Northumbria University NEWCASTLE

Úa

Lessons Learned from Model Developments Involving Ice-Ocean Interactions

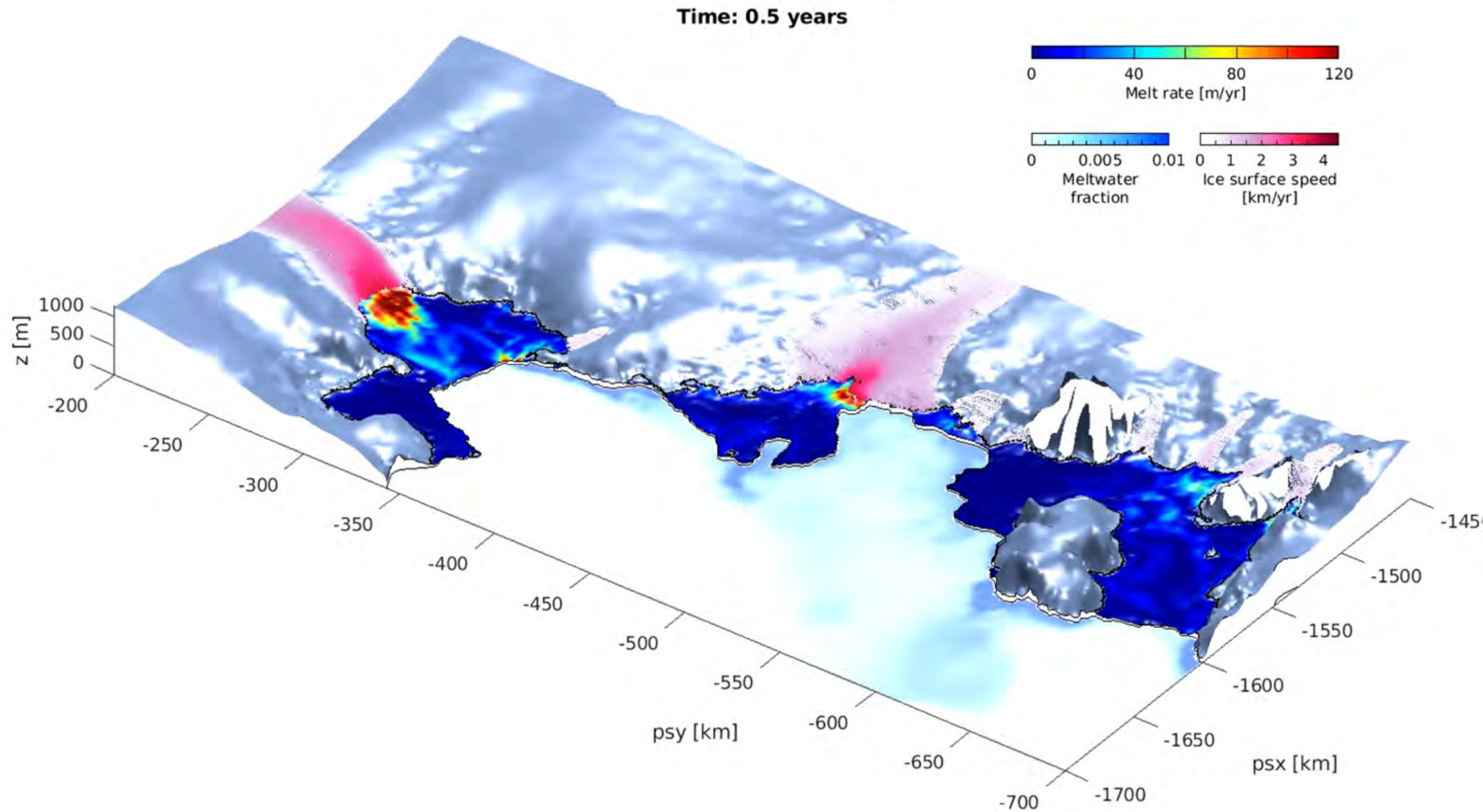
The Future of Earth System Modeling: Polar Climate
Caltech

29 Nov, 2018.

G. Hilmar Gudmundsson,
Northumbria University
Newcastle, UK

Úa+MITgcm

An example of a coupeld ice+ocean run of Pine Island and Thwaites Glaciers

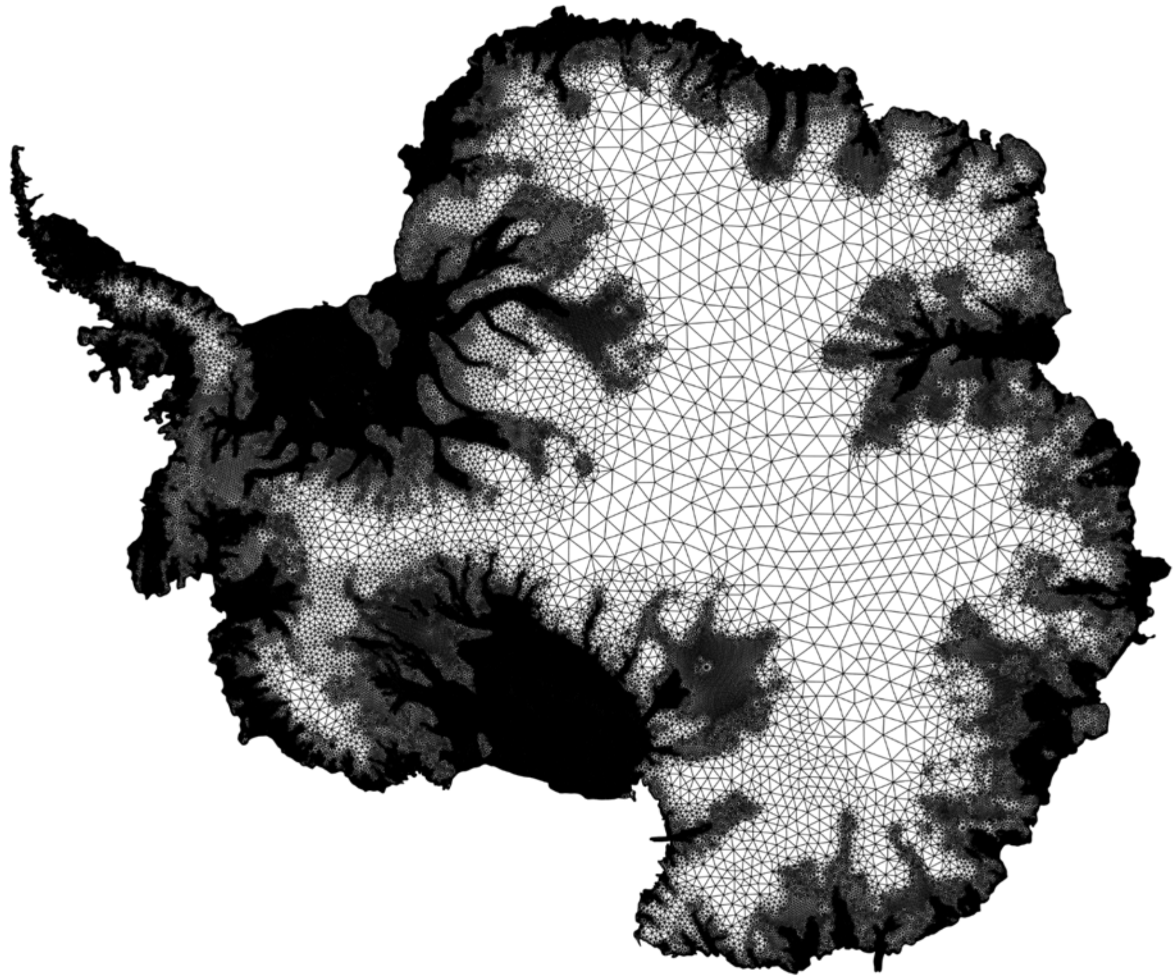


(Ongoing work by Jan DeRydt)

Structure

1. Some of the things I think are important to have in an ice-flow model and why.
2. Ice/Ocean interactions, what level of complexity is required?
3. Grounding-line dynamics and ice flux.
4. Lessons learned.
5. Future direction.

Unstructured meshes and automated mesh refinement
(local and global h adaptivity, p adaptivity)



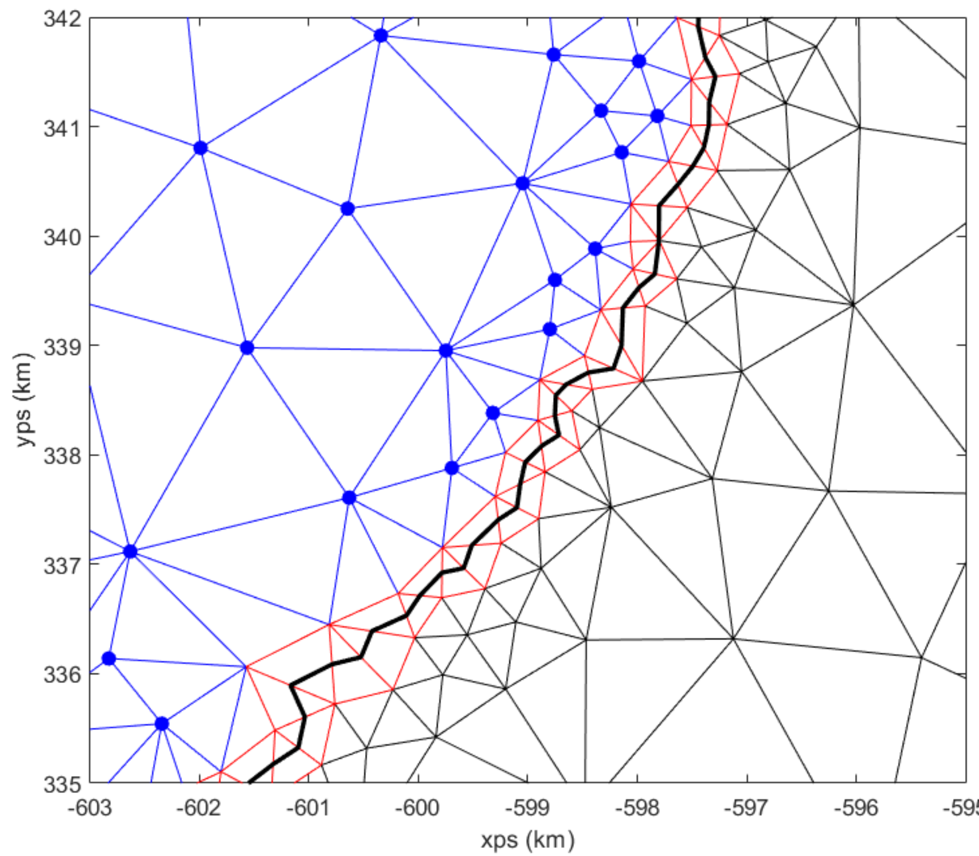
Unstructured meshes and automated mesh refinement (local and global h adaptivity, p adaptivity)

A close-up of a finite-element mesh of a Antarctica showing part of the grounding-line area of the Foundation Ice Stream.

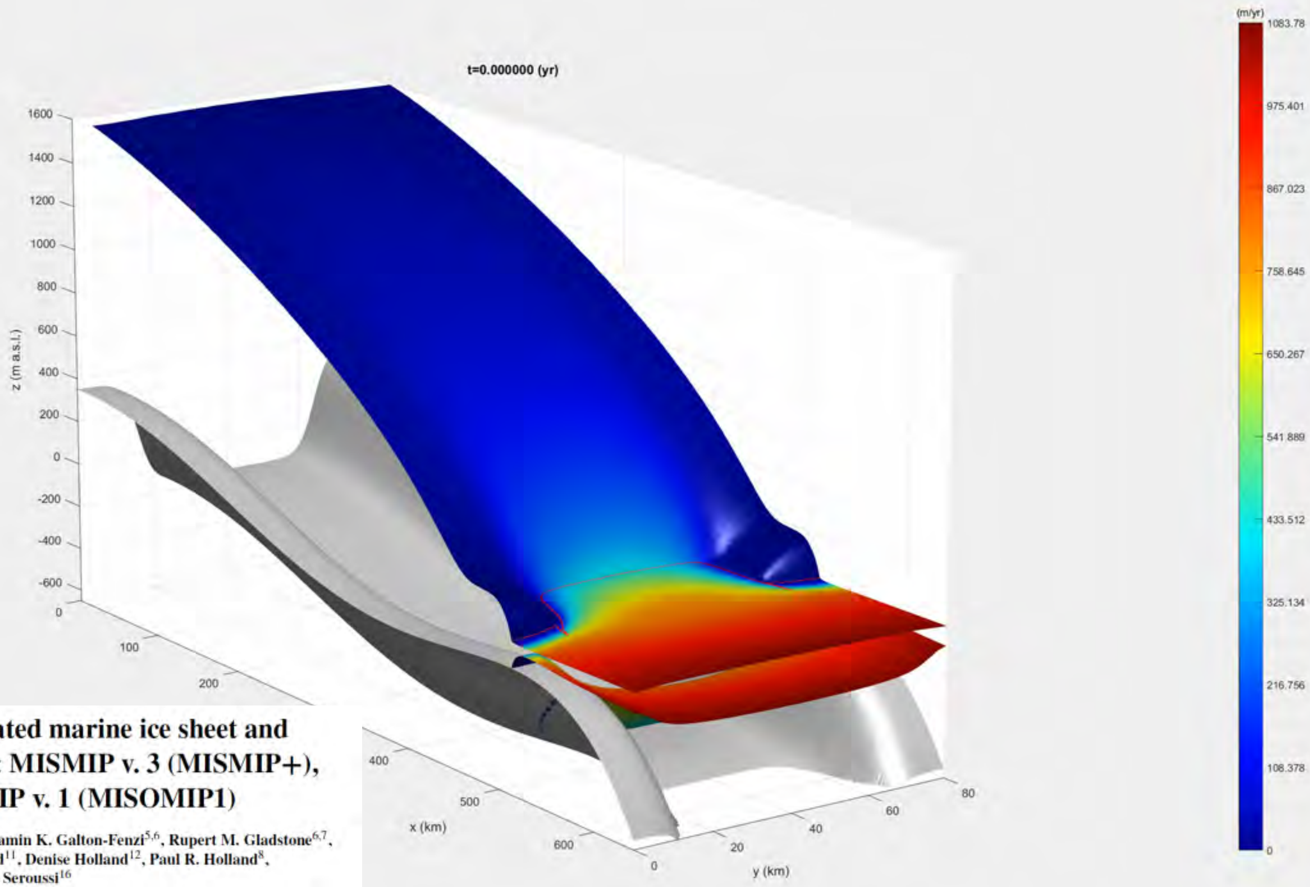
Elements over grounded and floating areas are shown in black and blue, respectively.

The grounding line is shown as a thick black line.

Elements with nodes located on both sides of the grounding line are in red.

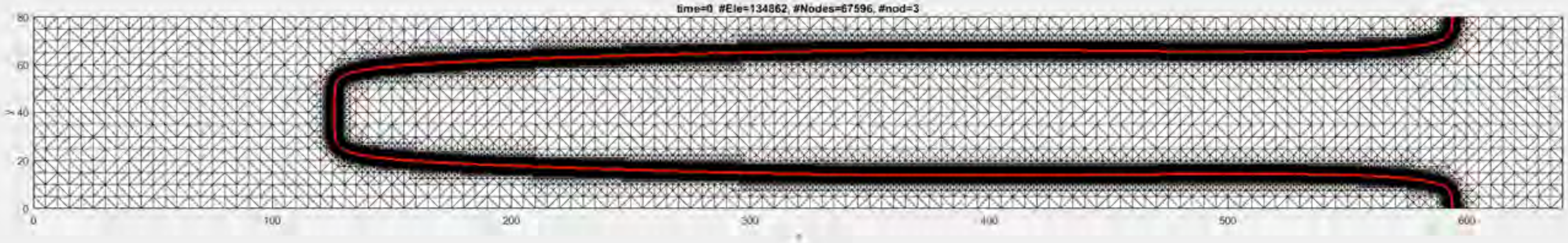


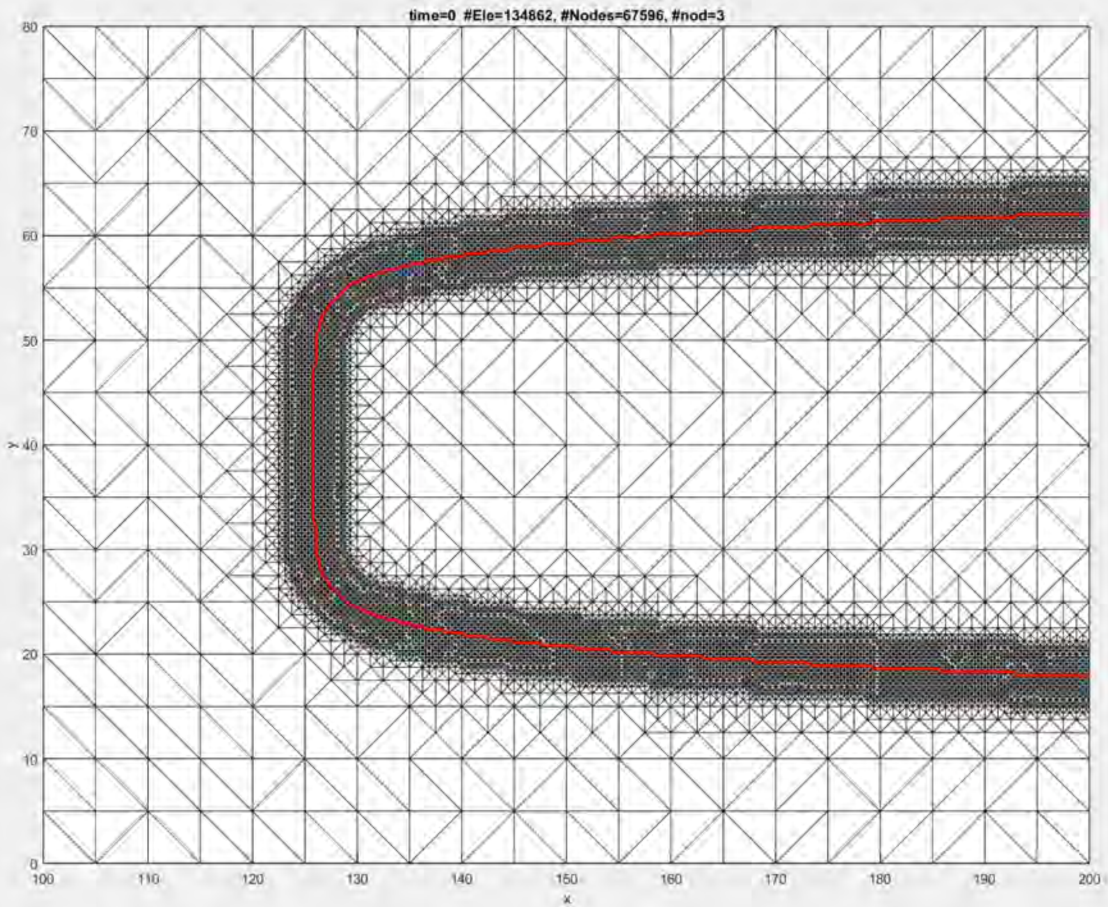
MISMIP+ 1rr intercomparison experiment



Experimental design for three interrelated marine ice sheet and ocean model intercomparison projects: MISMIP v. 3 (MISMIP+), ISOMIP v. 2 (ISOMIP+) and MISOMIP v. 1 (MISOMIP1)

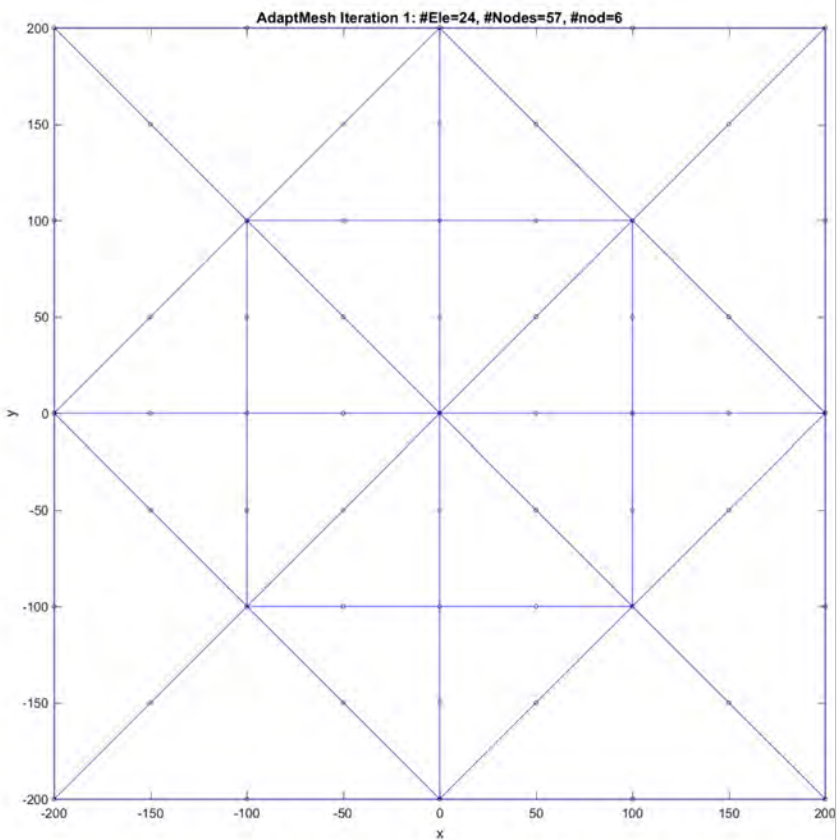
Xylar S. Asay-Davis[1], Stephen L. Cornford[2], Gaël Durand[3,4], Benjamin K. Galton-Fenzi[5,6], Rupert M. Gladstone[6,7], G. Hilmar Gudmundsson[8], Tore Hattermann[9,10], David M. Holland[11], Denise Holland[12], Paul R. Holland[8], Daniel F. Martin[13], Pierre Mathiot[8,14], Frank Pattyn[15], and Hélène Seroussi[16]


time=0 #Ele=134862, #Nodes=67596, #nod=3

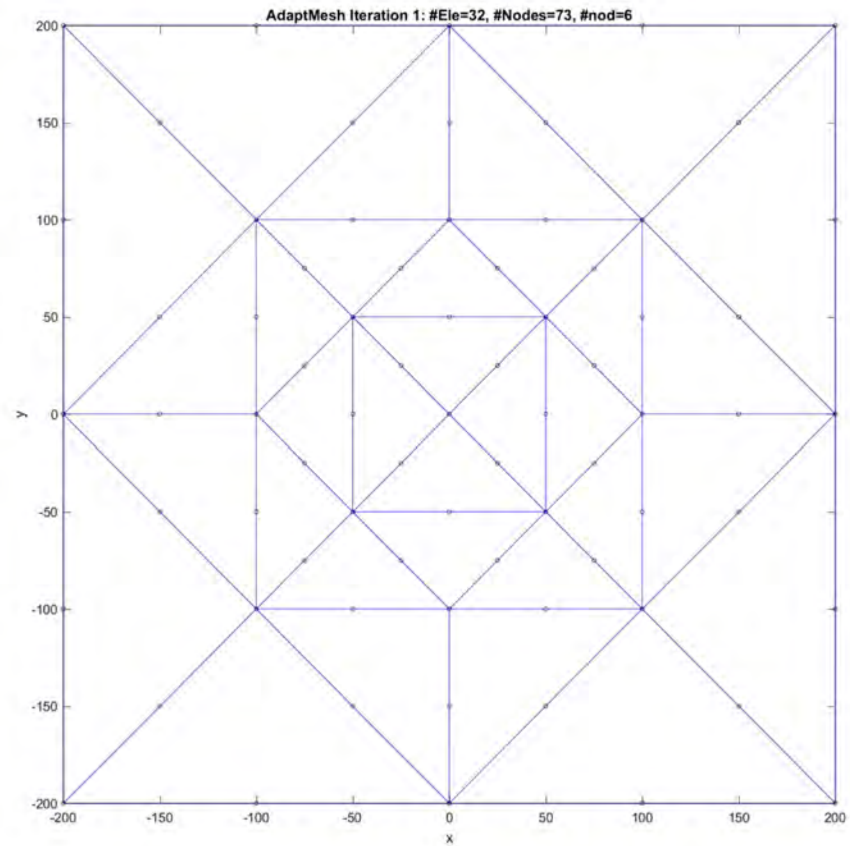

time=0 #Ele=134862, #Nodes=67596, #nod=3
y
x

Local mesh refinement:

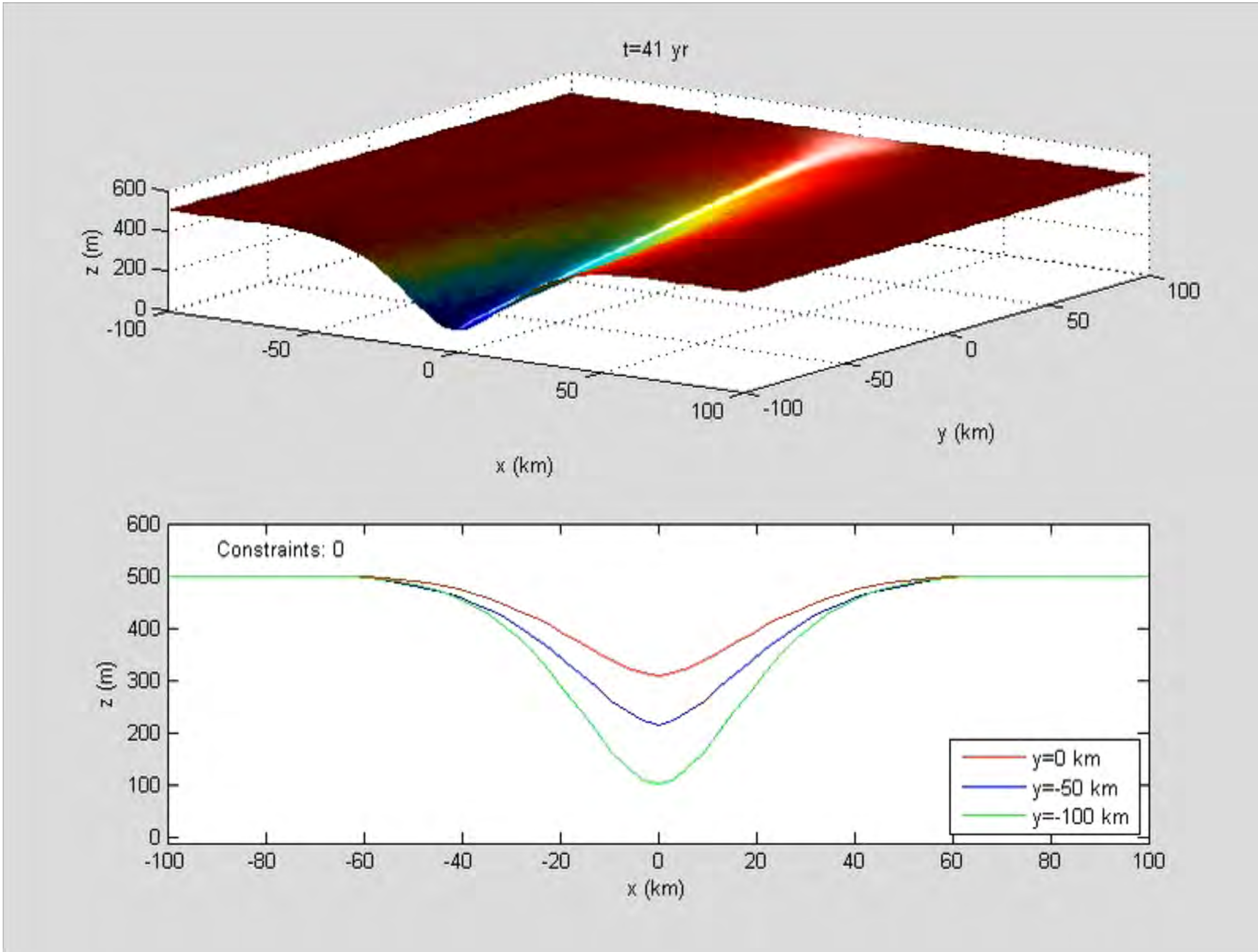
Newest Vertex Bisection

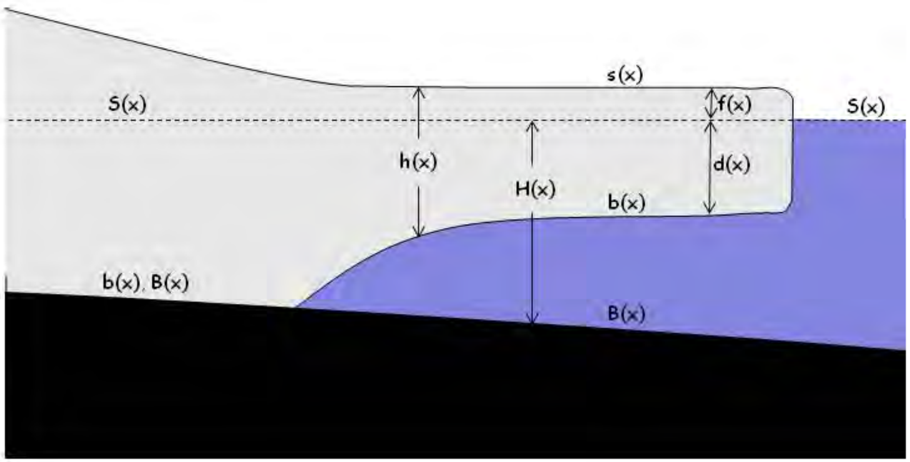


Red-Green



Automated thickness constraints



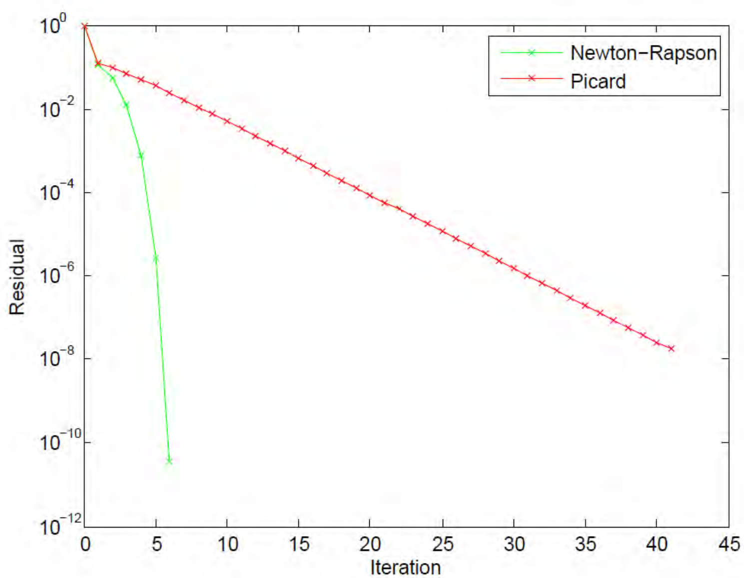


The numerical models solves

$$\nabla_h \boldsymbol{\sigma}_h + \boldsymbol{t}_b = \rho g h \nabla_h s$$

where

$$\boldsymbol{\sigma}_h = \begin{pmatrix} 2\tau_{xx} + \tau_{yy} & \tau_{xy} \\ \tau_{xy} & 2\tau_{yy} + \tau_{xx} \end{pmatrix}$$



Transient evolution solved implicitly for u, v and h using a Newton-Raphson solver.

$$\begin{bmatrix} K^{xu} & K^{xv} & K^{xh} \\ K^{yu} & K^{yv} & K^{yh} \\ K^{hu} & K^{hv} & K^{hh} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{v} \\ \Delta\mathbf{h} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_u \\ \mathbf{r}_v \\ \mathbf{r}_h \end{bmatrix}$$

Forward model:

$$F_x = \partial_x(4h\eta\partial_x u + 2h\eta\partial_y v) + \partial_y(h\eta(\partial_x v + \partial_y u)) - C^{-1/m}\,|\boldsymbol{u}|^{1/m-1}\,u - \rho g h \partial_x s$$
$$F_y = \partial_y(4h\eta\partial_y v + 2h\eta\partial_x u) + \partial_x(h\eta(\partial_y u + \partial_x v)) - C^{-1/m}\,|\boldsymbol{u}|^{1/m-1}\,v - \rho g h \partial_y s$$

Function to be minimized:

$$I = I_{\text{misfit}}(u,v,w) + I_{reg}(A,C) + I_{\text{barrier}}(A,C)) + < F_x(u,v), \lambda > + < F_y(u,v), \mu > + < w - G(u,v), \kappa >$$

Step 1: Solve forward problem

$$F_x(u,v) = 0$$
$$F_y(u,v) = 0$$

Step 2: Solve adjoint problem

$$\kappa \mathbf{1} = -\partial_w I_{\text{misfit}}$$

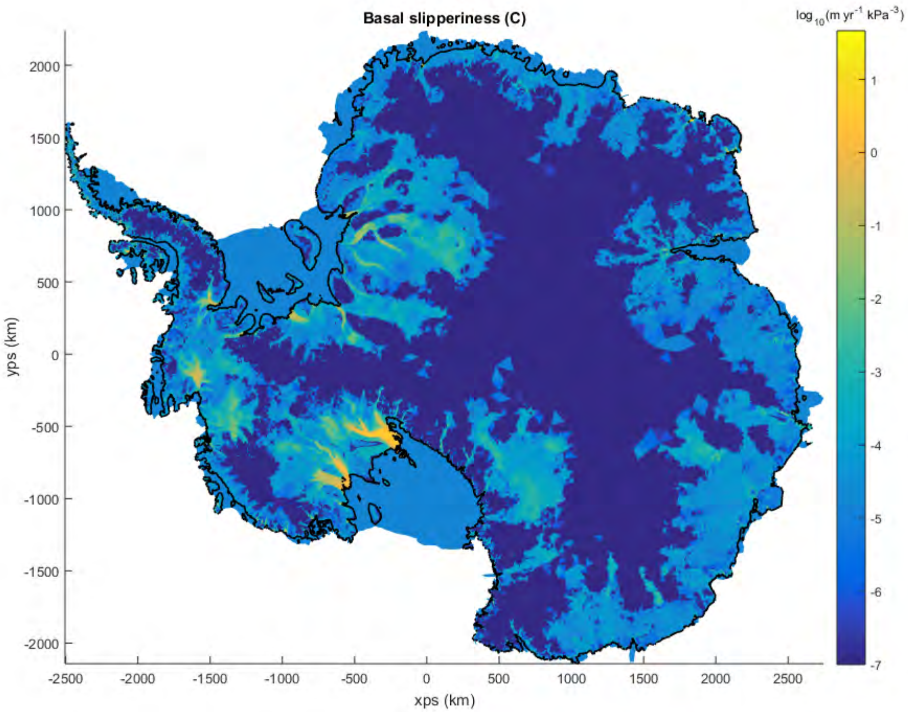
$$\begin{bmatrix} \partial_{\boldsymbol{u}} \boldsymbol{F}_x & \partial_{\boldsymbol{u}} \boldsymbol{F}_y \\ \partial_v \boldsymbol{F}_x & \partial_{\boldsymbol{v}} \boldsymbol{F}_y \end{bmatrix}^T \begin{bmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\mu} \end{bmatrix} = \begin{bmatrix} -\partial_{\boldsymbol{u}} I_{\text{misfit}} + \partial_{\boldsymbol{u}} \boldsymbol{G}^T \kappa \\ -\partial_{\boldsymbol{v}} I_{\text{misfit}} + \partial_{\boldsymbol{v}} \boldsymbol{G}^T \kappa \end{bmatrix}$$

Step 3: Insert to obtain gradient

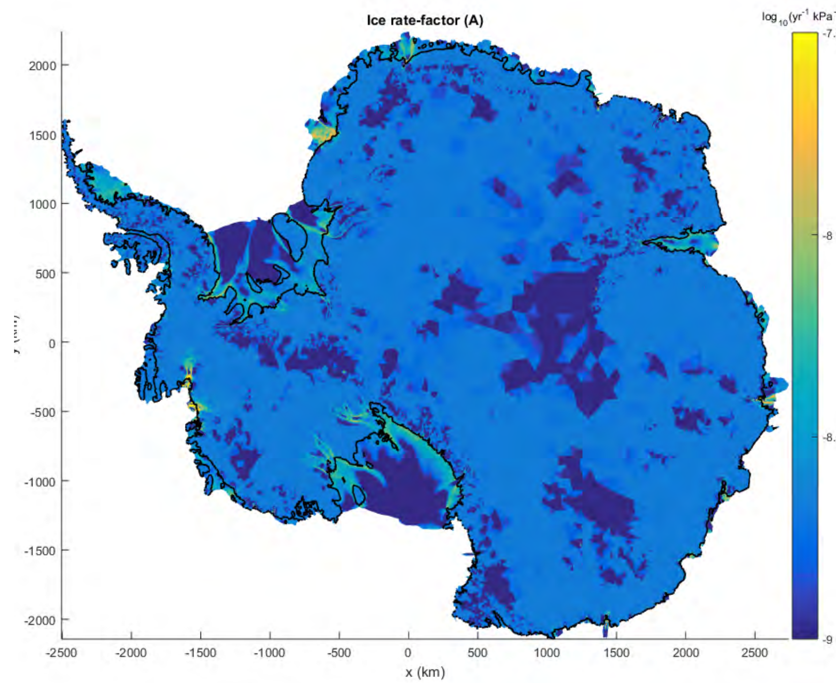
$$\frac{\partial I}{\partial C_p} = \frac{\partial I_{\text{reg}}}{\partial C_p} + \frac{\partial I_{\text{barrier}}}{\partial C_p} + \left\langle \frac{\partial F_x}{\partial C_q}, \lambda \right\rangle + \left\langle \frac{\partial F_y}{\partial C_q}, \mu \right\rangle$$

Data assimilation: solving for basal slipperiness, englacial viscosity, and ice thickness. Gradient based optimisation with gradients calculated using the adjoint method

Slipperiness estimate



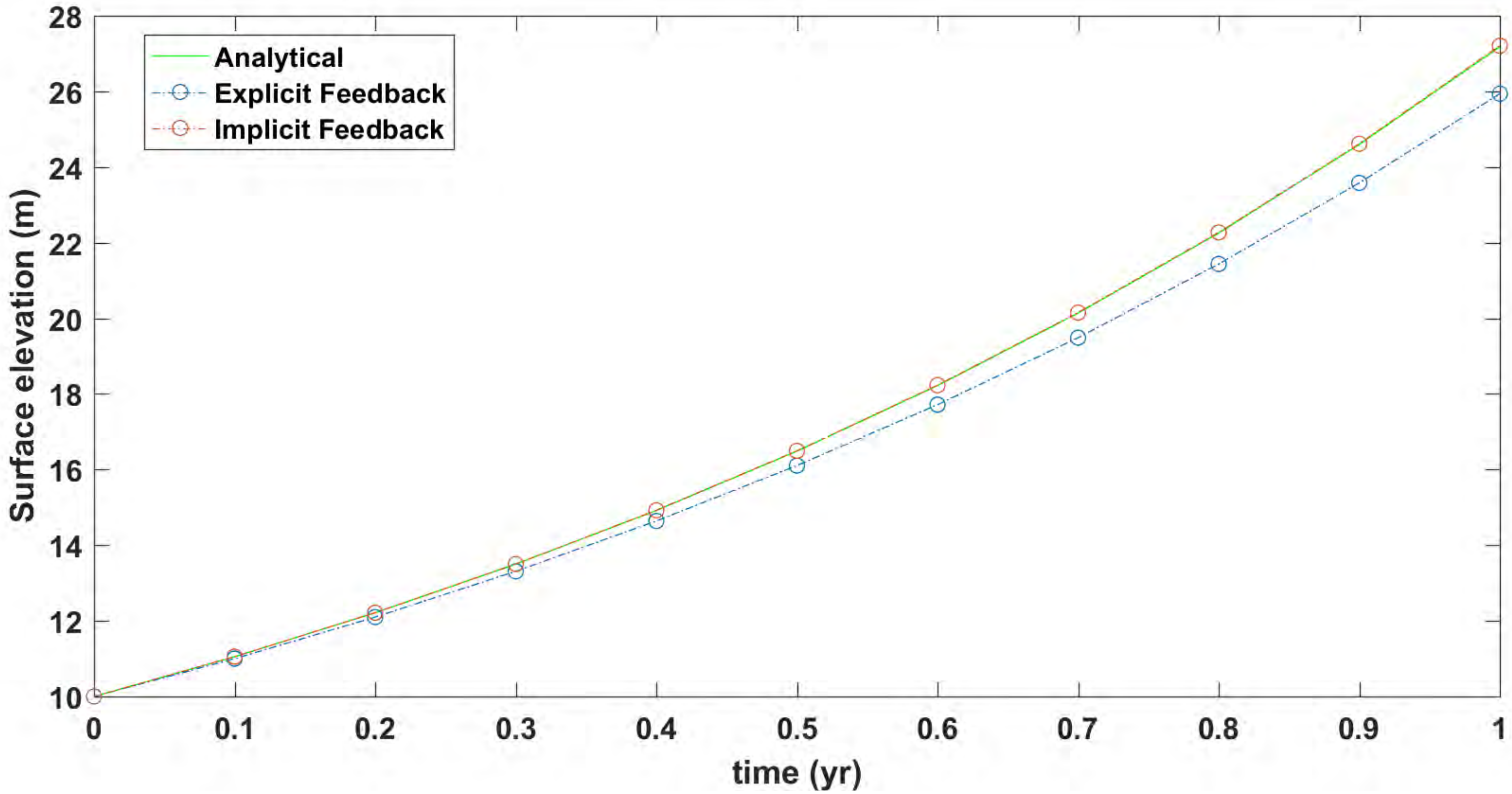
Ice rate factor



Implicit and explicit mass-balance/elevation feedback

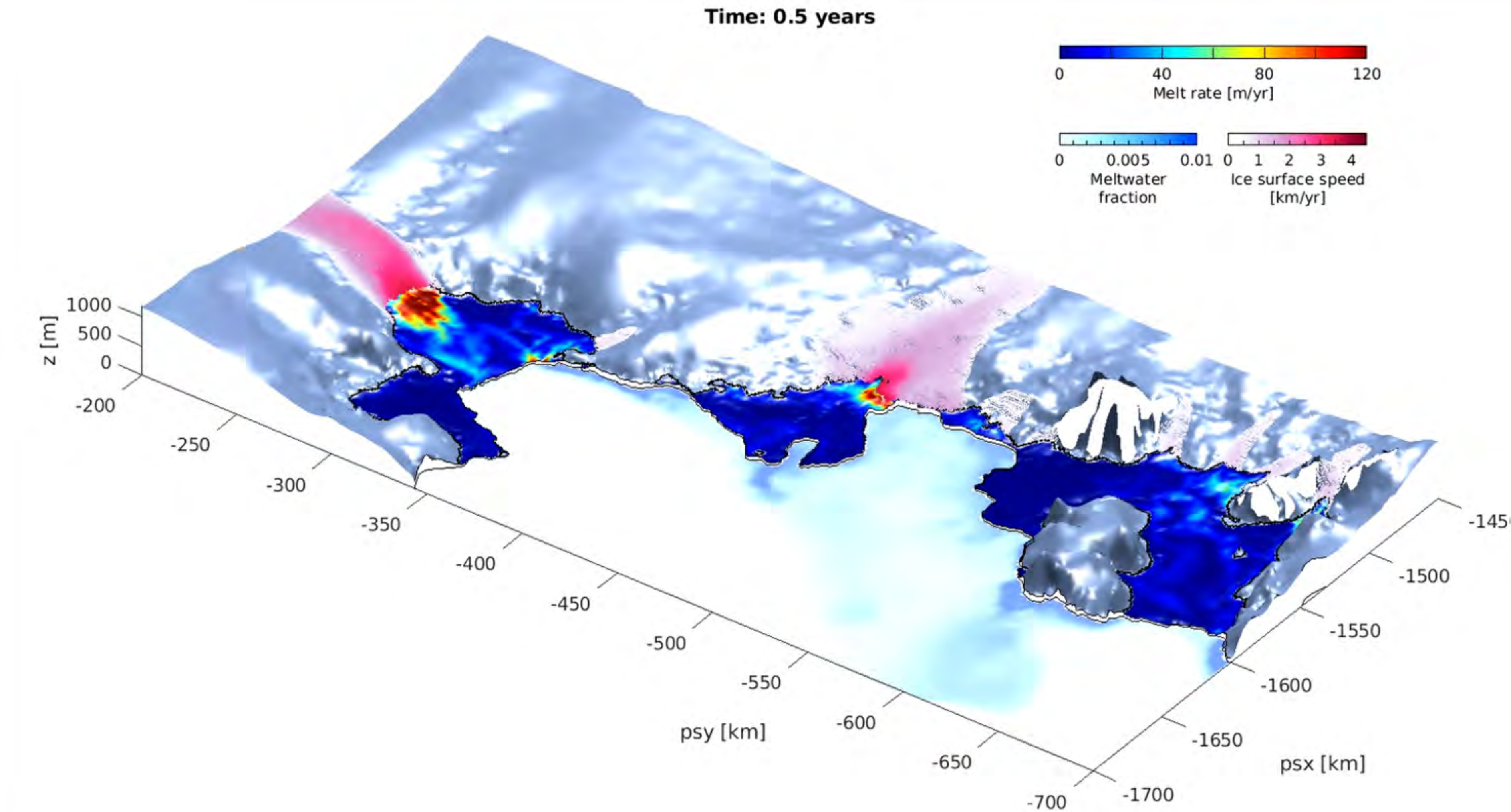
Implicit: h(t)=h(a(h(t)),….)

Explicit: h(t)=h(a(h(t-dt))



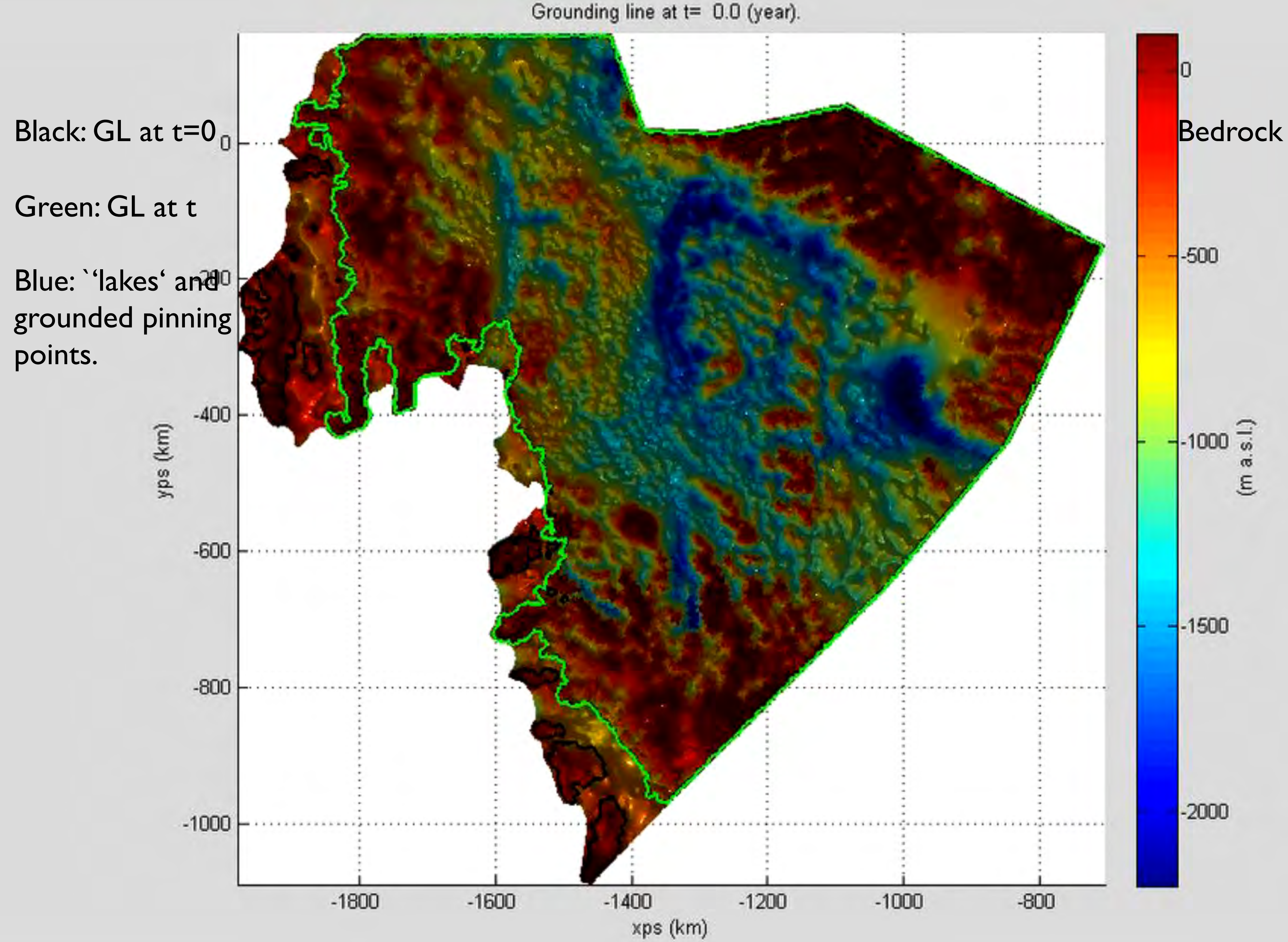
Úa+MITgcm

An example of a coupeld ice+ocean run of Pine Island and Thwaites Glaciers

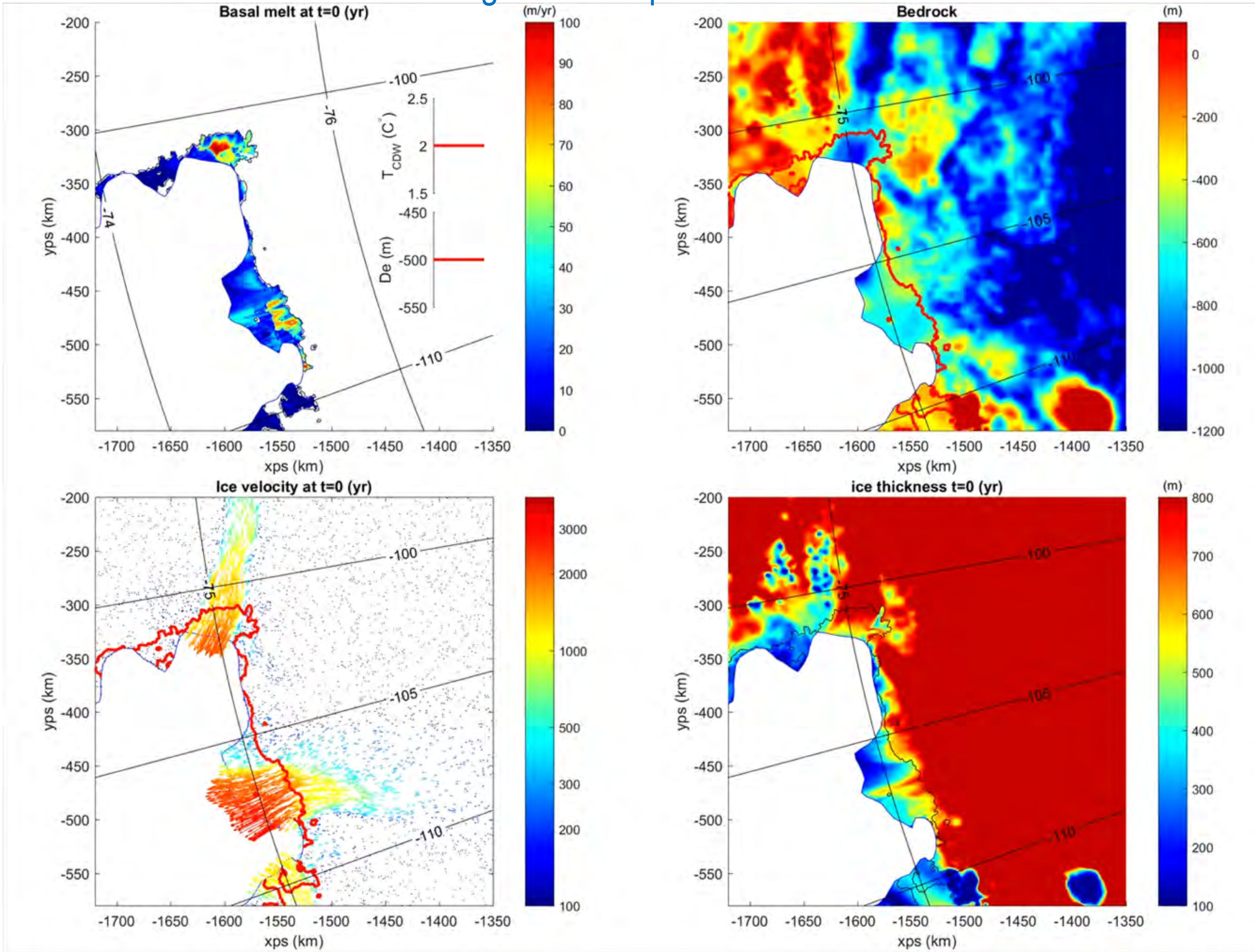


(Ongoing work by Jan DeRydt)

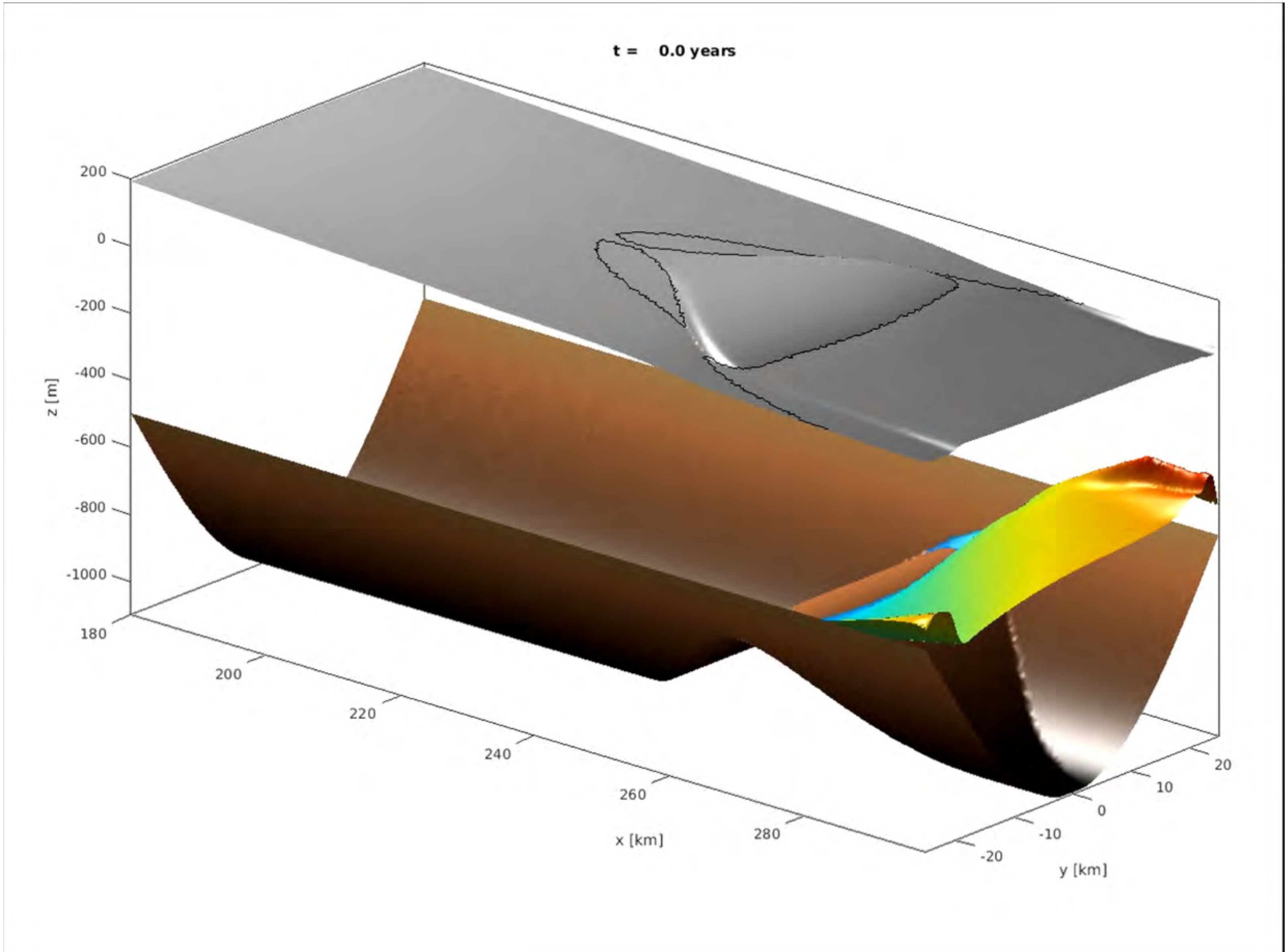
`Melt-Rate 3' from Favier et al (2014) calculated with Úa



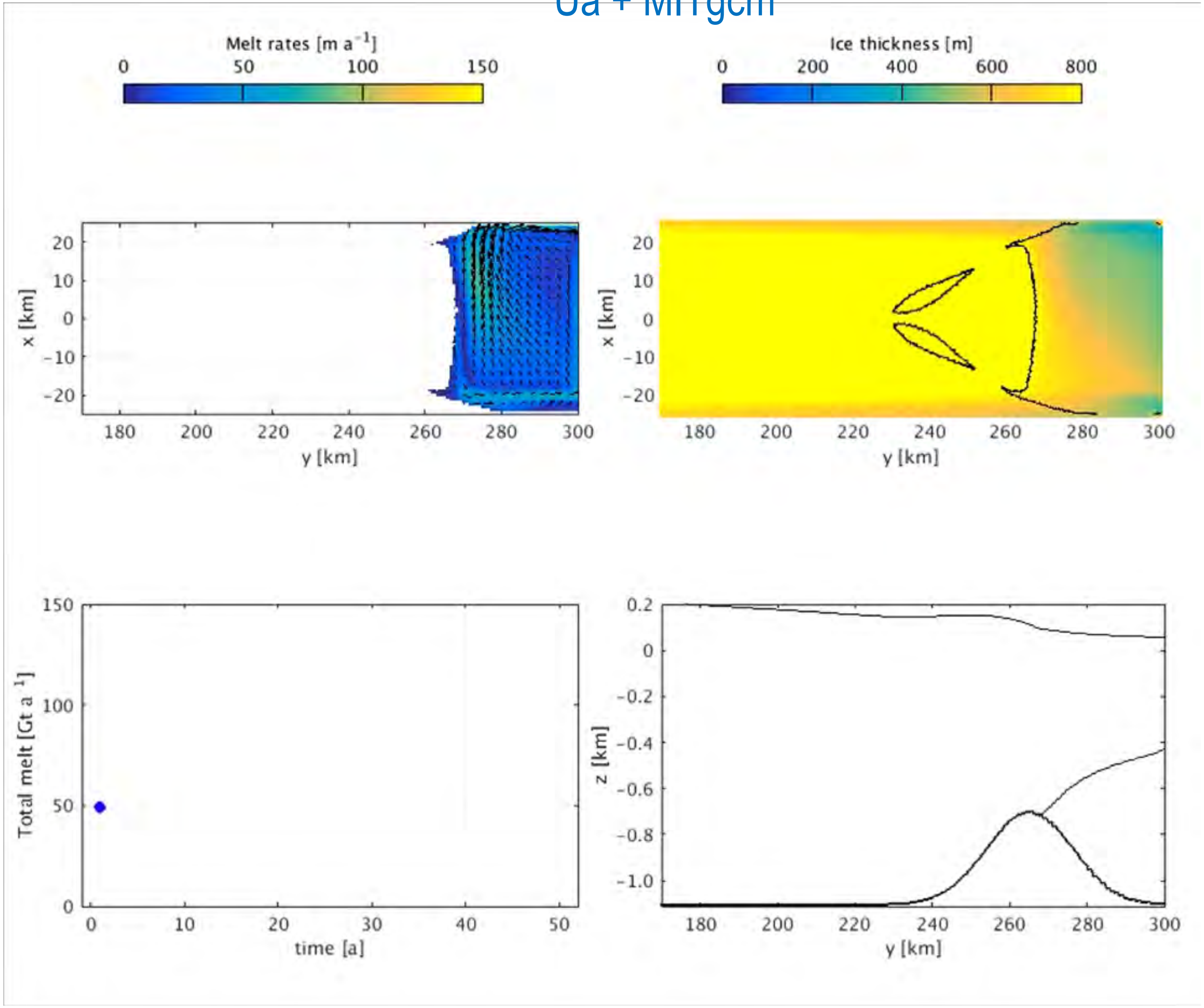
Pine Island Glacier and Thwaites glacier coupled ice-ocean run using an ocean plume model



Coupled ice-ocean run
Úa + MITgcm



Coupled ice-ocean run
Úa + MITgcm



Grounding-line flux formula applied as a flux condition in numerical simulations fails for buttressed Antarctic ice streams

Ronja Reese[1,2], **Ricarda Winkelmann**[1,2], **and G. Hilmar Gudmundsson**[3]

[1]Potsdam Institute for Climate Impact Research (PIK), Member of the Leibniz Association,
P.O. Box 60 12 03, 14412 Potsdam, Germany
[2]University of Potsdam, Institute of Physics and Astronomy, Karl-Liebknecht-Str. 24–25, 14476 Potsdam, Germany
[3]Department of Geography and Environmental Sciences, Northumbria University, Newcastle upon Tyne, NE1 8ST, UK

Correspondence: Ricarda Winkelmann (ricarda.winkelmann@pik-potsdam.de)

Received: 30 December 2017 – Discussion started: 15 January 2018
Revised: 16 August 2018 – Accepted: 26 August 2018 – Published: 9 October 2018

$$\boxed{q(x) = \theta^{\frac{nm}{m+1}} \rho_\mathrm{i} h^{\frac{1+m(n+3)}{m+1}} \left(\frac{1}{4^n} A(\rho_\mathrm{i} g)^{n+1} (1-\rho_\mathrm{i}/\rho_\mathrm{w})^n C^{1/m} \right)^{\frac{m}{m+1}}}$$

$$\theta_1 = \frac{\boldsymbol{n}_1 \cdot \mathbf{R}\boldsymbol{n}_1}{2\tau_\mathrm{f}} \qquad \theta_2 = \frac{\boldsymbol{n}_1 \cdot \boldsymbol{\tau}\boldsymbol{n}_1}{\tau_\mathrm{f}} \qquad \theta_3 = \frac{\boldsymbol{n}_2 \cdot \boldsymbol{\tau}\boldsymbol{n}_2}{\tau_\mathrm{f}}$$

$$\boldsymbol{\tau}_\mathrm{b} = C^{-1/m} |\boldsymbol{v}_\mathrm{b}|^{1/m-1} \boldsymbol{v}_\mathrm{b}$$

$$\mathbf{R} = \begin{pmatrix} 2\tau_{xx} + \tau_{yy} & \tau_{xy} \\ \tau_{xy} & \tau_{xx} + 2\tau_{yy} \end{pmatrix}$$

$$\dot{\epsilon}_{ij} = A\tau^{n-1}\tau_{ij}$$

$$\boldsymbol{\tau} = \begin{pmatrix} \tau_{xx} & \tau_{xy} \\ \tau_{xy} & \tau_{yy} \end{pmatrix}$$

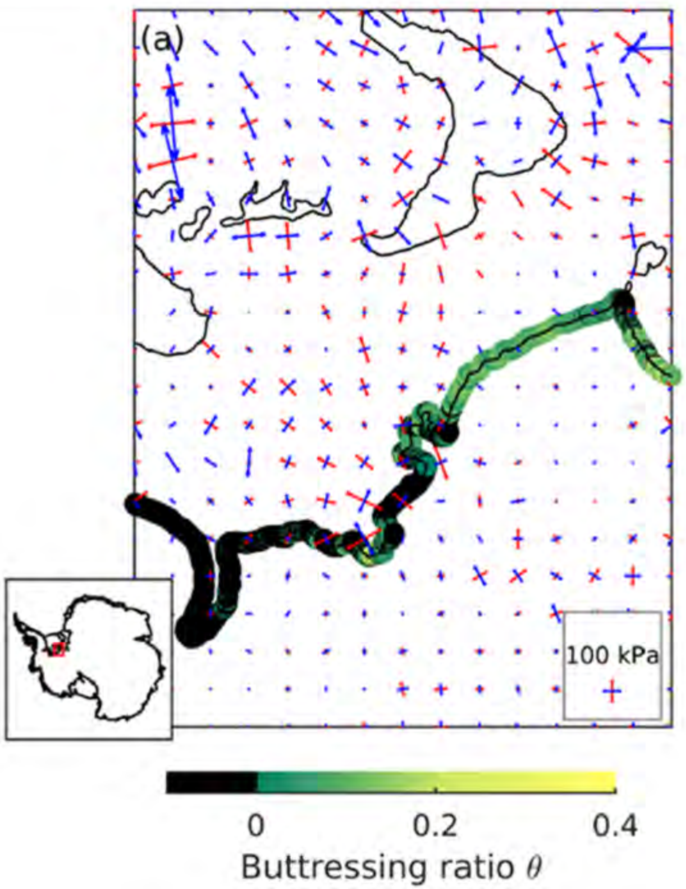
Grounding-line flux formula applied as a flux condition in numerical simulations fails for buttressed Antarctic ice streams

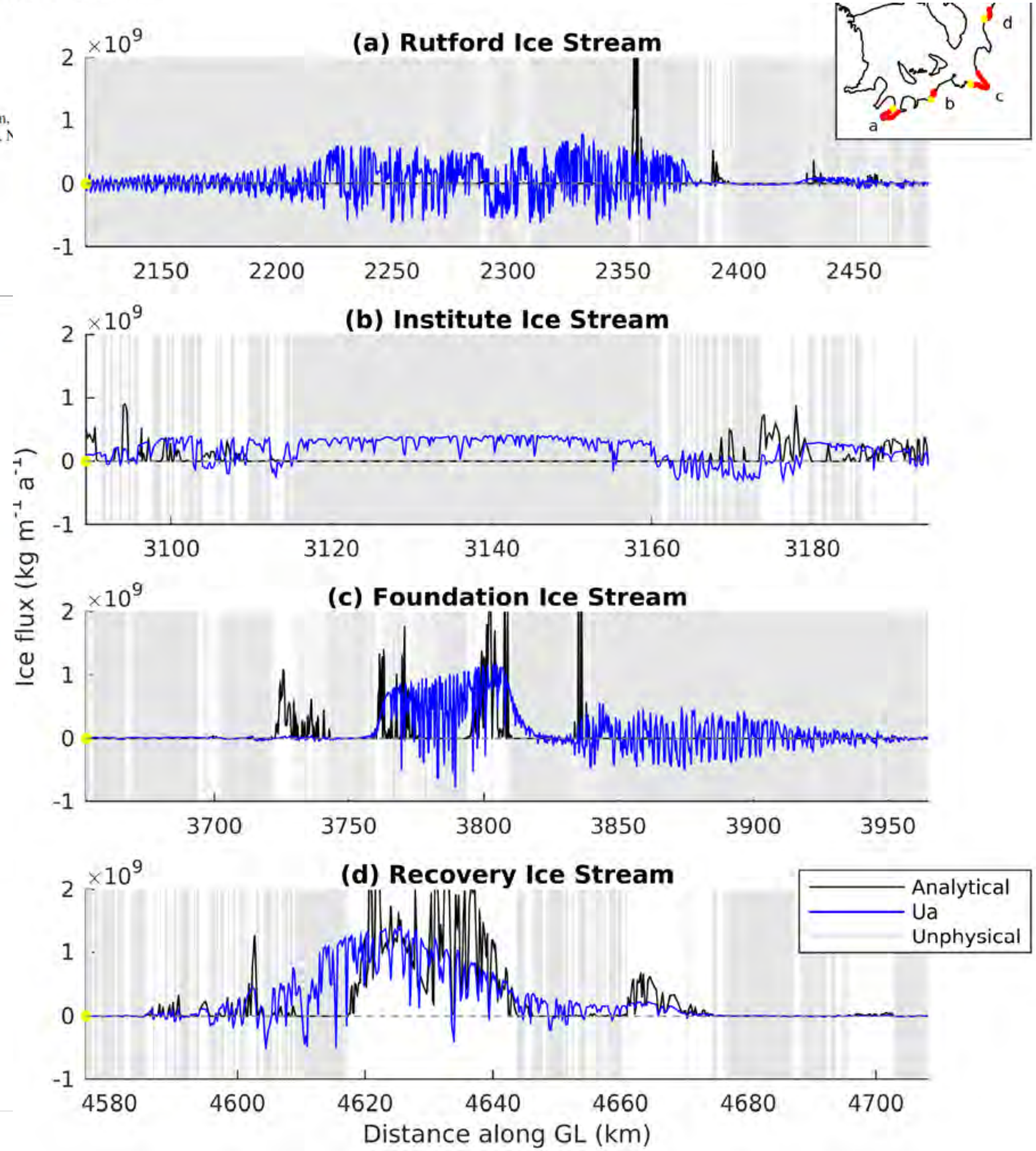
Ronja Reese[1,2], **Ricarda Winkelmann**[1,2], **and G. Hilmar Gudmundsson**[3]

[1]Potsdam Institute for Climate Impact Research (PIK), Member of the Leibniz Association, P.O. Box 60 12 03, 14412 Potsdam, Germany
[2]University of Potsdam, Institute of Physics and Astronomy, Karl-Liebknecht-Str. 24–25, 14476 Potsdam,
[3]Department of Geography and Environmental Sciences, Northumbria University, Newcastle upon Tyne, N

Correspondence: Ricarda Winkelmann (ricarda.winkelmann@pik-potsdam.de)

Received: 30 December 2017 – Discussion started: 15 January 2018
Revised: 16 August 2018 – Accepted: 26 August 2018 – Published: 9 October 2018





Úa is opensource

You can download Úa from github

`https://github.com/GHilmarG`

You can download the Úa repository as a single zip file from the github webpage.

If you have git installed locally on your computer, then you can clone the repository using:

```
git clone https://github.com/GHilmarG/UaSource
```

Once you've cloned the UaSource directory you can update it using `git pull`

Note: on github files are limited to 50Mb. For that reason the gmsh.exe is not part of the repository. If you want to use gmsh then you must download it separately.

Lessons learned

Lessons learned #1

Focus on algorithms!
Don’t worry about writing in a ’cool’ programming language.

Examples:

- NR solver is second order, any first-order algorithm (e.g. Piccard) will become slower with problem size irrespectively of programming langue.

- Fully implicit (ie uvh) forward time stepping combined with mesh-refinement is advantageous as it:
  a) imposes no CLF condition on the time step, i.e. time steps independent of mesh-resolution,
  b) in SSA this approach allows the grounding-line migration itself to be treated implicitly and eliminates an additional Piccard type loop.

Northumbria University NEWCASTLE

Lessons learned #2

Your time is more important than that of your computer/HPC facilities!

The things you don't have time to code up run infinitely slowly!

Lessons learned #3

Global and local mesh refinement is a good thing!

Why?

Many reasons, but mainly because most of the Antarctic Ice Sheet is rather boring (e.g. interior of East Antarctic) , but some are very exciting (grounding-line areas, margins).

Northumbria University NEWCASTLE

Lessons learned #4

Predicting what happens to the large ice sheets over the next 100 yr is more like predicting the weather than predicting the climate

Why?

System not in equilibrium, some adjustment processes slow (thermodynamics), others quick (mass redistribution, and GL migration).

Implications:

Data assimilation and inverse estimation of model parameters is important, so make sure this is done correctly and efficiently in your model (e.g. adjoints, analytical sensitives,…)

Estimate the slow processes, model the fast ones

Northumbria University NEWCASTLE

Lessons learned #5 (ice-ocean specific, Courtesy of Jan De Rydt)

Use tools you know,
don’t overthink the problem,
and work incrementally.

Why?

You only know what problems are really worth worrying about once you gone through all the steps at least a few times.

Interpolation between grids best avoided, hence, try to have the ice-sheet grid embedded in the ocean grid.

The ocean domain will most likely be an open one, don’t worry too much about global conservation of mass and energy, make sure it’s approximately correct at a local level, and that you can estimate associated errors.

Conservation of energy and volume is overrated! (Anonymous quote by an attendee to a Cambridge symposium on adjoints)

How will the future models be written...?

Northumbria University NEWCASTLE

frontiers in Earth Science

ORIGINAL RESEARCH
published: 11 April 2018
doi: 10.3389/feart.2018.00033

Relevance of Detail in Basal Topography for Basal Slipperiness Inversions: A Case Study on Pine Island Glacier, Antarctica

Teresa M. Kyrke-Smith[1], G. Hilmar Gudmundsson[1,2] and Patrick E. Farrell[3]*

[1] British Antarctic Survey, Cambridge, United Kingdom, [2] Geography and Environmental Sciences, University of Northumbria, Newcastle, United Kingdom, [3] Mathematical Institute, University of Oxford, Oxford, United Kingdom

2. THE MODEL

We consider the isothermal nonlinear Stokes equations:

$$\nabla \cdot \mathbf{u} = 0, \quad (1)$$

$$\nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{g} = \mathbf{0}, \quad (2)$$

where ρ is the ice density, $\mathbf{g} = (0, 0, -g)$ is the gravity vector, $\mathbf{u} = (u, v, w)$ is the ice velocity vector and $\boldsymbol{\sigma}$ the stress tensor. The stress tensor is given by

$$\boldsymbol{\sigma}(\mathbf{u}, p) = -p\mathrm{I} + \boldsymbol{\tau}(\mathbf{u}), \quad (3)$$

where p is the pressure, and $\boldsymbol{\tau}$ the deviatoric stress tensor. The deformation of the ice is described by the following constitutive relation:

$$\boldsymbol{\tau} = 2\eta\dot{\boldsymbol{\epsilon}}, \quad (4)$$

$$\eta = \frac{1}{2}A^{-1/n}\,\dot{\boldsymbol{\epsilon}}_{II}^{(1-n)/2n}, \quad (5)$$

$$\dot{\boldsymbol{\epsilon}}_{II} = \frac{1}{2}Tr(\dot{\boldsymbol{\epsilon}}^2), \quad (6)$$

$$\dot{\boldsymbol{\epsilon}} = \frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right), \quad (7)$$

AGU PUBLICATIONS

JGR

Journal of Geophysical Research: Earth Surface

RESEARCH ARTICLE
10.1002/2017JF004373

Key Points:
- We make new comparisons of ice sheet basal conditions estimated by two methods (model inversion versus seismic measurements)
- There is correlation between mean values of basal slipperiness and acoustic impedance when averaged over large enough length scales
- Seismic measurements of bed properties cannot be incorporated into models unless a physical theory is developed

Supporting Information:
- Supporting Information S1
- Figure S1
- Figure S2

Correspondence to:
T. M. Kyrke-Smith,

Can Seismic Observations of Bed Conditions on Ice Streams Help Constrain Parameters in Ice Flow Models?

Teresa M. Kyrke-Smith[1], G. Hilmar Gudmundsson[1], and Patrick E. Farrell[2]

[1] British Antarctic Survey, Cambridge, UK, [2] Mathematical Institute, University of Oxford, Oxford, UK

Abstract We investigate correlations between seismically derived estimates of basal acoustic impedance and basal slipperiness values obtained from a surface-to-bed inversion using a Stokes ice flow model. Using high-resolution measurements along several seismic profiles on Pine Island Glacier (PIG), we find no significant correlation at kilometer scale between acoustic impedance and either retrieved basal slipperiness or basal drag. However, there is a stronger correlation when comparing average values along the individual profiles. We hypothesize that the correlation appears at the length scales over which basal variations are important to large-scale ice sheet flow. Although the seismic technique is sensitive to the material properties of the bed, at present there is no clear way of incorporating high-resolution seismic measurements of bed properties on ice streams into ice flow models. We conclude that more theoretical work needs to be done before constraints on mechanical conditions at the ice-bed interface from acoustic impedance measurements can be of direct use to ice sheet models.

...in higher level languages using greater abstraction, e.g Fenics, Firebird, and other semi-automated systems

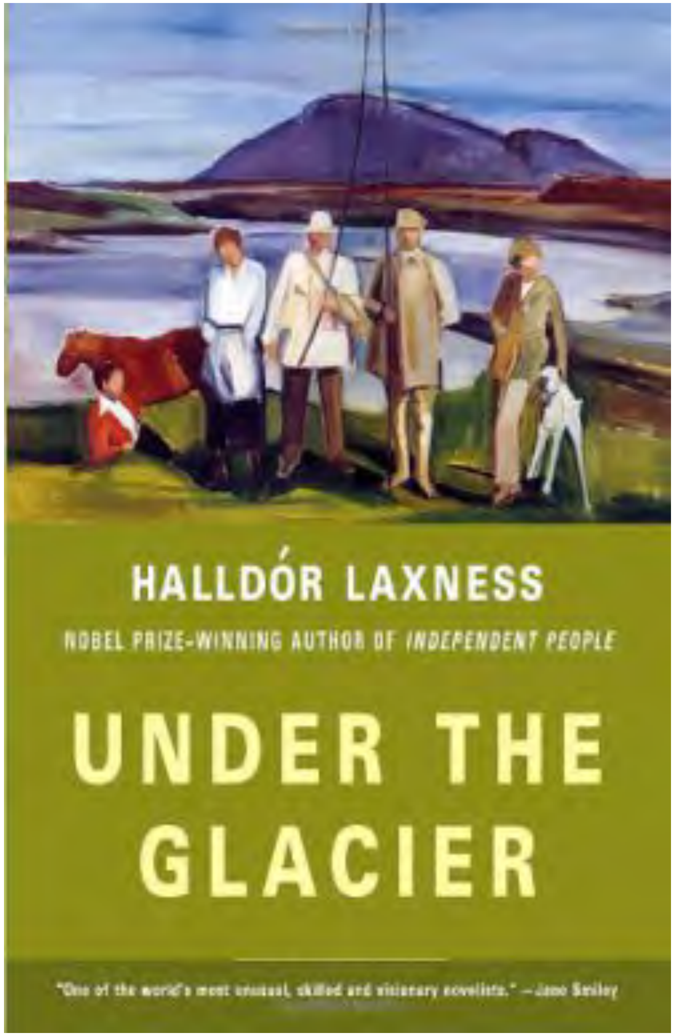
```
from dolfin import
#import sys; sys.exit(1)
mesh=Mesh("GmeshFileLargeAmp.xml")

V=VectorFunctionSpace(mesh,"CG",2,constrained_domain=periodicmap)
Q=FunctionSpace(mesh,"CG",1,constrained_domain=periodicmap)
P=FunctionSpace(mesh,"CG",2,constrained_domain=periodicmap)
W=MixedFunctionSpace([V, Q, P])

ds_bottom = ds[colours](3)
g = as_vector([0.01/1000.0, -9.81/1000.0])
n = FacetNormal(mesh)
I = Identity(2)
N = outer(n, n)
T = I - N
w= Function(W)
(u, p, lamda)=split(w)
w_test = TestFunction(W)
(Nu, Np, Nlamda)=split(w_test)
rho = Constant(910.0)          alpha = Constant(1.0e-20)
epsilon=sym(grad(u))
Tu = dot(T, u)
eps_min = Constant(1.0e-5)
eps_2 = 0.5 * tr(dot(epsilon, epsilon)) + eps_min**2.0
 A = Constant(A_val)
ng = Constant(ng_val)
C= Constant(C_val)              m = Constant(m_val)
eta = 0.5 * A**(-1.0/ng) * eps_2**((1.0-ng)/(2.0*ng))
 C_T = -C**(-1.0/m)*inner(Tu, Tu)**((1.0-m)/(2.0*m))

 F=(
     2*eta*inner(epsilon,grad(Nu))*dx
     -div(u)*Np*dx
     -div(Nu)*p*dx
     -rho*dot(Nu, g)*dx
     -inner(Nu, C_T*Tu)*ds_bottom
     +inner(lamda, dot(Nu, n))*ds_bottom
     +inner(Nlamda, dot(u, n))*ds_bottom
     +alpha*inner(lamda, Nlamda)*dx
    )

solve(F==0, w)
```

Northumbria University NEWCASTLE

Full Stokes solver, Glen‘s flow law, Weertman sliding law in python using the Fenics library

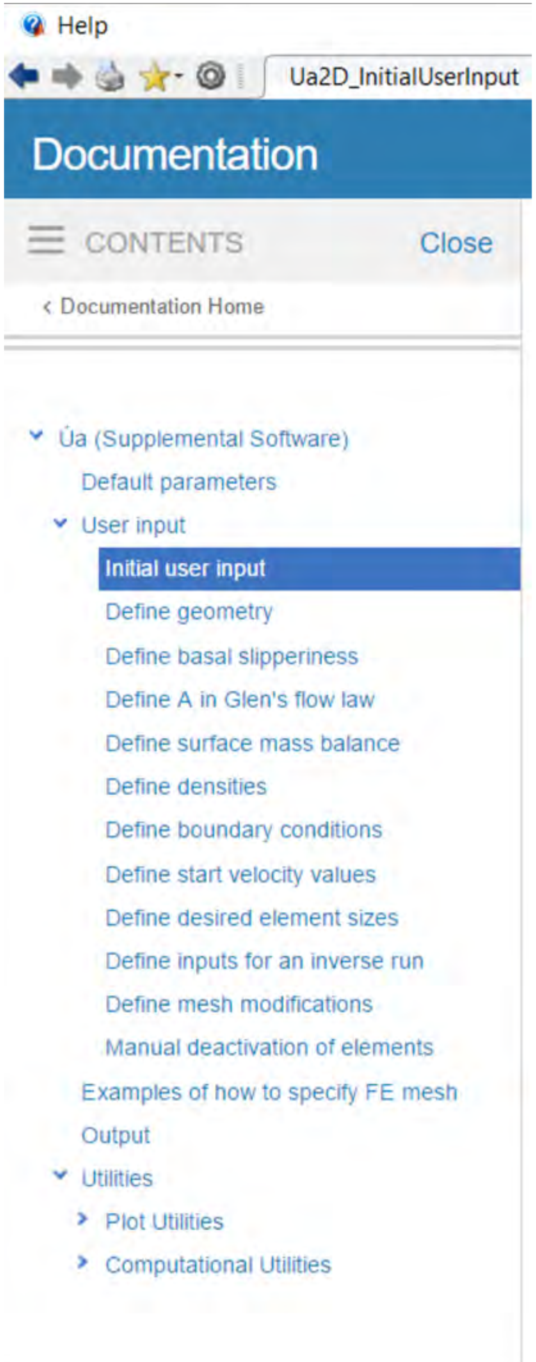
Focus on the math and the physics, the numerical implementation is done automatically.

Why the name Úa?



Nobel laureate Halldór Laxness's Under the Glacier" is a" one-of-a-kind masterpiece, a wryly provocative novel at once earthy and otherworldly.

Piling improbability on top of improbability, Under the Glacier" overflows with comedy both wild and deadpan as it conjures a phantasmagoria as beguiling as it is profound.



Manuals are so 1970s!

Users don‘t even read the README files!

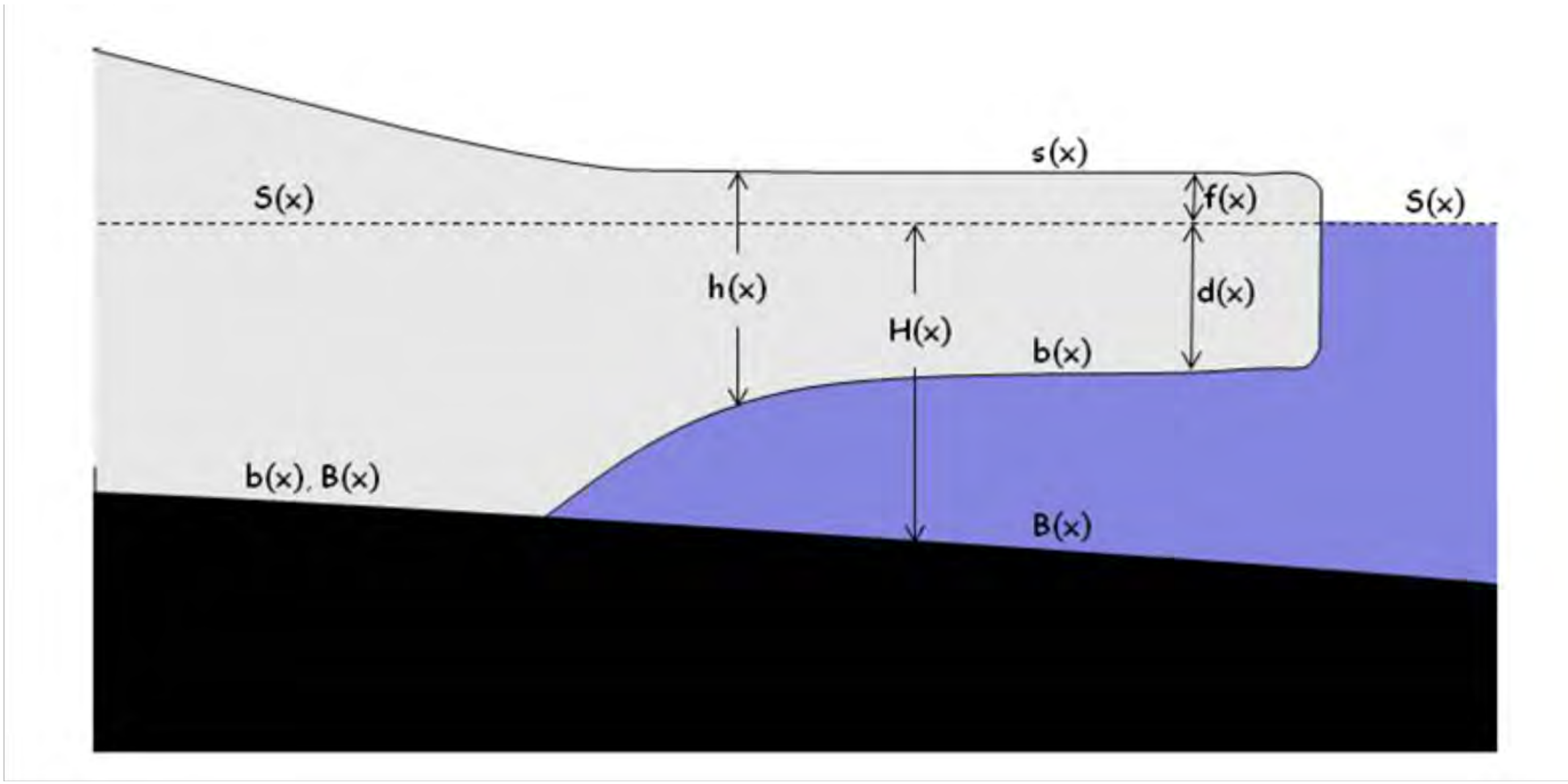
The Úa help system is fully integrated into the matlab help.

Úa will appear as a seperate Toolbox.

Open the matlab documentation (doc [ret]) and locate the Úa Toolbox.

Úa variables

The geometrical variables are:



Inverse modeling

We have a forward model

$$F(u(p), p) = 0$$

where p are model parameters and u the state variable. We consider the problem of minimimizing an objective function J with respect to p. Typically the objective function J can be thought of as a sum of two terms

$$J(u, p) = I(u) + R(p)$$

where I is a misfit term and R a regularisation term.

Misfit functions in *Úa*

$$I = I_u + I_v$$

$$\begin{aligned} I = & \frac{1}{2\mathcal{A}} \int ((u - u_{\text{meas}})/u_{\text{error}})^2 \, dA \\ & + \frac{1}{2\mathcal{A}} \int ((v - v_{\text{meas}})/v_{\text{error}})^2 \, dA \end{aligned}$$

Regularisation in *Úa*

$$R = (\boldsymbol{p} - \hat{\boldsymbol{p}})^T \boldsymbol{K}_{pp}^{-1} (\boldsymbol{p} - \hat{\boldsymbol{p}})$$ Bayesian motivated

$$R = \frac{1}{2\mathcal{A}} \int \left(\gamma_s^2 \left(\nabla (p - \hat{p}) \right)^2 + \gamma_a^2 \left(p - \hat{p} \right)^2 \right) \, dA$$ Tikhonov regularisation