

ISSM Workshop 2014

Ice Sheet System Model

Ice flow models

Mathieu MORLIGHEM¹

¹University of California, Irvine



Outline

1 Ice flow equations

- Approximations implemented

- Ice flow equation

- Diagnostic parameters

- Boundary conditions

2 Combining models

- Methods implemented in ISSM

- Penalties

- Tiling method

- Utilization

Ice Sheet flow equations

Incompressibility

$$\forall \mathbf{x} \in \Omega \quad \nabla \cdot \mathbf{v} = \text{Tr}(\dot{\epsilon}) = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (1)$$

- $\mathbf{v} = (u, v, w)$ ice velocity (m/yr)
- $\dot{\epsilon}$ strain rate tensor (yr^{-1})

Incompressible viscous fluid

$$\boldsymbol{\sigma}' = 2\mu\dot{\epsilon} \quad (2)$$

- $\boldsymbol{\sigma}'$ deviatoric stress
- μ ice viscosity
- $\dot{\epsilon}$ strain rate tensor

Glen's flow law

$$\mu = \frac{B}{2 \dot{\epsilon}_e^{\frac{n-1}{n}}} \quad (3)$$

- B ice hardness
- n Glen's law coefficient ($n = 3$)
- $\dot{\epsilon}_e$ effective strain rate (related to the second invariant)

Ice Sheet flow equations

Conservation of momentum

$$\forall \mathbf{x} \in \Omega \quad \nabla \cdot \boldsymbol{\sigma}' - \nabla P + \rho \mathbf{g} = \mathbf{0} \quad (4)$$

Assumptions:

- 1 Stokes flow (quasi-static assumption)
- 2 Coriolis effect negligible

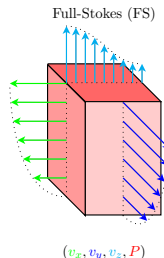
Boundary conditions

Ice/Air interface: Free surface	Γ_s	$\boldsymbol{\sigma} \cdot \mathbf{n} = P_{atm} \mathbf{n} \simeq \mathbf{0}$
Ice/Ocean interface: water pressure	Γ_w	$\boldsymbol{\sigma} \cdot \mathbf{n} = P_w \mathbf{n}$
Ice/Bedrock interface (1): lateral friction	Γ_b	$(\boldsymbol{\sigma} \cdot \mathbf{n} + \beta \mathbf{v})_{\parallel} = \mathbf{0}$
Ice/Bedrock interface (2): impenetrability	Γ_b	$\mathbf{v} \cdot \mathbf{n} = 0$
Side boundaries: Dirichlet	Γ_u	$\mathbf{v} = \mathbf{v}_{obs}$

Models description

Full-Stokes model (FS):

- Momentum balance + incompressibility
- 3D model
- Four unknowns (v_x, v_y, v_z, p)



Model equations

$$\left\{ \begin{array}{l} \frac{\partial}{\partial x} \left(2\mu \frac{\partial v_x}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) - \frac{\partial p}{\partial x} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial y} \left(2\mu \frac{\partial v_y}{\partial y} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) - \frac{\partial p}{\partial y} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) + \frac{\partial}{\partial z} \left(2\mu \frac{\partial v_z}{\partial z} \right) - \frac{\partial p}{\partial z} - \rho g = 0 \end{array} \right.$$

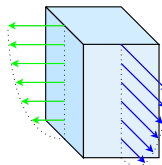
$$\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} = 0$$

Models description

Higher-order model (HO):

- [Blatter, 1995, Pattyn, 2003]
- 3D model
- Horizontal and vertical velocity decoupled
- 2 (v_x, v_y) + 1 (v_z) unknowns

Blatter-Pattyn (BP)



(v_x, v_y)

Model equations

$$\begin{cases} \frac{\partial}{\partial x} \left(2\mu \frac{\partial v_x}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) - \frac{\partial p}{\partial x} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial y} \left(2\mu \frac{\partial v_y}{\partial y} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) - \frac{\partial p}{\partial y} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) + \frac{\partial}{\partial z} \left(2\mu \frac{\partial v_z}{\partial z} \right) - \frac{\partial p}{\partial z} - \rho g = 0 \end{cases}$$

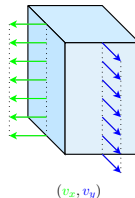
$$\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} = 0$$

Models description

MacAyeal-Morland (SSA)

Shelfy-stream approximation (SSA):

- [MacAyeal, 1989]
- 2D model
- Horizontal and vertical velocity decoupled
- 2 (v_x, v_y) + 1 (v_z) unknowns



Model equations

$$\left\{ \begin{array}{l} \frac{\partial}{\partial x} \left(2\mu \frac{\partial v_x}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) - \frac{\partial p}{\partial x} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial y} \left(2\mu \frac{\partial v_y}{\partial y} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) - \frac{\partial p}{\partial y} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) + \frac{\partial}{\partial z} \left(2\mu \frac{\partial v_z}{\partial z} \right) - \frac{\partial p}{\partial z} - \rho g = 0 \end{array} \right.$$

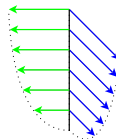
$$\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} = 0$$

Models description

Hutter (SIA)

Shallow ice approximation (SIA):

- [Hutter, 1983]
- 3D analytical model
- 2 unknowns (v_x, v_y) computed separately



(v_x, v_y)

Model equations

$$\left\{ \begin{array}{l} \frac{\partial}{\partial x} \left(2\mu \frac{\partial v_x}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) - \frac{\partial p}{\partial x} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial y} \left(2\mu \frac{\partial v_y}{\partial y} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) - \frac{\partial p}{\partial y} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) + \frac{\partial}{\partial z} \left(2\mu \frac{\partial v_z}{\partial z} \right) - \frac{\partial p}{\partial z} - \rho g = 0 \end{array} \right.$$

$$\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} = 0$$

Material non-linearity

Model equations

$$\begin{cases} \frac{\partial}{\partial x} \left(2\mu \frac{\partial v_x}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) - \frac{\partial p}{\partial x} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial y} + \mu \frac{\partial v_y}{\partial x} \right) + \frac{\partial}{\partial y} \left(2\mu \frac{\partial v_y}{\partial y} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) - \frac{\partial p}{\partial y} = 0 \\ \frac{\partial}{\partial x} \left(\mu \frac{\partial v_x}{\partial z} + \mu \frac{\partial v_z}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v_y}{\partial z} + \mu \frac{\partial v_z}{\partial y} \right) + \frac{\partial}{\partial z} \left(2\mu \frac{\partial v_z}{\partial z} \right) - \frac{\partial p}{\partial z} - \rho g = 0 \\ \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} = 0 \end{cases}$$

Glen's flow law

$$\mu = \frac{B}{2 \dot{\epsilon}_e^{\frac{n-1}{n}}} \quad (5)$$

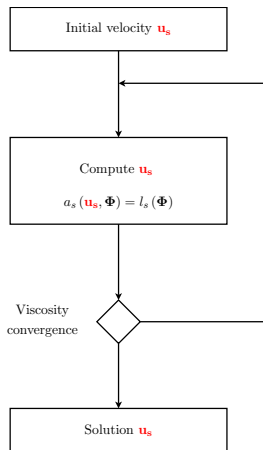
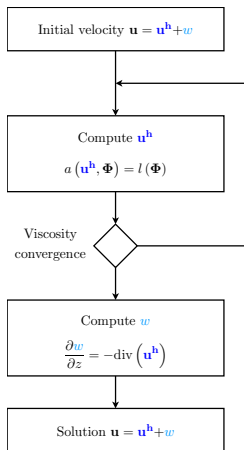
- B ice hardness
- n Glen's law coefficient ($n = 3$)
- $\dot{\epsilon}_e$ effective strain rate (second invariant)

ISSM › Treatment of non-linearity with fixed point (Picard) or Newton's method



Material non-linearity

Treatment of non-linearity with fixed point:



Vertical velocity computed with incompressibility for SSA and HO

Flow equation

`setflowequation` is used to generate the approximation used to compute the velocity

- Arguments:
 - 1 model
 - 2 approximation names
 - 3 approximation domains
- Domains can be Argus files or array of element flags
- Approximation available
 - FS (Full-Stokes model)
 - HO (Higher-order model)
 - SSA (Shallow Shelf Approximation)
 - SIA (Shallow Ice Approximation)
- Possibility of coupling models

Flow equation

`setflowequation` is used to generate the approximation used to compute the velocity

- Examples

```
1 md=setflowequation(md, 'SIA', 'all')
2 md=setflowequation(md, 'FS', 'all')
3 md=setflowequation(md, 'SSA', 'all')
4 md=setflowequation(md, 'HO', 'all')
```

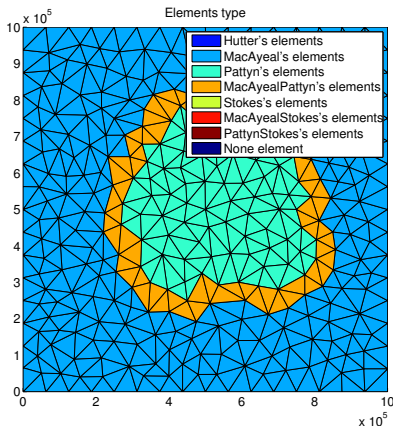
- To display the type of approximation:

```
1 >> plotmodel(md, 'data', 'elements_type')
```

Flow equation

- To display the type of approximation:

```
1 >> plotmodel(md, 'data', 'elements_type')
```



Flow equation class

```

1  >> md.flowequation
2
3  ans =
4
5      flow equation parameters:
6          isSIA           : 0          -- is the Shallow Ice Approximation (SIA) ?
7          isSSA           : 1          -- is the Shelfy-Stream Approximation (SSA)?
8          isL1L2          : 0          -- is the L1L2 approximation used ?
9          isHO            : 0          -- is the Higher-Order (HO) approximation ?
10         isFS            : 0          -- are the Full-FS (FS) equations used ?
11         fe_SSA          : 'P1'       -- Finite Element for SSA 'P1', 'Plbubble'...
12         fe_HO           : 'P1'       -- Finite Element for HO 'P1', 'Plbubble'...
13         fe_FS           : 'MINIcondensed' -- Finite Element for FS 'MINIcondensed'...
14         vertex_equation : N/A        -- flow equation for each vertex
15         element_equation : N/A       -- flow equation for each element
16         borderSSA       : N/A       -- vertices on SSA's border (for tiling)
17         borderHO        : N/A       -- vertices on HO's border (for tiling)
18         borderFS        : N/A       -- vertices on FS' border (for tiling)

```

StressBalance class

```

1  >> md.stressbalance
2
3  ans =
4
5  StressBalance solution parameters:
6
7  Convergence criteria:
8      restol          : 0.0001      -- mechanical equilibrium residual convergence criterion
9      reltol          : 0.01        -- velocity relative convergence criterion, NaN: not applied
10     abstol          : 10          -- velocity absolute convergence criterion, NaN: not applied
11     isnewton         : 0           -- 0: Picard's fixed point, 1: Newton's method, 2: hybrid
12     maxiter          : 100        -- maximum number of nonlinear iterations
13     viscosity_overshoot : 0       -- over-shooting constant new-new+C*(new-old)
14
15  boundary conditions:
16     spcvx            : N/A         -- x-axis velocity constraint (NaN means no constraint) [m/yr]
17     spcvy            : N/A         -- y-axis velocity constraint (NaN means no constraint) [m/yr]
18     spcvz            : N/A         -- z-axis velocity constraint (NaN means no constraint) [m/yr]
19
20  Rift options:
21     rift_penalty_threshold : 0      -- threshold for instability of mechanical constraints
22     rift_penalty_lock      : 10     -- number of iterations before rift penalties are locked
23
24  Penalty options:
25     penalty_factor        : 3       -- offset used by penalties: penalty = Kmax*10^offset
26     vertex_pairing        : N/A     -- pairs of vertices that are penalized
27
28  Other:
29     shelf_dampening       : 0       -- use dampening for floating ice ? Only for FS model
30     FSreconditioning      : 1000000000000000 -- multiplier for incompressibility equation. Only for FS model
31     referential           : N/A     -- local referential
32     loadingforce          : N/A     -- loading force applied on each point [N/m^3]
33     requested_outputs     : {'default'} -- additional outputs requested

```

Boundary conditions

Boundary conditions created automatically or manually

- Automatically:

```
1 >> md=SetIceSheetBC(md)
2 >> md=SetIceShelfBC(md, 'Front.exp')
3 >> md=SetMarineIceSheefBC(md, 'Front.exp')
```

- Manually: fields to change

- md.stressbalance.spcvx
- md.stressbalance.spcvy
- md.stressbalance.spcvz
- md.mask.ice_levelset

- To display the boundary conditions

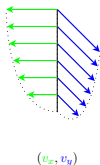
```
1 >> plotmodel(md, 'data', 'BC')
```

Models description

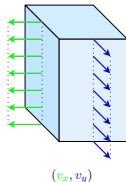
"Everything should be made as simple as possible, but no simpler." Albert Einstein

Model	Dim.	Unknowns	Reference
FS (Full-Stokes)	3d	4	[Stokes, 1845]
HO (Blatter-Pattyn)	3d	2 + 1	[Blatter, 1995, Pattyn, 2003]
SSA (Shallow shelf)	2d	2 + 1	[MacAyeal, 1989]
SIA (Shallow ice)	2d	2 + 1	[Hutter, 1983]

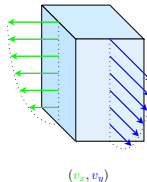
Hutter (SIA)



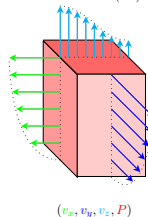
MacAyeal-Morland (SSA)



Blatter-Pattyn (BP)

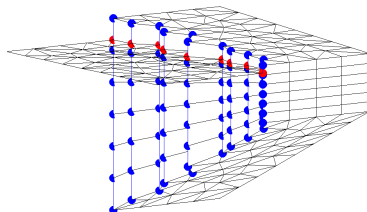
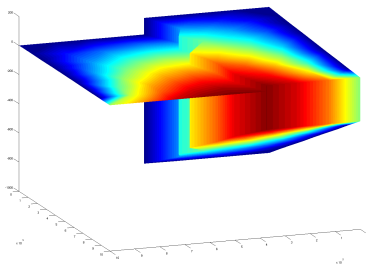


Full-Stokes (FS)



Penalty method

- Only to couple SSA and HO
- Very stiff spring to penalize differences between velocities at the interface

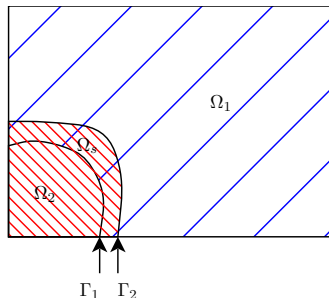
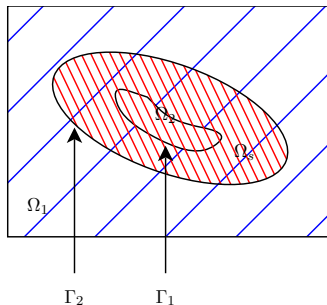


Using penalties to couple models:

```
1 md=setflowequation(md,'SSA','FloatingIce.exp','fill','HO','coupling','penalties')
```

Domain Decomposition

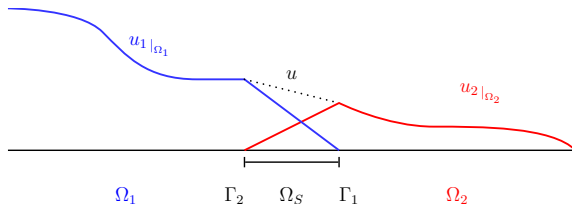
- $\Omega = \Omega_1 \cup \Omega_2$
- $\Omega_S = \Omega_1 \cap \Omega_2 \neq \emptyset$
- $\mathbf{u} = \mathbf{u}_1|_{\Omega_1} + \mathbf{u}_2|_{\Omega_2} \in \tilde{V}(\Omega) = (V_1(\Omega_1) + V_2(\Omega_2))$



Find $\mathbf{u} = \mathbf{u}_1|_{\Omega_1} + \mathbf{u}_2|_{\Omega_2} \in \tilde{V}$,

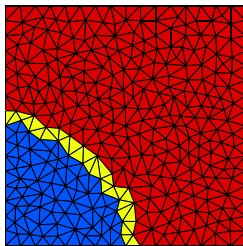
$$\forall (\mathbf{v}_1, \mathbf{v}_2) \in \tilde{V} \quad a(\mathbf{u}_1 + \mathbf{u}_2, \mathbf{v}_1 + \mathbf{v}_2) = l(\mathbf{v}_1 + \mathbf{v}_2)$$

Discretization



We take advantage of the discretization to avoid the redundancy:

- Create one layer of elements in the superposition zone



Multi-model formulation

Two different models: a_1 , a_2 and l_1 , l_2

Find $\mathbf{u} = \mathbf{u}_1|_{\Omega_1} + \mathbf{u}_2|_{\Omega_2} \in (V_1 + V_2)$, such that:

$$\forall \mathbf{v} = \mathbf{v}_1|_{\Omega_1} + \mathbf{v}_2|_{\Omega_2} \in (V_1 + V_2)$$

$$\underbrace{a_1(\mathbf{u}_1|_{\Omega_1}, \mathbf{v}_1|_{\Omega_1})}_{\text{model 1}} + \underbrace{a_2(\mathbf{u}_2|_{\Omega_2}, \mathbf{v}_2|_{\Omega_2})}_{\text{model 2}} +$$

$$\underbrace{a_2(\mathbf{u}_1|_{\Omega_1}, \mathbf{v}_2|_{\Omega_2}) + a_1(\mathbf{u}_2|_{\Omega_2}, \mathbf{v}_1|_{\Omega_1})}_{\text{model coupling}}$$

$$= \underbrace{l_1(\mathbf{v}_1|_{\Omega_1})}_{\text{model 1}} + \underbrace{l_2(\mathbf{v}_2|_{\Omega_2})}_{\text{model 2}}$$

- Coupling different mechanical models
- Easy to implement (local modification of stiffness matrices)

Flow equation

`setflowequation` is used to generate the approximation used to compute the velocity

- Examples

```
1 md=setflowequation(md,'HO',md.mask.groundedice_levelset>0,'fill','SSA','coupling','penalties')
2 md=setflowequation(md,'HO',md.mask.groundedice_levelset>0,'fill','SSA','coupling','tiling')
3 md=setflowequation(md,'FS','Contour.exp','fill','HO')
```

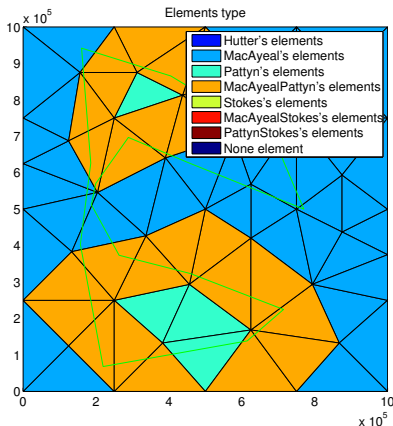
- Use `exptool` to create EXP contours

```
1 >> exptool('Contour.exp')
```

Flow equation

- To display the type of approximation:

```
1 >> plotmodel(md, 'data', 'elements_type', 'edgecolor', 'k', 'expdisp', 'Contour.exp')
```



Bibliography I



Blatter, H. (1995).

Velocity and stress-fields in grounded glaciers: A simple algorithm for including deviatoric stress gradients.

J. Glaciol., 41(138):333–344.



Hutter, K. (1983).

Theoretical glaciology: material science of ice and the mechanics of glaciers and ice sheets.

D. Reidel Publishing Co, Dordrecht, The Netherlands.



MacAyeal, D. (1989).

Large-scale ice flow over a viscous basal sediment: Theory and application to Ice Stream B, Antarctica.

J. Geophys. Res., 94(B4):4071–4087.



Pattyn, F. (2003).

A new three-dimensional higher-order thermomechanical ice sheet model: Basic sensitivity, ice stream development, and ice flow across subglacial lakes.

J. Geophys. Res., 108(B8):1–15.

Bibliography II



Stokes, G. (1845).

On the theories of internal friction of fluids in motion.

Trans. Cambridge Philos. Soc., 8:287–305.

Thanks!

