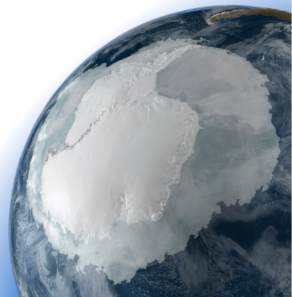


# Ice Sheet System Model

## Application of Operation Ice Bridge Data

Johannes BONDZIO<sup>1</sup>,

<sup>1</sup>University of California, Irvine, CA



# Overview

- 1 Introduction
- 2 Mesh
- 3 Parameterization
- 4 Control Method Solution
- 5 Transient Solution
- 6 Plot Results

## Goals

Use your new ISSM skills to adjust a coarse Greenland model by adding IceBridge data.

We refine the mesh in the Jakobshavn basin and add higher resolution bedrock and surface elevation data from IceBridge in this area.

### Steps:

- Refine Greenland mesh using given Jakobshavn outline
- Parameterize, include the high-resolution IceBridge bedrock and surface data
- Plot ice base and surface data
- Stress Balance: run 2 inverse method runs to solve for control drag (20 steps recommended)
- Transient: run 20 year runs, with coarse and refined bedrock and surface elevation data
- Plot transient results

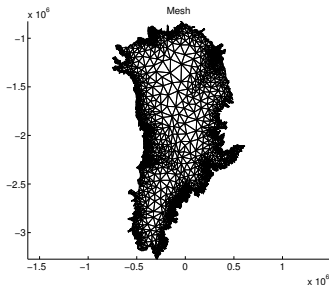
## First Run Step: Mesh

We modify the experiment from the Greenland SeaRISE talk, and improve from there.

First, run the first step in `runme.m` in directory `IceBridge` to mesh the Greenland domain as done in the previous talk.

Step 1 is interrupted after making the default mesh. Plot the result.

It should look like this:



## Mesh, 2/5

### Adapt

We want to refine the mesh in the Jakobshavn area. An outline of this area **Jak\_outline.exp** can be found in this directory.

Try using **exptool** to view this outline:

```
1  >> exptool('Jak_outline.exp');
```

## Mesh, 3/5

### Adapt

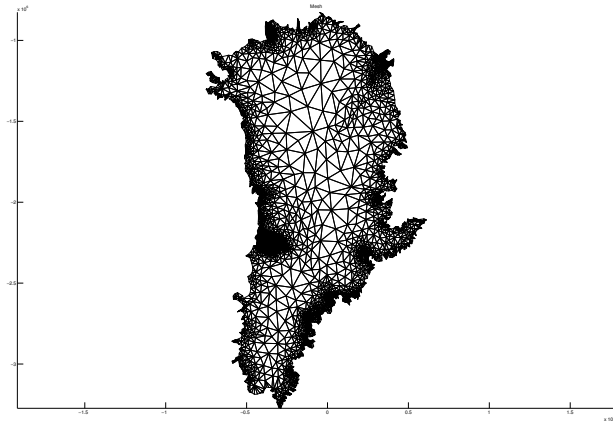
Next, we modify the `bamg` command by imposing a 3 km resolution within the Jakobshavn area: using `hmaxVertices`.

Note that you need to deactivate the previous `bamg` command.

```
26
27  %Refine mesh in the region of Jakobshavn (resolution = 3000 m)
28  hmaxVertices=NaN*ones(md.mesh.numberofvertices,1);
29  in=ContourToMesh(md.mesh.elements,md.mesh.x,md.mesh.y,...
30    'Jak_outline.exp','node',1);
31  hmaxVertices(find(in))=3000;
32  md=bamg(md,'hmax',400000,'hmin',5000,'gradation',1.7,'field',vel,...
```

Check your results using `plotmodel`.

# Mesh, 4/5 Solution



Try **zoom** to make a close-up of the Jakobshavn domain.

# Mesh, 5/5

## Full Solution

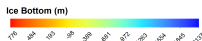
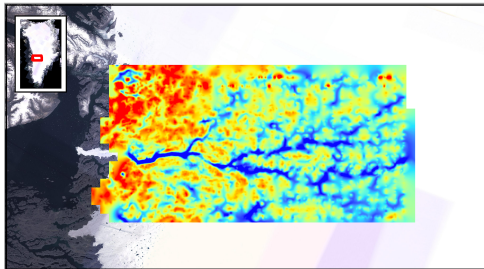
```
22
23 %Mesh greenland without refinement in Jak basin
24 md=bamg(md,'hmax',400000,'hmin',5000,'gradation',1.7,'field',vel,'err',8);
25 return;
26
27 %Refine mesh in the region of Jakobshavn (resolution = 3000 m)
28 hmaxVertices=NaN*ones(md.mesh.numberofvertices,1);
29 in=ContourToMesh(md.mesh.elements,md.mesh.x,md.mesh.y,...
30     'Jak_outline.exp','node',1);
31 hmaxVertices(find(in))=3000;
32 md=bamg(md,'hmax',400000,'hmin',5000,'gradation',1.7,'field',vel,...
33     'err',8,'hmaxVertices',hmaxVertices);
34
35 %convert x,y coordinates (Polar stereo) to lat/lon
36 [md.mesh.lat,md.mesh.long]=xy2ll(md.mesh.x,md.mesh.y,+1,39,71);
```

## Parameterization

We want to include high-resolution bedrock and surface elevation data acquired in the IceBridge mission. Accessible on:

<https://data.cresis.ku.edu/data/grids/>

[Jakobshavn\\_2008\\_2011\\_Composite\\_XYZGrid.txt](#)



## Parameterization, 2/6

Bedrock data is read, transformed into a usable grid, and interpolated to the mesh in the parameter file **Greenland.par**:

```
30 %Reading IceBridge data for Jakobshavn
31 disp('      reading IceBridge Jakobshavn bedrock');
32 fid = fopen(' ../Data/Jakobshavn_2008_2011_Composite_XYZGrid.txt');
33 titles = fgets(fid);
34 data = fscanf(fid, '%g,%g,%g,%g,%g',[5 266400]);
35 fclose(fid);
36
37 [xi,yi]= ll2xy(md.mesh.lat,md.mesh.long,+1,45,70);
38 bed = flipud(reshape(data(:,5),[360 740])); bed(find(bed==-9999))=NaN;
39 bedy = flipud(reshape(data(:,1),[360 740]));
40 bedx = flipud(reshape(data(:,2),[360 740]));
41
42 %Insert Icebridge bed and recalculate thickness
43 bed_jks=InterpFromGridToMesh(bedx(1,:),bedy(:,1),bed,xi,yi,NaN);
44 in=ContourToMesh(md.mesh.elements,md.mesh.x,md.mesh.y,...
45     'Jak_grounded.exp','node',1);
46 bed_jks(~in)=NaN;
47 pos=find(~isnan(bed_jks));
48 md.geometry.base(pos)=bed_jks(pos);
```

Modify **Greenland.par** such that the surface elevation data is also included for the Jakobshavn area.

## Parameterization, 3/6

### Solution

```
30 %Reading IceBridge data for Jakobshavn
31 disp('      reading IceBridge Jakobshavn bedrock');
32 fid = fopen('./Data/Jakobshavn_2008_2011_Composite_XYZGrid.txt');
33 titles = fgets(fid);
34 data = fscanf(fid, '%g,%g,%g,%g,%g',[5 266400]);
35 fclose(fid);
36
37 [xi,yi]= ll2xy(md.mesh.lat,md.mesh.long,+1,45,70);
38 bed = flipud(reshape(data(:,5),[360 740])); bed(find(bed==-9999))=NaN;
39 surf = flipud(reshape(data(:,4),[360 740])); surf(find(surf==-9999))=NaN;
40 bedy = flipud(reshape(data(:,1),[360 740]));
41 bedx = flipud(reshape(data(:,2),[360 740]));
42
43 %Insert Icebridge bed and recalculate thickness
44 bed_jks=InterpFromGridToMesh(bedx(1,:)',bedy(:,1),bed,xi,yi,NaN);
45 surf_jks=InterpFromGridToMesh(bedx(1,:)',bedy(:,1),surf,xi,yi,NaN);
46 in=ContourToMesh(md.mesh.elements,md.mesh.x,md.mesh.y,...
47     'Jak_grounded.exp','node',1);
48 bed_jks(~in)=NaN;
49 surf_jks(~in)=NaN;
50 pos=find(~isnan(bed_jks));
51 md.geometry.base(pos)=bed_jks(pos);
52 md.geometry.surface(pos)=surf_jks(pos);
```

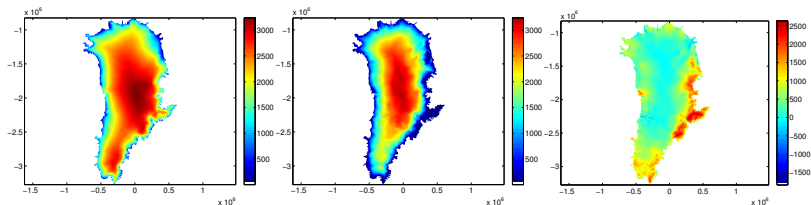
Plot surface elevation, thickness, and base.

## Parameterization, 4/6

### Solution

```
1 >> plotmodel(md,'data',md.geometry.surface);  
2 >> plotmodel(md,'data',md.geometry.thickness);  
3 >> plotmodel(md,'data',md.geometry.base);
```

They should look like:



## Parameterization, 5/6

### Solution

To plot the difference in ice base topography between the SeaRISE and IceBridge datasets:

Modify the parameterization step in your runme and save the model under a different name.

A difference in fields can be plotted using:

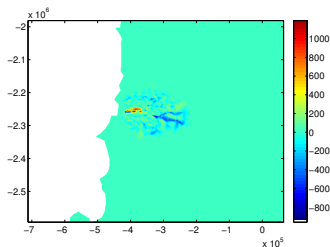
```
1 >> md2=loadmodel('Models/Greenland.Parameterization2');  
2 >> md=loadmodel('Models/Greenland.Parameterization');  
3 >> plotmodel(md, 'data', md2.geometry.base-md.geometry.base);
```

## Parameterization, 6/6

### Solution

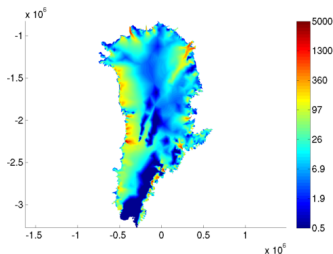
A difference in fields can be plotted using:

```
1 >> md2=loadmodel('Models/Greenland.Parameterization2');  
2 >> md=loadmodel('Models/Greenland.Parameterization');  
3 >> plotmodel(md,'data',md2.geometry.base-md.geometry.base);
```



## Stress Balance

Use control methods to inversely solve for Greenland friction coefficient  
Comparable to the **Greenland** experiment



The observed velocity map contains some gaps; exclude these from the inversion by creating a new exp file that outlines the gaps in the velocity data, using **exptool**:

```
1 >> exptool('data_gaps.exp');
```

## Stress Balance, 2/3

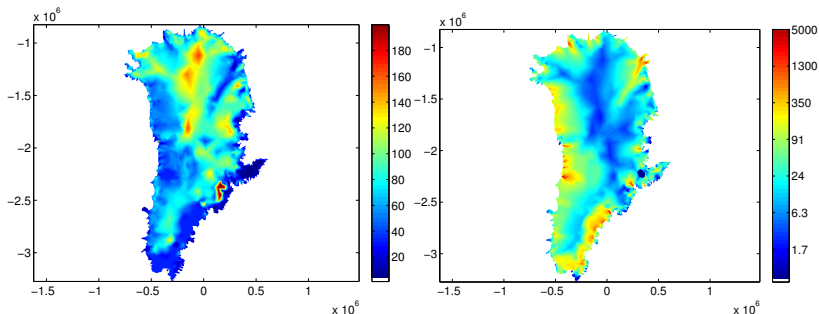
```
74 md.inversion.max_parameters=200*ones(md.mesh.numberofvertices,1);
75 in=ContourToMesh(md.mesh.elements,md.mesh.x,md.mesh.y,...
76   'data_gaps.exp','node',1);
77 md.inversion.cost_functions_coefficients(find(in),1)=0.0;
78 md.inversion.cost_functions_coefficients(find(in),2)=0.0;
```

Next, run the model and plot the resulting friction coefficient and velocity pattern

```
1 >> plotmodel(md,'data',md.results.StressbalanceSolution.FrictionCoefficient);
2 >> plotmodel(md,'data',md.results.StressbalanceSolution.Vel,...
3   'log',10,'caxis',[0.5 5000]);
```

## Stress Balance, 3/3

```
1 >> plotmodel(md, 'data', md.results.StressbalanceSolution.FrictionCoefficient);  
2 >> plotmodel(md, 'data', md.results.StressbalanceSolution.Vel, ...  
3     'log', 10, 'caxis', [0.5 5000]);
```



# Transient

## Forcing

Do a transient run..

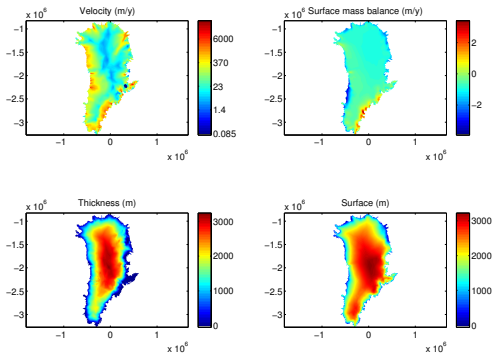
## Transient Results

### Plot Plan

Your results are located in **md.results.TransientSolution**. Plot your results.

First, plot the initial plan view of velocity, surface mass balance, thickness, and surface.

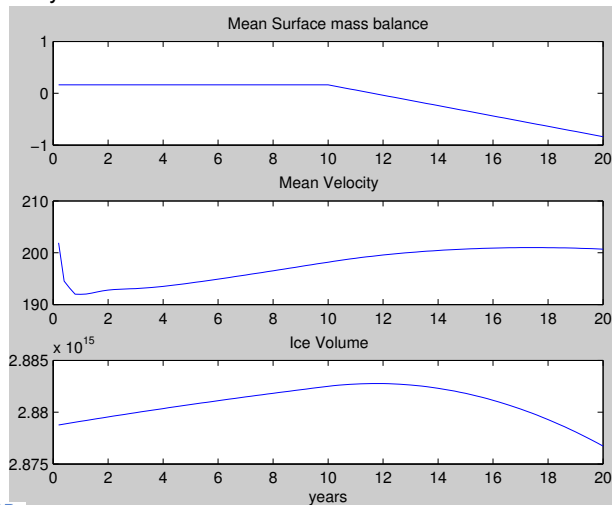
They should look like:



## Transient Results

### Plot time series

They should look like:



## Transient Results

Some suggestions what to explore further:

- How would you make a plot of time series of results from the SeaRISE and IceBridge experiments?
- How would you make a plot of the difference between final and initial ice thickness?
- ...

We can help you to implement your own ideas in the code.

Thanks!

