ISSM Workshop 2012

JPL

Ice Sheet System model

Inverse Methods

Chris BORSTAD[1], Bao DUONG[5,1], Feras HABBAL[2,1], Daria HALKIDES[1,3], Michiel HELSEN[2], Eric LAROUR[1], **Mathieu MORLIGHEM**[2], Lan NGUYEN[5,1], Gilberto PÉREZ[4,1], Eric RIGNOT[2,1], John SCHIERMEIER[1], Nicole SCHLEGEL[1], Hélène SEROUSSI[1]

[1] Jet Propulsion Laboratory - California Institute of Technology

[2] University of California, Irvine

[3] Joint Institute for Regional Earth System Science & Engineering, UCLA

[4] University of Southern California

[5] Cal Poly Pomona

ISSM

Inverse Methods

Introduction
Model equations
Inverse problem
Algorithm

Hands on 1 (ice rigidity)
Initial state
Inversion

Hands on 2 (friction)

Outline

1. Introduction
   - Model equations
   - Inverse problem
   - Algorithm
2. Hands on 1 (ice rigidity)
   - Initial state
   - Inversion
3. Hands on 2 (friction)

Inverse Methods

Introduction
Model equations
Inverse problem
Algorithm

Hands on 1 (ice rigidity)
Initial state
Inversion

Hands on 2 (friction)

Ice flow model equations

Field equations

1. Stokes flow:

$$\nabla \cdot \boldsymbol{\sigma} + \rho \boldsymbol{g} = \boldsymbol{0} \tag{1}$$

2. Incompressibility:

$$\nabla \cdot \boldsymbol{v} = 0 \tag{2}$$

3. Constitutive law:

$$\boldsymbol{\sigma}' = \boldsymbol{\sigma} + p\,\mathbf{I} = 2\mu\dot{\varepsilon} \qquad \mu = \frac{B}{2\,\dot{\varepsilon}_e^{1-1/n}} \tag{3}$$

Boundary conditions

Ice/Air interface: Free surface	$\boldsymbol{\sigma} \cdot \boldsymbol{n} = P_{atm}\,\boldsymbol{n} \simeq \boldsymbol{0}$	on Γ_s
Ice/Ocean interface: water pressure	$\boldsymbol{\sigma} \cdot \boldsymbol{n} = P_w\,\boldsymbol{n}$	on Γ_w
Ice/Bedrock interface (1): lateral friction	$\left(\boldsymbol{\sigma} \cdot \boldsymbol{n} + \alpha^2 \boldsymbol{v}\right)_{\parallel} = \boldsymbol{0}$	on Γ_b
Ice/Bedrock interface (2): impenetrability	$\boldsymbol{v} \cdot \boldsymbol{n} = -\dot{M}_b\,n_z$	on Γ_b

+ Dirichlet condition

Inverse Methods
Introduction
Model equations
Inverse problem
Algorithm
Hands on 1 (ice rigidity)
Initial state
Inversion
Hands on 2 (friction)

Ice flow model equations

Field equations

1. Stokes flow:

$$\nabla \cdot \boldsymbol{\sigma} + \rho \boldsymbol{g} = \boldsymbol{0} \quad (1)$$

2. Incompressibility:

$$\nabla \cdot \boldsymbol{v} = 0 \quad (2)$$

3. Constitutive law:

$$\boldsymbol{\sigma}' = \boldsymbol{\sigma} + p\,\mathbf{I} = 2\mu\dot{\varepsilon} \qquad \mu = \frac{B}{2\,\dot{\varepsilon}_e^{1-1/n}} \quad (3)$$

Boundary conditions

Ice/Air interface: Free surface $\boldsymbol{\sigma} \cdot \boldsymbol{n} = P_{atm}\, \boldsymbol{n} \simeq \boldsymbol{0}$ on Γ_s

Ice/Ocean interface: water pressure $\boldsymbol{\sigma} \cdot \boldsymbol{n} = P_w\, \boldsymbol{n}$ on Γ_w

Ice/Bedrock interface (1): lateral friction $\left(\boldsymbol{\sigma} \cdot \boldsymbol{n} + \alpha^2 \boldsymbol{v}\right)_{\parallel} = \boldsymbol{0}$ on Γ_b

Ice/Bedrock interface (2): impenetrability $\boldsymbol{v} \cdot \boldsymbol{n} = -\dot{M}_b\, n_z$ on Γ_b

+ Dirichlet condition

Inverse Methods

Introduction
Model equations
Inverse problem
Algorithm

Hands on 1 (ice rigidity)
Initial state
Inversion

Hands on 2 (friction)

Ice flow model equations

Field equations

1. Stokes flow:

$$\nabla \cdot \boldsymbol{\sigma} + \rho \boldsymbol{g} = \mathbf{0} \tag{1}$$

2. Incompressibility:

$$\nabla \cdot \boldsymbol{v} = 0 \tag{2}$$

3. Constitutive law:

$$\boldsymbol{\sigma}' = \boldsymbol{\sigma} + p\,\mathbf{I} = 2\mu\dot{\varepsilon} \qquad \mu = \frac{B}{2\,\dot{\varepsilon}_e^{1-1/n}} \tag{3}$$

Boundary conditions

Ice/Air interface: Free surface	$\boldsymbol{\sigma} \cdot \boldsymbol{n} = P_{atm}\ \boldsymbol{n} \simeq \mathbf{0}$	on Γ_s
Ice/Ocean interface: water pressure	$\boldsymbol{\sigma} \cdot \boldsymbol{n} = P_w\ \boldsymbol{n}$	on Γ_w
Ice/Bedrock interface (1): lateral friction	$\left(\boldsymbol{\sigma} \cdot \boldsymbol{n} + \alpha^2 \boldsymbol{v}\right)_{\parallel} = \mathbf{0}$	on Γ_b
Ice/Bedrock interface (2): impenetrability	$\boldsymbol{v} \cdot \boldsymbol{n} = -\dot{M}_b\, n_z$	on Γ_b

+ Dirichlet condition

Inverse Methods

Introduction
Model equations
Inverse problem
Algorithm

Hands on 1 (ice rigidity)
Initial state
Inversion

Hands on 2 (friction)

Inverse problems

- Basal friction and ice hardness are difficult to measure
- Use extra datasets to infer unknows

$\rightarrow$ ex: surface velocities derived from InSAR

PDE-constrained optimization

Minimize cost function

$$\mathcal{J}\left(\boldsymbol{v},\alpha\right)=\frac{1}{2}\int_{\Gamma_s}\left(v_x-v_x^{obs}\right)^2+\left(v_y-v_y^{obs}\right)^2 dS+\mathcal{R}\left(\alpha\right) \tag{4}$$

Subject to:

$$\begin{aligned}
\nabla\cdot\mu\left(\nabla\boldsymbol{v}+\nabla\boldsymbol{v}^T\right)-\nabla p+\rho\boldsymbol{g} &= \boldsymbol{0} && \text{in } \Omega\\
\nabla\cdot\boldsymbol{v} &= 0 && \text{in } \Omega\\
\boldsymbol{\sigma}\cdot\boldsymbol{n} &= \boldsymbol{f} && \text{on } \Gamma_s\cup\Gamma_w\\
\left(\boldsymbol{\sigma}\cdot\boldsymbol{n}+\alpha^2\boldsymbol{v}\right)_{\parallel} &= \boldsymbol{0} && \text{on } \Gamma_b\\
\boldsymbol{v}\cdot\boldsymbol{n} &= -\dot{M}_b\, n_z && \text{on } \Gamma_b
\end{aligned} \tag{5}$$

Inverse Methods

Introduction
Model equations
Inverse problem
Algorithm

Hands on 1 (ice rigidity)
Initial state
Inversion

Hands on 2 (friction)

Inverse problems

- Basal friction and ice hardness are difficult to measure
- Use extra datasets to infer unknows

$\rightarrow$ ex: surface velocities derived from InSAR

PDE-constrained optimization

Minimize cost function

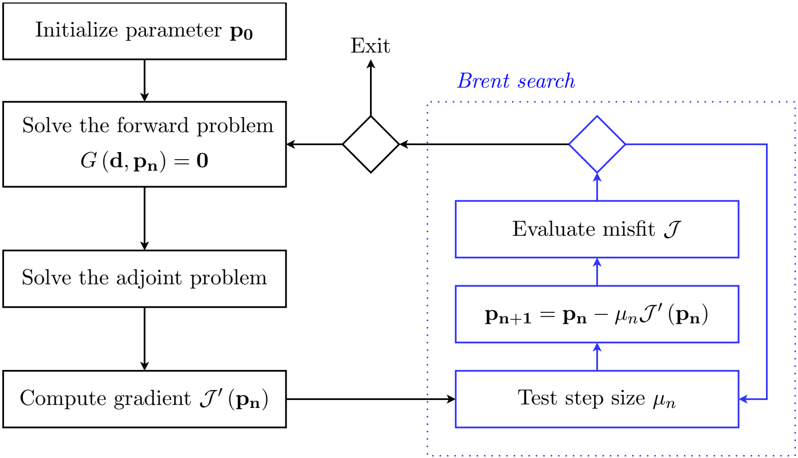
$$\mathcal{J}(\boldsymbol{v},\alpha) = \frac{1}{2}\int_{\Gamma_s} \left(v_x - v_x^{\mathrm{obs}}\right)^2 + \left(v_y - v_y^{\mathrm{obs}}\right)^2 dS + \mathcal{R}(\alpha) \tag{4}$$

Subject to:

$$\begin{aligned}
\nabla \cdot \mu\left(\nabla \boldsymbol{v} + \nabla \boldsymbol{v}^T\right) - \nabla p + \rho \boldsymbol{g} &= \boldsymbol{0} && \text{in } \Omega \\
\nabla \cdot \boldsymbol{v} &= 0 && \text{in } \Omega \\
\boldsymbol{\sigma} \cdot \boldsymbol{n} &= \boldsymbol{f} && \text{on } \Gamma_s \cup \Gamma_w \\
\left(\boldsymbol{\sigma} \cdot \boldsymbol{n} + \alpha^2 \boldsymbol{v}\right)_{\parallel} &= \boldsymbol{0} && \text{on } \Gamma_b \\
\boldsymbol{v} \cdot \boldsymbol{n} &= -\dot{M}_b\, n_z && \text{on } \Gamma_b
\end{aligned} \tag{5}$$

Inverse Methods
Introduction
Model equations
Inverse problem
Algorithm
Hands on 1 (ice rigidity)
Initial state
Inversion
Hands on 2 (friction)

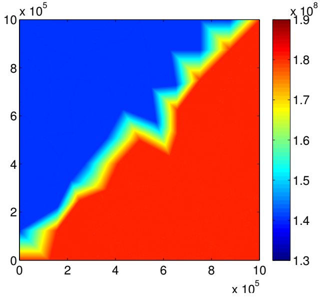
Algorithm of resolution

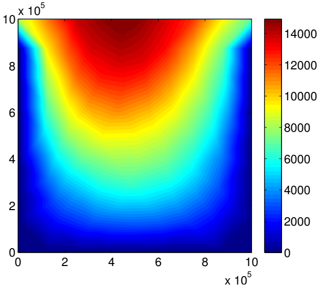


Hands on 1 (ice rigidity)

```
%Generate observation
md = model;
md = triangle(md,'DomainOutline.exp',100000);
md = setmask(md,'all','');
md = parameterize(md,'Square.par');
md = setflowequation(md,'macayeal','all');
md.cluster = generic('np',2);
md = solve(md,DiagnosticSolutionEnum);
```

```
md=SetIceShelfBC(md,'Front.exp');
```



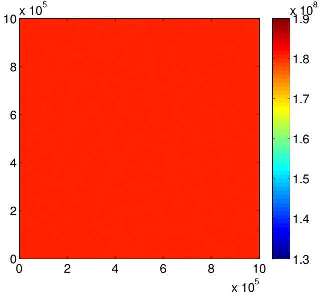


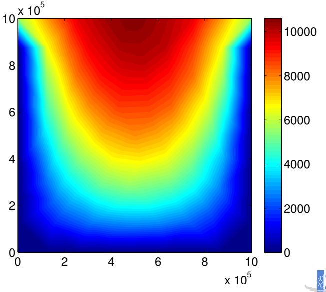
Start from constant hardness

```
%Modify rheology, now constant
loadmodel('model1.mat');
md.materials.rheology_B(:) = 1.8*10^8;

%results of previous run are taken as observations
md.inversion.vx_obs  = md.results.DiagnosticSolution.Vx;
md.inversion.vy_obs  = md.results.DiagnosticSolution.Vy;
md.inversion.vel_obs = md.results.DiagnosticSolution.Vel;

md = solve(md,DiagnosticSolutionEnum);
```





Inverse Methods

Introduction
Model equations
Inverse problem
Algorithm

Hands on 1 (ice rigidity)
Initial state
Inversion

Hands on 2 (friction)

Inverse method

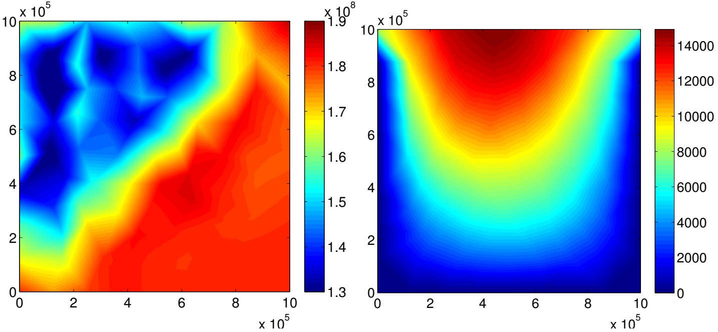
```
%invert for ice rigidity
loadmodel('model2.mat');

%Set up inversion parameters
nsteps = 40;
md.inversion.iscontrol = 1;
md.inversion.control_parameters = {'MaterialsRheologyBbar'};
md.inversion.nsteps  = nsteps;
md.inversion.cost_functions = 101*ones(nsteps,1);
md.inversion.cost_functions_coefficients = ones(md.mesh.numberofvertices,1);
md.inversion.maxiter_per_step  = 10*ones(nsteps,1);
md.inversion.step_threshold    = .8*ones(nsteps,1);
md.inversion.gradient_scaling  = 10^7*ones(nsteps,1);
md.inversion.min_parameters    = ...
     paterson(273)*ones(md.mesh.numberofvertices,1);
md.inversion.max_parameters    = ...
     paterson(200)*ones(md.mesh.numberofvertices,1);

%Go solve!
md=solve(md,DiagnosticSolutionEnum);
```

Inverse Methods
Introduction
Model equations
Inverse problem
Algorithm
Hands on 1 (ice rigidity)
Initial state
Inversion
Hands on 2 (friction)

Inverse method



Inverse Methods

Introduction
Model equations
Inverse problem
Algorithm

Hands on 1 (ice rigidity)
Initial state
Inversion

Hands on 2 (friction)

Hands on 2 (friction)

- Same example but now for a grounded glacier:
- Changes step 1:
  1. increase bed and surface elevation by 100 m
  2. mask is now all grounded
  3. $B = 1.8 \times 10^8$ uniform
  4. friction coefficient: 50, and 10 for $600000 < x < 400000$
- Changes step 2:
  1. friction coefficient uniform (50)
- Changes step 3:
  1. We now invert for `'FrictionCoefficient'`
  2. Do we keep the same cost function ?
  3. gradient now scaled to 10
  4. we want the parameter to be between 1 and 100


Thanks!