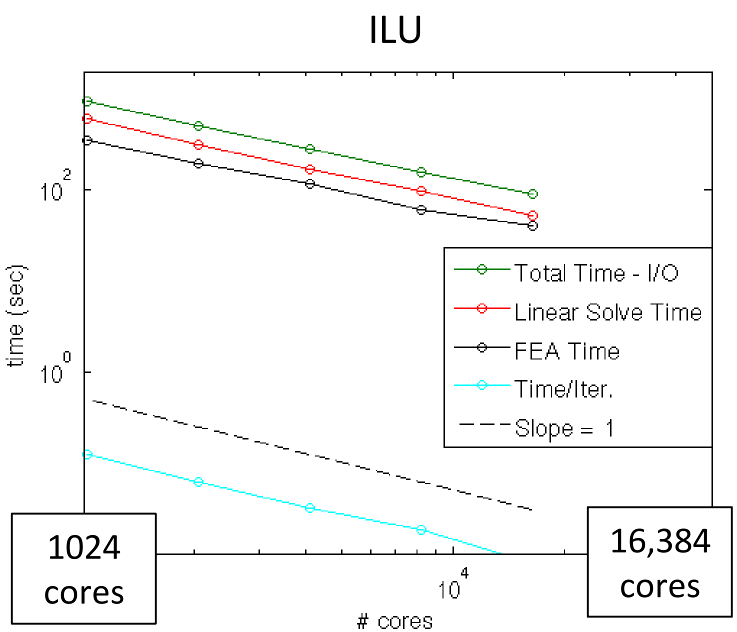
AMG

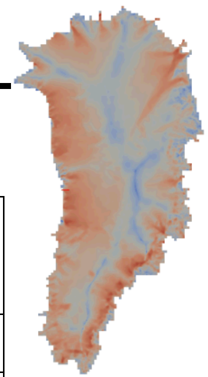


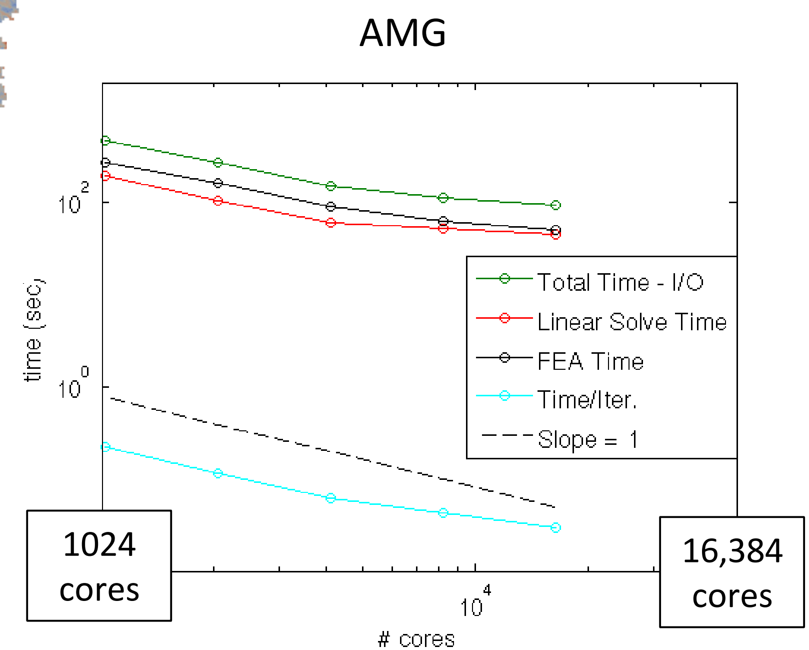
1024 core run:

- AMG preconditioner solves are much faster than ILU (e.g., 194.3 sec for AMG vs. 607.9 sec for ILU).
  - Primarily due to better convergence rate obtained with AMG vs. ILU.

Strong Scaling Study for a Fine-Resolution GIS Problem (cont'd)



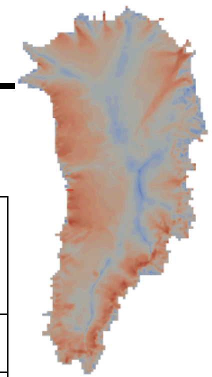




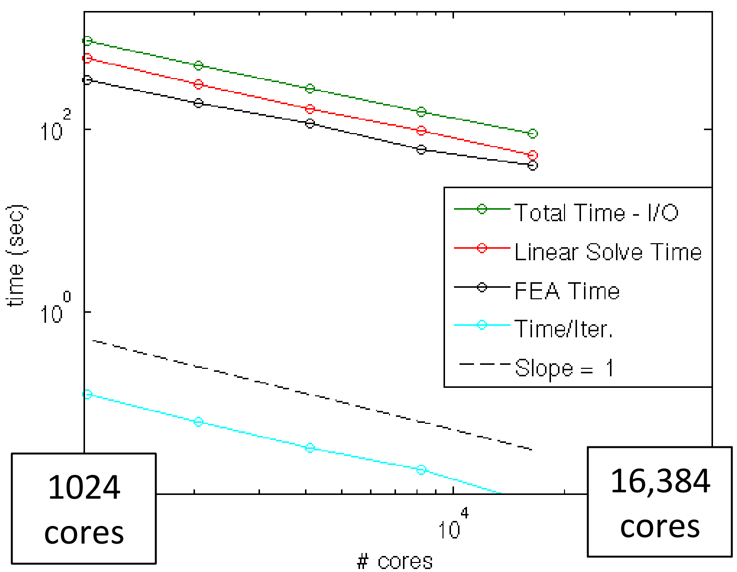
<u>16,384 core run:</u>

- ILU preconditioner fairly effective relative to AMG when # dofs/core is modest (e.g., 10K dofs/core).
  - ILU requires slightly more iterations/linear solve but cost/iteration is higher for AMG.
  - AMG solver is very inefficient when # dofs/core is small; communication costs in coarse level processing dominate.

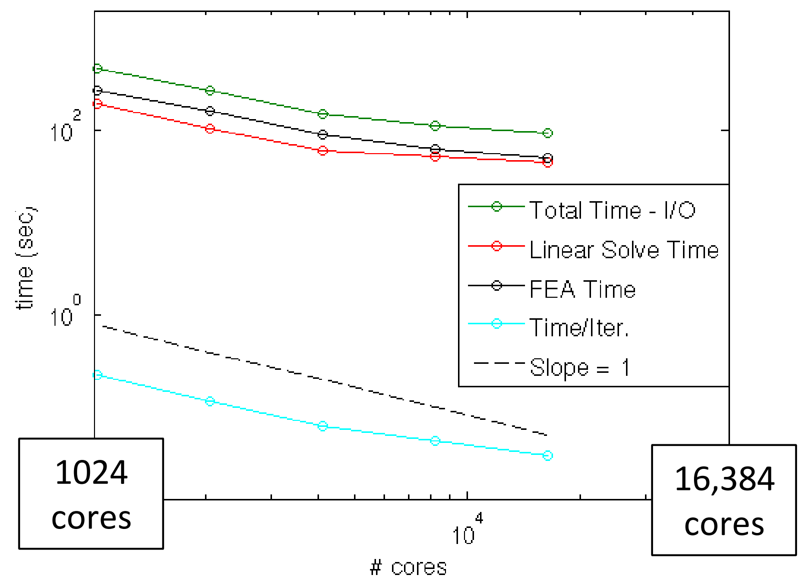
Strong Scaling Study for a Fine-Resolution GIS Problem (cont'd)



ILU



AMG

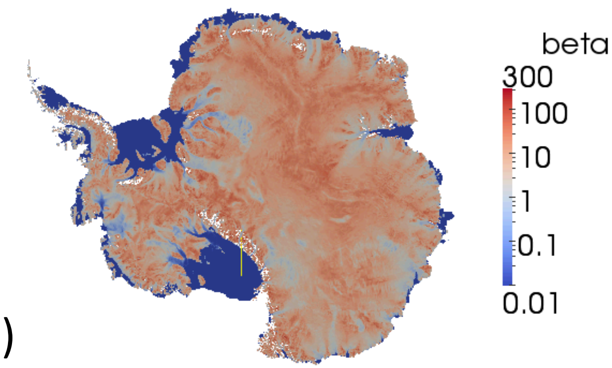


Summary:

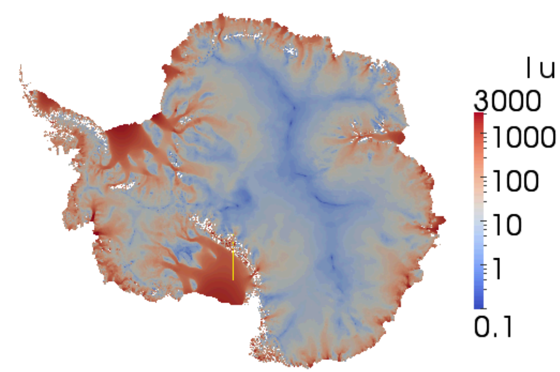
- ILU preconditioner scales better in the strong sense than AMG.
- However, ILU-preconditioned solve is slower for lower #s of cores (more dofs/core).

Weak Scaling Study for a Moderate-Resolution AIS Problem

- 3 hexahedral meshes considered:
  - 8 km horizontal resolution + 5 vertical layers (2.52M dofs) → 16 cores of *Hopper*.
  - 4 km horizontal resolution + 10 vertical layers (18.5M dofs) → 128 cores of *Hopper*.
  - 2 km horizontal resolution + 20 vertical layers (141.5M dofs) → 1024 cores of *Hopper*.



- Ice sheet geometry based on BEDMAP2 (Fretwell *et al*., 2013) and 3D temperature field from (Pattyn, 2010)
- Realistic regularized* basal friction coefficient and bed topographies calculated by solving a deterministic inversion problem that minimizes modeled and observed surface velocity mismatch on finest (2km) resolution geometry (Perego *et al.*, 2014; top right).

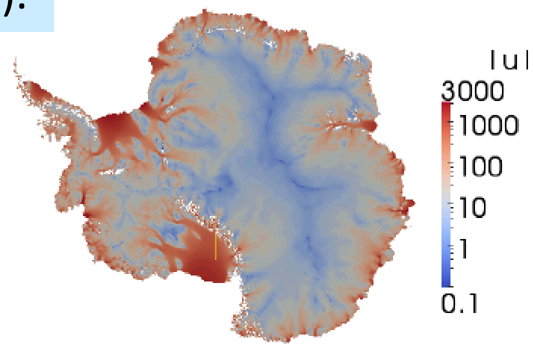


- **Preconditioner:** ILU vs. new AMG (with aggressive semi-coarsening).
- **Iterative linear solver:** GMRES.

*Setting $\beta = \delta > 0$, with $\delta \ll 1$ under ice shelves.

Weak Scaling Study for a Moderate-Resolution AIS Problem (cont'd)

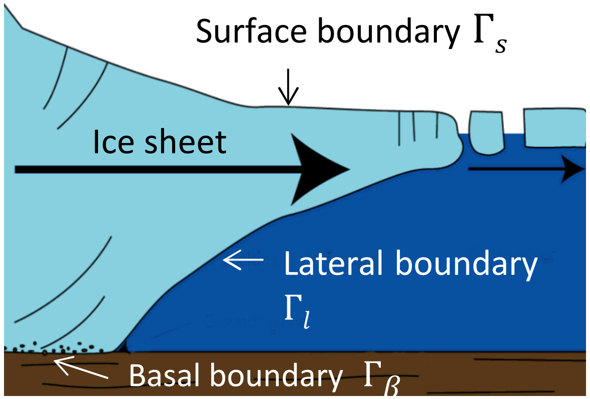
Antarctica is fundamentally different than Greenland:
AIS contains large ice shelves (floating extensions of land ice).



- ***Along ice shelf front:*** open-ocean BC (Neumann).
- ***Along ice shelf base:*** zero traction BC (Neumann).

⇒ For vertical grid lines that lie within ice shelves, top and bottom BCs resemble Neumann BCs so sub-matrix associated with one of these lines is almost* singular.

(vertical > horizontal coupling)
+
Neumann BCs
=
nearly singular submatrix associated with vertical lines

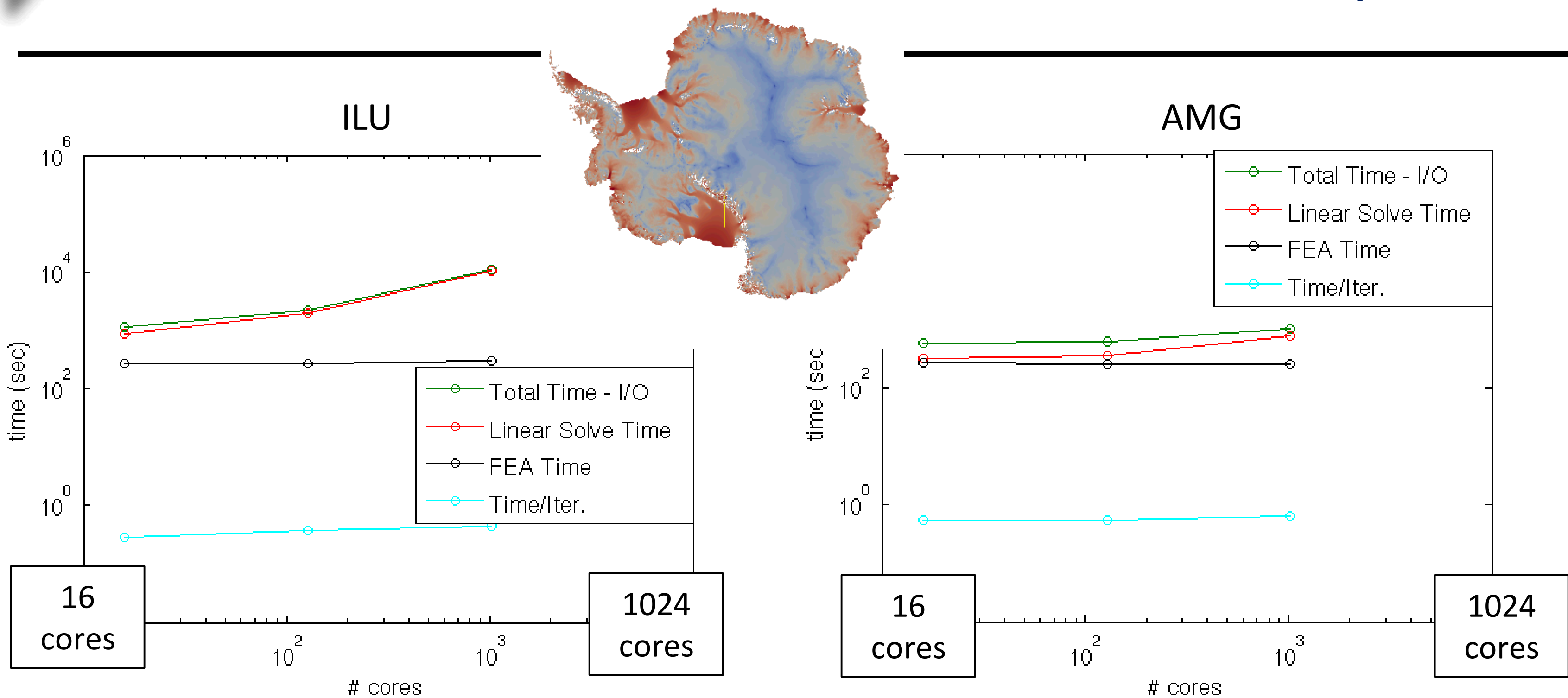


⇒ Ice shelves give rise to severe ill-conditioning of linear systems!

*Completely singular in the presence of islands and some ice tongues.

Sandia National Laboratories

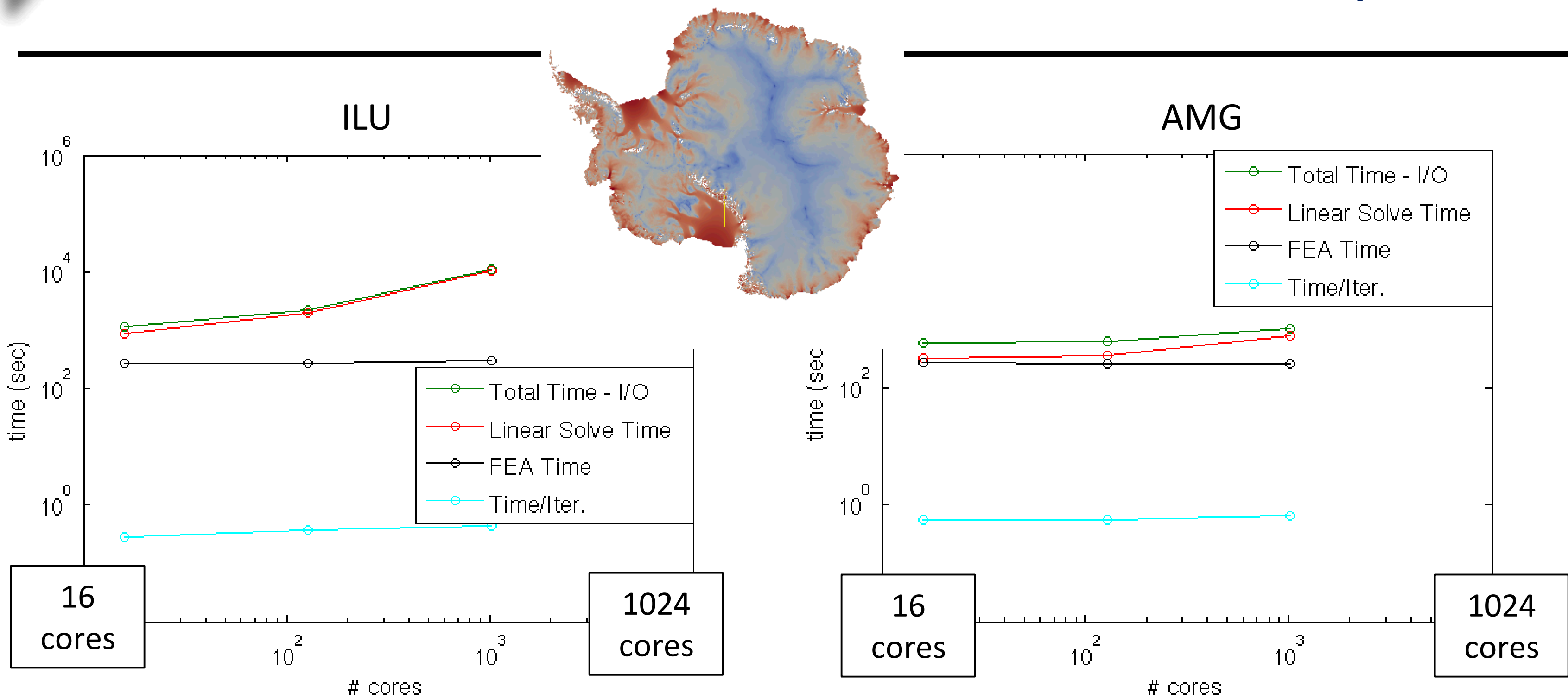
Weak Scaling Study for a Fine-Resolution AIS Problem (cont'd)



ILU vs. AMG:

- ILU solver > 10× slower than AMG solver on 1024 core problem.
  - Due to extremely poor convergence of ILU solver (~700 iterations/solve) → resulting from ill-conditioning of underlying linear systems.
- AMG iterations do grow as problem refined (14.4 iterations/solve on 16 cores vs. 35.5 iterations/solve on 1024 cores), but it is better suited to linear systems associated with AIS.

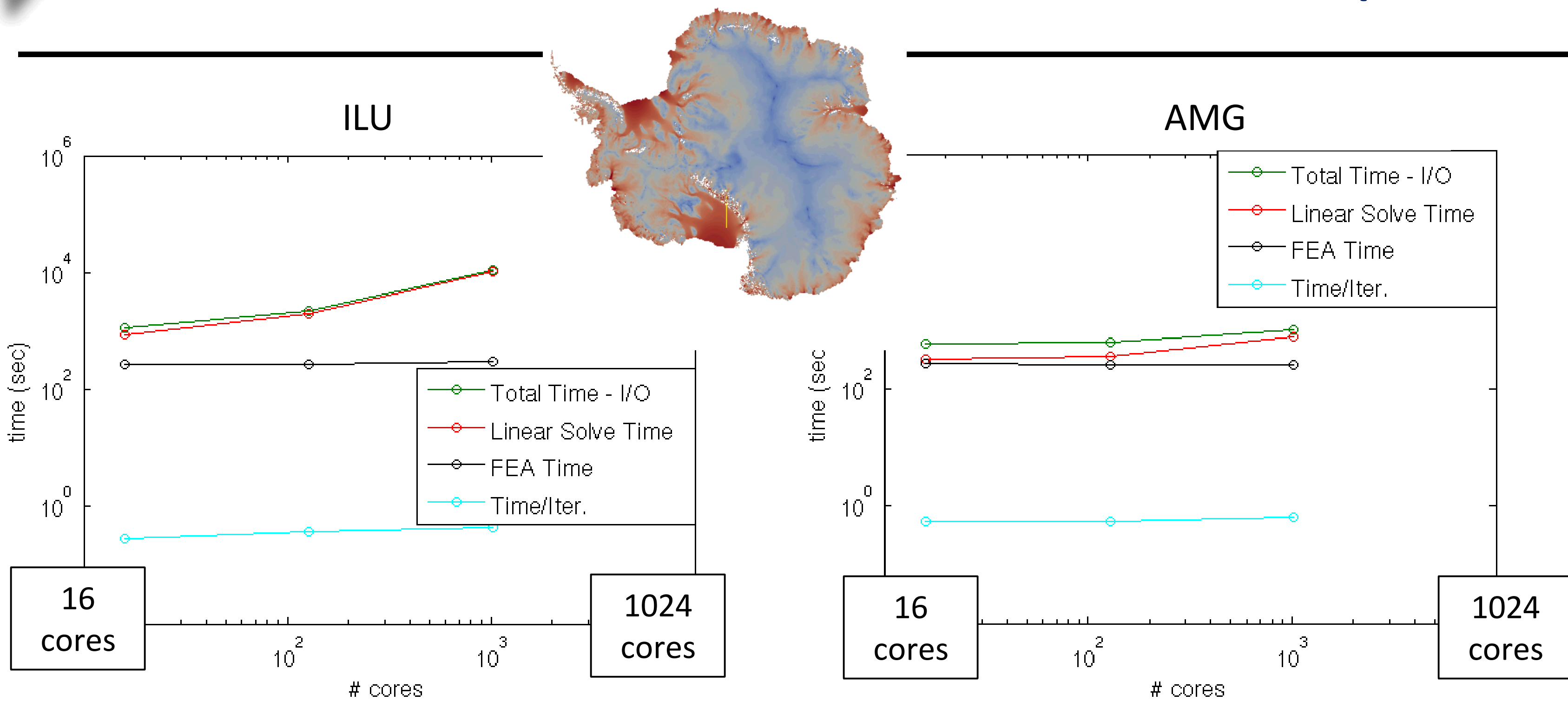
Weak Scaling Study for a Fine-Resolution AIS Problem (cont'd)



GMRES vs. CG:

- GMRES solver found to be more effective than CG, even though problem is symmetric.
  - We believe GMRES is somewhat less sensitive to rounding errors associated with the severe ill-conditioning induced by the presence of ice shelves.
  - GMRES and CG minimize different norms.

Weak Scaling Study for a Fine-Resolution AIS Problem (cont'd)



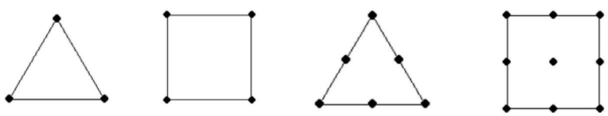
Summary:

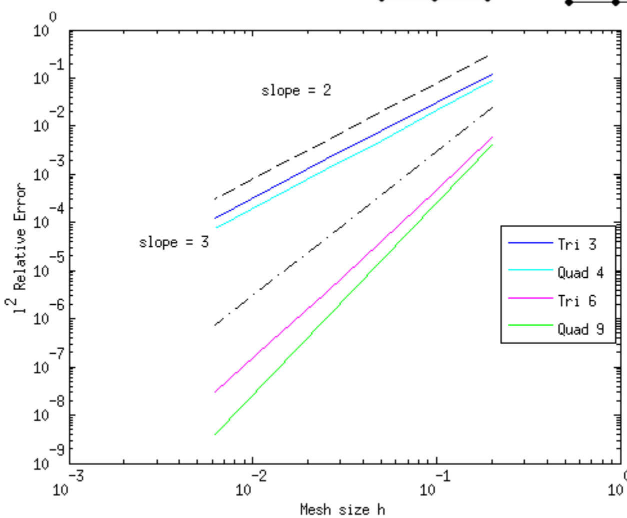
- Severe ill-conditioning caused by ice shelves!
- GMRES less sensitive than CG to rounding errors from ill-conditioning [also minimizes different norm].
- AMG preconditioner less sensitive than ILU to ill-conditioning.

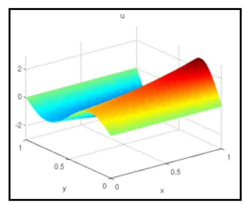
(vertical > horizontal coupling)
+
Neumann BCs
=
nearly singular submatrix associated with vertical lines

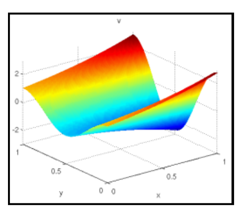
Appendix: Verification/Mesh Convergence Studies

Stage 1: solution verification on 2D MMS problems we derived.

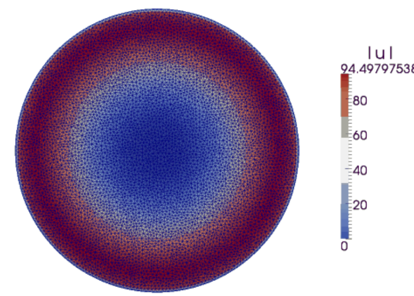




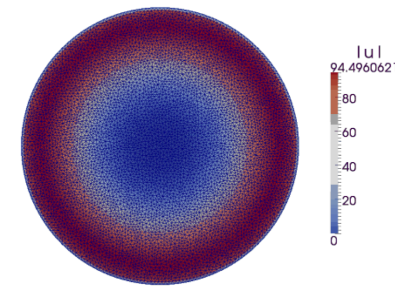




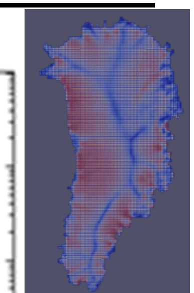
Stage 2: code-to-code comparisons on canonical ice sheet problems.

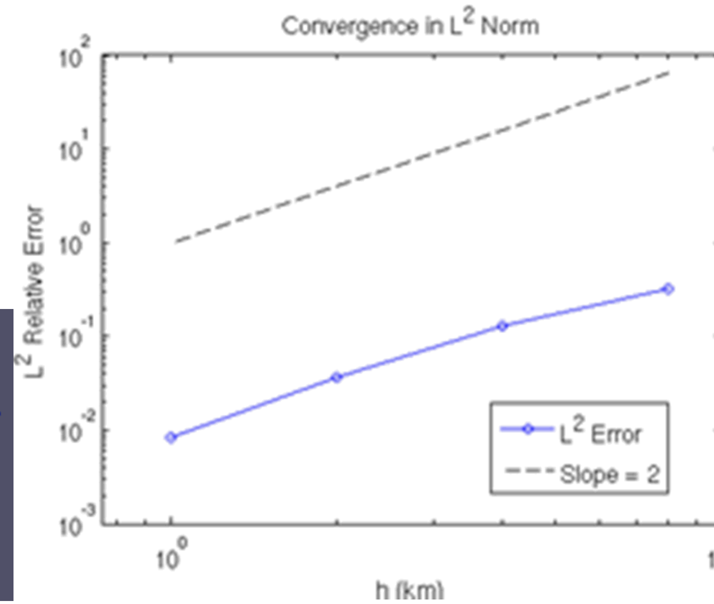


Albany/FELIX

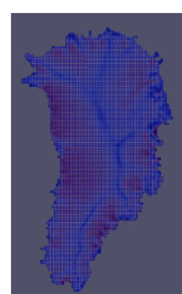


LifeV



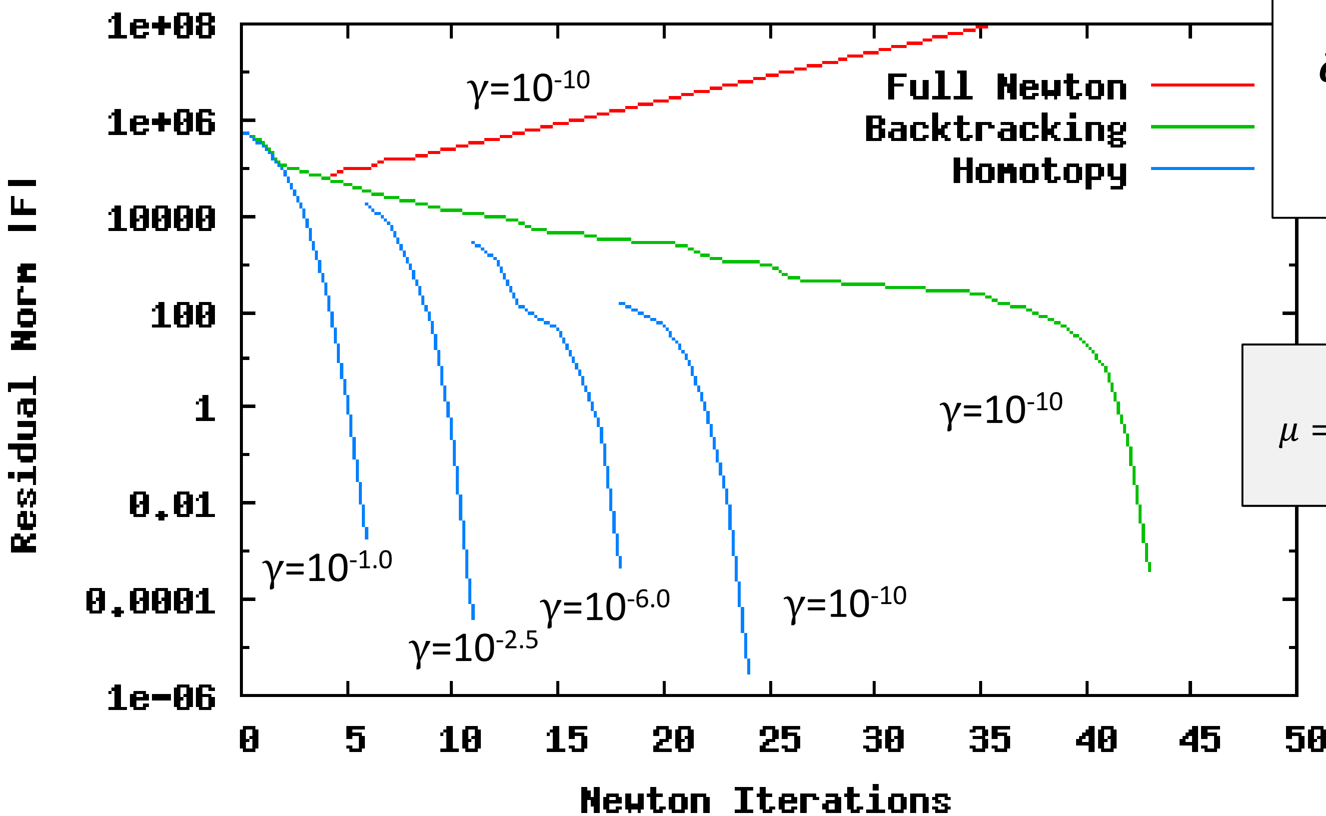


Stage 3: full 3D mesh convergence study on Greenland w.r.t. reference solution.



Are the Greenland problems resolved?
Is theoretical convergence rate achieved?

Appendix: Robustness of Newton's Method via Homotopy Continuation (LOCA)



$$\dot{\boldsymbol{\epsilon}}_1{}^T = (2\dot{\epsilon}_{11} + \dot{\epsilon}_{22}, \dot{\epsilon}_{12}, \dot{\epsilon}_{13})$$
$$\dot{\boldsymbol{\epsilon}}_2{}^T = (2\dot{\epsilon}_{12}, \dot{\epsilon}_{11} + 2\dot{\epsilon}_{22}, \dot{\epsilon}_{23})$$
$$\dot{\epsilon}_{\mathrm{ij}} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial uj}{\partial xi}\right)$$

Glen's Law Viscosity:

$$\mu = \frac{1}{2} A^{-\frac{1}{n}} \left(\frac{1}{2}\sum_{ij} \dot{\boldsymbol{\epsilon}}_{ij}{}^2 + \gamma\right)^{\left(\frac{1}{2n} - \frac{1}{2}\right)}$$

γ = regularization parameter

$n = 3$
(Glen's law exponent)

- Newton's method most robust with full step + homotopy continuation of $\gamma \rightarrow 10^{-10}$: converges out-of-the-box!

Sandia National Laboratories