Introduction to the Bering-Bagley Glacier System Model using Elmer/Ice

Thomas Trantow

trantow@colorado.edu

May 2022

Contents

1 Notebook Purpose

The purpose of this notebook is to give a non-exhaustive introduction to the Bering-Bagley Glacier System (BBGS) model in Elmer/Ice as described in Trantow and Herzfeld [2018], Trantow et al. [2020], Trantow and Herzfeld [subm] and Trantow [2020]. A more comprehensive introduction to Elmer and Elmer/Ice can be found at `http://elmerice.elmerfem.org/`. This notebook serves to get the user prepared for running the BBGS model using the datasets and SIF files found on this GitHub Page (`https://github.com/trantow/bbgs_elmer`).

All the files needed for the full surge simulations described in Trantow and Herzfeld [subm] are available in the elmer_code_JGR folder while an example SIF file for Trantow et al. [2020] is found in the main folder (bbgs_elmer/crev_BBGS_C2_swath_IAMG_20181126.sif). Additional SIF files and input DEM files for this project can be attained via email: trantow@colorado.edu. Files common to both experiments are also found in the main folder.

2 Modeling with Elmer/Ice

Useful Links for Elmer and Elmer/Ice

- Elmer at CSC (documentation, how to install, etc.)
  `http://www.elmerfem.org/` `https://www.csc.fi/web/elmer`
- Elmer (and thereby Elmer/Ice) source code
  `https://github.com/ElmerCSC/elmerfem`
- Elmer Forum
  `http://elmerfem.org/forum/`
- Elmer/Ice webpage
  `http://elmerice.elmerfem.org/`
- Elmer/Ice wiki
  `http://elmerfem.org/elmerice/wiki/doku.php?id=start`

Starting Out

This is not a tutorial on how to use Elmer and Elmer/Ice in general, instead it is specific for the BBGS model. If you are unfamiliar with Elmer, I'd recommend trying out the Elmer/Ice courses and tutorials at:
`http://elmerice.elmerfem.org/courses-tutorials`

The most helpful tutorial is the one on Tête Rousse Glacier (or the Arctic Valley Glacier = Midtre Lovenbreen). I based the BBGS on the code from this tutorial and their basic structures are mostly the same. It is a great baseline for modeling a real glacier (as opposed to some idealistic glacier).

Another very helpful tool is the ElmerSolver manual:

http://www.elmerfem.org/blog/documentation/

The manual contains all the syntax and commands used in Elmer. It is the single best reference for interpreting Elmer specific code.

The rest of this document assumes you have a basic understanding on how to run Elmer. Just know that to run a Solver Input File (sif) you use the command

ElmerSolver input_file_name.sif.

I would recommend the program tmux if you are running the model remotely.
https://danielmiessler.com/study/tmux/#invocation.

Docker

Note that Elmer is notoriously difficult to install properly on a computer. However, one can use the program docker to run Elmer and Elmer/ice on most computers. See:

https://hub.docker.com/r/nwrichmond/elmerice/
and also https://www.docker.com/resources/what-container.

Input: What is needed to start an Elmer simulation

The input file for Elmer is the Solver Input File (SIF file) which has the .sif suffix. See:
http://elmerice.elmerfem.org/wiki/lib/exe/fetch.php?media=courses:2021:2021_11_elmersif.pdf
Each model run I made has an associated SIF file, but only the few referenced in Trantow et al. [2020] and Trantow and Herzfeld [subm] are available on the GitHub repository. See also the section in the Elmer User Manual. There are also Solvers and User Functions specific for ice flow calculations, most of which are downloaded via the standard Elmer/Ice download. I have made a specific one for Bering Glacier called USF_Bering.f90 (in both project folders on this GitHub page) that handles the initial glacier geometry. To compile the f90 code of the USF use the following command:

```
elmerf90 USF_Bering.f90 -o USF_Bering
```

To create the initial glacier geometry (or finite element mesh) you'll need need a contour outline if the glacier, a surface DEM and a bed DEM to create a glacier mesh (see next section). Note that the mesh used in each run is specified in the SIF file under the header section near the top of the file. E.g.:

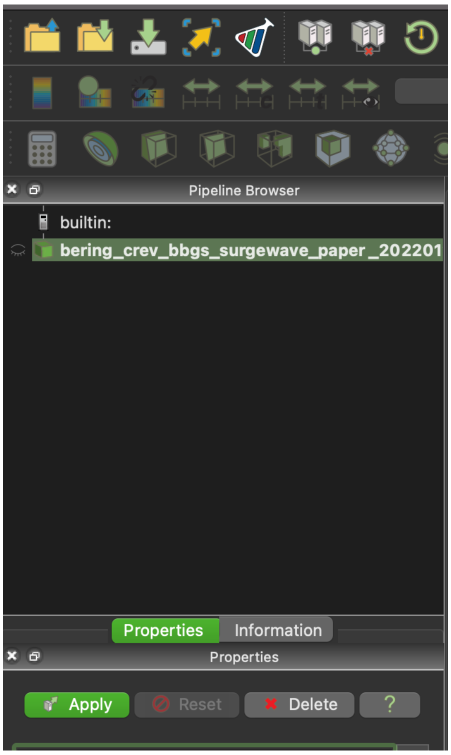
```
Header
```

```
  Mesh DB "." "BBGS_400_cv12"
End
```

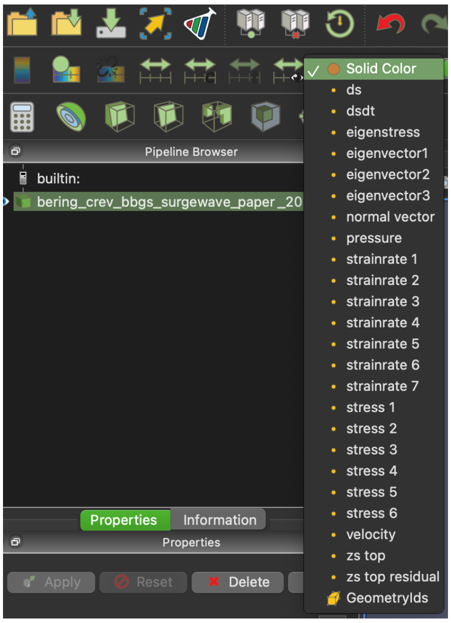
Post processing and Paraview

There are two output file types. The first is the .result file that Elmer uses if you want to start a model run from existing run states (see restart files in the Solver Manual). The second is .vtu file which are output made at each time step readable by the post processing software Paraview (free open source software: https://www.paraview.org/download/). I recommend using Paraview to view all you modeling results, to make quick plots and the save the model data a csv files.

You can load in your model results in Paraview by going to File/Open and selecting you series of output vtu files. Paraview will automatically group the vtu files together when looking for data to load in, so you don't need to load in each individual vtu file. After it is loaded in you should see it highlighted under the Pipeline Browser. Next, just click Apply.



The glacier outline should then popup. To view a specific output (like velocity), just select it in the dropdown menu that by default says "Solid Color":



at the top where it says

To save a screen shot of your view in Paraview, go to File/Screenshot.

To save a csv file of the data, go to File/Save Data.

Input DEMs

Input DEMs (surface and bedrock) are needed for creating a mesh for a 3D FEM model and are on the GitHub repository. Both are at 200m-by-200m resolution initially, i.e. before Elmer interpolates them to the FEM grid. The bedrock DEM is called bed_JPL_v7_fourptsall.dat and the input surface DEMs for the early 2011 phase is s0_swath_raddev_v2_4model.dat and for the surge-wave simulation is s11_VHswath_wfront_dem4model.dat.

One generates these DEMs via kriging (see Trantow and Herzfeld [2016]). I krigged CryoSat-2 Baseline-C data for the surface DEMs [Bouffard, 2015] (for swath data, see Helm et al. [2014]) and the JPL WISE data [Rignot et al., 2013] for the bedrock DEMs.

The DEMs for use in generating an Elmer mesh need to be in a specific format. I wrote code that changes the standard XYZ 3 column format of a DEM into the model format (dem4model.m in this GitHub repo). Note that you need to use UTM coordinates in modeling!! You can also use Polar-stereo, but some format that uses meters, i.e. not lat/lon. Elmer wants the DEM data to be ordered in a particular way which is what the dem4model.m code does. The function takes as input the DEM data, a string specifying the glacier and the resolution of the DEM. It will output the DEM in proper Elmer format and appends '_4model.dat' to the filename.

Now you can specify this DEM with the 4model suffix in the USF FORTRAN file (e.g. USF_Bering.f90). Just search for the '.dat' in the USF file and replace the old DEMs with your DEM. Note that you also need to specify the resolution and bounds that are used in the DEM. The bounds are actually defined in the dem4model code:

```
switch glacier
    % Bering - Bagley (with glims)
    case {'BBGS', 'Bering','bering'}
        x = (336000:res:508000);
        y = (6660000:res:6724000);
    % Negribreen
    case {'negri','Negri','Negribreen'}
        x = (540000:res:600000);
        y = (8710000:res:8760000);
end
```

Creating a Mesh:

See the meshing tutorials online that are usually contained within each specific example. The way in which the FEM mesh is created always changes for Elmer, but it usually involves using the program gmsh (`https://gmsh.info/`), which you may need to download and install.

I put an example file for starting the construction of a mesh called Makegeo_20210810.m in this repo. All it requires is a companion contour outline of the glacier which is also given on this repo (contour_BBGS_v12_UTM.dat).

1. In your makegeo matlab file (you can use Makegeo_20210810.m as a template), change the mesh resolution to your desired resolution by setting the "lc_out" variable.
2. Specify your contour in the A variable (I use a two column format of UTM-X and UTM-Y, see for example contour_BBGS_v12_UTM.dat)
3. Specify your output name in the fid variable. make sure it has the extension '.geo'. I typically specify the glacier, the resolution and the contour file used to generate the mesh (e.g. BBGS_400_cv12.geo)
4. Run the matlab file to generate the .geo file
5. Now in the terminal, run: gmsh filename.geo -1 -2
6. then run: ElmerGrid 14 2 filename.msh -autoclean

There are other commands listed at the bottom of the makegeo matlab files, but they are no longer needed in the newest versions of Elmer. But now, you should have a 2D footprint of the glacier and end up with the 4 mesh files within a folder:

```
mesh.boundary
mesh.elements
```

```
mesh.header
mesh.nodes
```

Put this folder in the same directory as your SIF file (e.g. the folder BBGS_400_cv12). I put my own version of this folder, i.e. the mesh files for the BBGS simulations, in the repo. The extrusion of the mesh into vertical layers is done within the SIF file in the Simulation section. For 5 vertical layers:

Extruded Mesh Levels = Integer 5

Again, all this is in any standard Elmer Meshing tutorial, but the specifics I use for a given project are given in the following sections.

3 Bering-Bagley Glacier System Model Specifics

Mesh and Contour Outline:

I find that the best resolution for running on the Desktop computers is 400 m and the best amount of vertical layers is 5.

Bed topo DEM:

The bed topography I use is: bed_JPL_v7_fourptsall.dat

The surface DEMs I use for the early 2011 phase is
s0_swath_raddev_v2_4model.dat [Trantow et al., 2020] and for the surge-wave simulation is
s11_VHswath_wfront_dem4model.dat [Trantow and Herzfeld, subm].

Project folders

(1)The specific full-surge simulation files for the 2022 surge paper (JGR) are in the folder elmer_code_JGR

The **quiescent phase** SIF file for the surge paper is called
full_BBGS_s11_veit_swath_bedv7_accumlars_20211005.sif

The file uses the input surface DEM s11_VHswath_wfront_dem4model.dat which is based on Summer 2016 CryoSat-2 data as processed by V. Helm (TFMRA-swath).

The **surge phase** SIF file is called crev_BBGS_surgewave_s0_20170815.sif. This SIF file is the one that produced the limited surge results used in the paper. Note that this run is unstable and may diverge before the simulation ends. It uses an input corresponding to the Summer 2011 CryoSat-2 as processed by ESA Baseline-C s0_BBGS_v2_59959_on_BBGS.dat (note that the 59959 notes a correction to the heights that was made by me based on an error in the Baseline-c data).

The surge simulations use the surge-wave friction law which is contained in the USF_Bering.f file in the surge_wave directory. Note that while this has the same name as USF_Bering.f files elsewhere, it is different in that it has an addition function called SurgeWave. Note that this functions uses an input file (called flowline_BBGS.dat) that specifies the along flowline coordinates and mean-elevation along the glacier. I'll put that FORTRAN code here for reference:

```
!!!!!!!!!!!!!!!!!!!
 ! User Function SurgeWave
! Track propagation of surge wave and assign basal sliding coeff
!!----------------------------------------------------------------------------!!

 FUNCTION SurgeWave( model, n, time ) RESULT(result)
        USE DefUtils
        IMPLICIT None
       TYPE(Model_t) :: model
        TYPE(Solver_t), Target :: Solver
        TYPE(Element_t) :: Element
       TYPE(Element_t), POINTER :: BoundaryElement
       TYPE(Nodes_t) :: ElementNodes
        TYPE(ValueList_t), POINTER :: BC
        INTEGER :: n, i, insize, starti, vel, tdI, fronti, w, length, nboundary
        INTEGER :: effectarea, disti, temp(1), f, BoundaryElementNode
        REAL(KIND=dp) :: time, result
        REAL(KIND=dp) :: x, y, ref
        ! Define variable for start distance along track
       REAL(KIND=dp) :: dstart
        ! Define variables for location of surge front at time td
        REAL(KIND=dp) :: xt, yt, td
       ! Variables for flowline data and basal fric param along flowline and
          distance measure
        REAL(KIND=dp), ALLOCATABLE :: xf(:), yf(:), df(:), dbeta(:), dist(:)
        LOGICAL :: FirstTime=.True.

        SAVE FirstTime
        SAVE xf, yf, df

        ! Time in days (add 0.1 so it is converted to the correct integer)
        td = (365 * time) + 0.1
        tdI = INT(td)

        ! -----------------------
        ! get element information
        ! -----------------------
        BoundaryElement => model % CurrentElement
```

```
IF ( .NOT. ASSOCIATED(BoundaryElement) ) THEN
     CALL FATAL('SurgeWave','No boundary element found')
END IF
BC => GetBC()
IF ( .NOT. ASSOCIATED(BC) ) THEN
     CALL FATAL('SurgeWave','No boundary condition found')
END IF

! Get locations of nodes
nboundary = GetElementNOFNodes(BoundaryElement)
DO BoundaryElementNode=1,nboundary
       IF ( n == BoundaryElement % NodeIndexes(BoundaryElementNode) )
           EXIT
END DO

! get element nodes
! -----------------
CALL GetElementNodes(ElementNodes,BoundaryElement)
x = ElementNodes % x(BoundaryElementNode)
y = ElementNodes % y(BoundaryElementNode)

! Define size of flowline file
insize = 15730

! Define start location as index in flowline data (rift i=9000, 2011 i =
    14000)
starti = 13000

!Define surge wave velocity (divide by the seperation distance, so 50m/
    day => 5)
vel = 5

! Deifine effected area width (divided by 10)
w = 4000

! Define surge front width f (divided by 10)
f = 60

! Load flowline data (along track flow seperated by 10m)
IF (FirstTime) THEN
       FirstTime = .False.
       OPEN(10,file="flowline_BBGS.dat")
       ALLOCATE(xf(insize), yf(insize), df(insize))
       READ(10,*)(xf(i), yf(i), df(i), i=1,insize)
       CLOSE(10)
END IF

! Find start location along track
dstart = df(starti)
```

```
    ! Calculate legth of effected area and location of surge front (and
        index)
    length = vel * tdI
    fronti = starti + length ! Add to go downglacier

   ! Actual length of effected area is min of width an dlengthi
    effectarea = MIN(length,w)

    ! put default pointer back to where it belongs
    ! ---------------------------------------------
    model % CurrentElement => BoundaryElement

   ! Calculate basal sliding coeff for each along flowline point
    ALLOCATE(dbeta(insize))
    ALLOCATE(dist(insize))
    DO i=1, insize
           IF (i > (fronti+f)) THEN
                  dbeta(i) = 0.00001
           ELSE IF (i <=(fronti +f) .AND. i >= fronti) THEN
                 dbeta(i) = 0.0000001 * (1 + 9*(i - fronti)/f)
          ELSE IF (i < (fronti - effectarea)) THEN !Subtract to go
              upglacier
                  dbeta(i) = 0.00001
           ELSE ! Effected area
                  dbeta(i) = 0.0000001 * (1 + 9 * (fronti - i)/w)
          END IF
          !dbeta(i) = 0.0001
           ! find reference to flowline distance point (squared distance)
          dist(i) = (x-xf(i))**2 + (y-yf(i))**2
    END DO

    temp = MINLOC(dist)
    disti = temp(1)
    !IF (dbeta(disti) <0.00005) THEN
    !      write(*,*)'Nodenumber',n,'x',x,'y',y,'Beta',dbeta(disti)
   !END IF

    ! give friction coeff
    result = dbeta(disti)

END FUNCTION SurgeWave
```

(2) Computers & Geosciences runs for varying CryoSat-2 input DEMs:

The USF file used in these runs is also USF_Bering.f90. The SIF file for the TFMRA swath run is called crev_BBGS_C2_swath_IAMG_20181126.sif. Due to the shear number of DEM surface inputs involved in this paper, I do not put them all in this repo. Each one discussed in the paper are available if you contact me at trantow@colorado.edu.

Bibliography

Jerome Bouffard. CRYOSAT-2: Level 2 product evolutions and quality improvements in Baseline C (v3). 2015. 4

Veit Helm, Angelika Humbert, and Heinrich Miller. Elevation and elevation change of Greenland and Antarctica derived from CryoSat-2. *The Cryosphere*, 8(4):1539–1559, 2014. 4

E. Rignot, J. Mouginot, C.F. Larsen, Y. Gim, and D. Kirchner. Low-frequency radar sounding of temperate ice masses in Southern Alaska. *Geophysical Research Letters*, 40(20):5399–5405, 2013. 4

T. Trantow. *Surging in the Bering-Bagley Glacier System, Alaska – Understanding Glacial Acceleration through New Methods in Remote Sensing, Numerical Modeling and Model-Data Comparison*. PhD thesis, University of Colorado, 2020. 1

T. Trantow and U. Herzfeld. Spatiotemporal mapping of a large mountain glacier from CryoSat-2 altimeter data: surface elevation and elevation change of Bering Glacier during surge (2011-2014). *International Journal of Remote Sensing*, pages 0–28, 2016. doi: 10.1080/01431161.2016.1187318. 4

Thomas Trantow and Ute C. Herzfeld. Crevasses as indicators of surge dynamics in the Bering Bagley Glacier System, Alaska: Numerical experiments and comparison to image data analysis. *Journal of Geophysical Research: Earth Surface*, 2018. doi: 10.1029/2017JF004341. 1

Thomas Trantow and Ute C Herzfeld. Evolution of a surge cycle of the bering-bagley glacier system from observations and numerical modeling. *Journal of Geophysical Research Earth Surfaces*, subm. 1, 2, 6

Thomas Trantow, Ute C Herzfeld, Veit Helm, and Johan Nilsson. Sensitivity of glacier elevation analysis and numerical modeling to cryosat-2 siral retracking techniques. *Computers & Geosciences*, page 104610, 2020. 1, 2, 6