Ice Sheet System Model 2008
Theory Guide

Authors:

Éric Larour[1]
Mathieu Morlighem[2,4]
Hélène Seroussi[2,4]
Éric Rignot[2,3]

[1]Division 35, Thermal and Cryogenics Section,
Mechanical Division, MS 157-316.
Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena, CA 91109.

[2]Division 33, Radar Science and Enginnering Section,
Communications, Tracking and Radar Division, MS 157-316.
Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena, CA 91109.

[3]University of California, Irvine
Department of Earth System Science
Croul Hall, Irvine, CA 92697-3100

[4]Laboratoire de Mécanique des sols, Structures et matériaux (MSSMat)
École Centrale Paris, CNRS UMR 8579
Grande Voie des Vignes, 92295 Châtenay-Malabry Cedex, FRANCE

February 12, 2009

Summary

This manual explains ...

Key-words : ICE, ice flow, finite elements, 2-D, 3-D, matlab

Contents

Introduction

This document...

KEYWORD: Models, weak formulation, Galerkin, finite element, gauss quadrature,Full stokes, ice sheet numerical finite element model,...

Chapter 1

Physical basis

This chapter describes the principles of glacier mechanics. We introduce the constitutive relation of ice: Glen's flow law, as well as some basic concepts such as conservation of mass and momentum balance. These equations are used to deduce the three models used in *ISSM* (Ice Sheet System Model). In a second part, the boundary conditions are presented. The last part is dedicated to a thermodynamic model that was also implemented in *ISSM*.

1.1 Notations and geometry

We consider an ice sheet extending into the ocean. The floating part is an ice shelf. Deformable subglacial till is present below the grounded ice. The geometry is described by the following scheme:

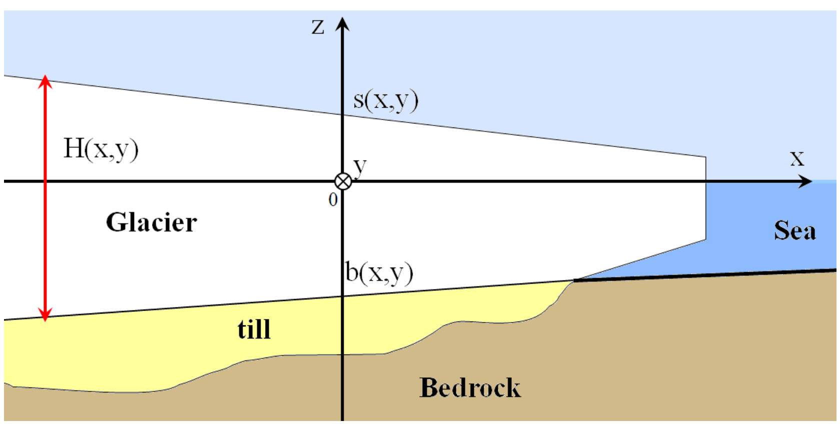


Figure 1.1: Schematic cross section of the glacier

The z axis is vertically pointing upward and the sea level is at $z = 0$, the xy-plane is horizontal. We note:

- $H(x, y)$ glacier thickness
- $s(x, y)$ glacier's upper surface z coordinate
- $b(x, y)$ glacier's lower surface z coordinate

Regarding kinematic quantities, we adopt the following notations:

- $\overrightarrow{v}(x, y, z)$ is the glacier velocity
- $u(x, y, z)$ velocity x-component
- $v(x, y, z)$ velocity y-component

  - $w(x, y, z)$ velocity z-component
  - $\varepsilon(x, y, z)$ strain tensor
  - $\dot{\varepsilon}(x, y, z)$ strain rate tensor
  - $\sigma(x, y, z)$ stress tensor

1.2 Deformation Law

1.2.1 Flow law

The deformation of ice involves the deviatoric stress tensor because hydrostatic pressure does not contribute to ice deformation. It only depends on shear (Hooke, 2005 [4] p13 and Paterson,1994 [10]). The deviatoric stress is:

$$\sigma' = \sigma - \frac{1}{3}Tr(\sigma) = \sigma + P\,[I] \tag{1.1}$$

where σ is the stress tensor and P is the hydrostatic pressure[1].The effective strain and the effective shear stress are defined as follows:

$$\dot{\varepsilon_e} = \frac{1}{\sqrt{2}}\left(\sum_{i,j=1..3} \dot{\varepsilon}_{ij}^2\right)^{1/2} \qquad \sigma'_e = \frac{1}{\sqrt{2}}\left(\sum_{i,j=1..3} \sigma'^2_{ij}\right)^{1/2} \tag{1.2}$$

where $\dot{\varepsilon}$ is the strain rate tensor. The most common flow law is Glen's flow law, based on John W. Glen's experiments (Glen, 1955 [1]). He used the form:

$$\boxed{\dot{\varepsilon_e} = \left(\frac{\sigma'_e}{B}\right)^n} \tag{1.3}$$

where B is a viscosity parameter which increases as the ice becomes stiffer ($MPa.a^{\frac{1}{n}}$), n is empirically determined(most studies have found that $n \simeq 3$, see Hooke, 2005 [4] p15 and Paterson, 1994 [10] p86) and is called flow law exponent.

1.2.2 Constitutive relation of ice

We consider that ice is an isotropic and incompressible material. This implies that the principal axes of the deviatoric stress σ' and the strain rate tensor $\dot{\varepsilon}$ coincide. The ice that constitutes glaciers is not isotropic and incompressible; but this approximation is a convenient starting point for calculations of glacier flow. If the principal axes of deviatoric stress and strain rate coincide, the flow law becomes (Hooke, 2005 [4], p269):

$$\dot{\varepsilon} = \frac{{\sigma'_e}^{n-1}}{B^n}\sigma' \tag{1.4}$$

Then, combining Glen's flow law to eliminate σ_e, we obtain:

$$\dot{\varepsilon} = \frac{\dot{\varepsilon_e}^{\frac{n-1}{n}}}{B}\sigma' \tag{1.5}$$

Following MacAyeal (MacAyeal, 1989 [7]), we define the viscosity μ as:

$$\boxed{\mu = \frac{B}{2\left(\dot{\varepsilon_e}^{1-\frac{1}{n}}\right)}} \tag{1.6}$$

Then the constitutive relation for ice becomes:

[1] $Tr(\sigma) = \sum_i \sigma_{ii}$ is the stress trace

$$\boxed{\sigma' = 2\mu\dot{\varepsilon}} \tag{1.7}$$

1.3 Local Conservation of Mass and Momentum

1.3.1 Mass balance

The most general equation of local mass balance includes a production term $\dot{M}$. With ρ the density of ice, the equation is:

$$\frac{\partial\rho}{\partial t} + div\left(\rho\overrightarrow{v}\right) = \dot{M} \tag{1.8}$$

In a glacier, there is no production of mass locally (surface accumulation and basal melting are not involved in local mass balance, they are treated as boundary conditions). The ice is also treated as incompressible: ρ is constant (Hooke, 2005 [4] p14). The previous equation can be rewritten as follows:

$$\boxed{div\left(\overrightarrow{v}\right) = 0} \tag{1.9}$$

1.3.2 Momentum balance

The fundamental equation of momentum conservation (with $\overrightarrow{g}$ the acceleration due to gravity):

$$\underline{Div}\left(\sigma\right) + \rho\overrightarrow{g} = \rho\frac{d\overrightarrow{v}}{dt} \tag{1.10}$$

For a glacier, the acceleration is small. Thus the acceleration term is negligible compared to the other terms: the ice is described by a quasi-static model (Paterson,1994 [10] p258). If we use the deviatoric stress defined in (1.1), the momentum balance becomes:

$$\underline{Div}\left(\sigma'\right) - \overrightarrow{grad}P + \rho\overrightarrow{g} = 0 \tag{1.11}$$

using the constitutive relation (1.7) gives:

$$\begin{cases} 2\underline{Div}\left(\mu\dot{\varepsilon}\right) - \overrightarrow{grad}P + \rho\overrightarrow{g} = 0 \\ \\ \mu = \dfrac{B}{2\left(\dot{\varepsilon_e}^{1-\frac{1}{n}}\right)} \end{cases} \tag{1.12}$$

The momentum balance can hence be described by three different sets of equations that are strictly equivalent to each other:

Momentum balance in terms of stress

$$\boxed{\begin{cases} \dfrac{\partial\sigma_{xx}}{\partial x} + \dfrac{\partial\sigma_{xy}}{\partial y} + \dfrac{\partial\sigma_{xz}}{\partial z} = 0 \\ \\ \dfrac{\partial\sigma_{yx}}{\partial x} + \dfrac{\partial\sigma_{yy}}{\partial y} + \dfrac{\partial\sigma_{yz}}{\partial z} = 0 \\ \\ \dfrac{\partial\sigma_{zx}}{\partial x} + \dfrac{\partial\sigma_{zy}}{\partial y} + \dfrac{\partial\sigma_{zz}}{\partial z} - \rho g = 0 \end{cases}} \tag{1.13}$$

Momentum balance in terms of deviatoric stress

$$\boxed{\left\{\begin{array}{l} \dfrac{\partial \sigma'_{xx}}{\partial x} + \dfrac{\partial \sigma'_{xy}}{\partial y} + \dfrac{\partial \sigma'_{xz}}{\partial z} - \dfrac{\partial P}{\partial x} = 0 \\[2ex] \dfrac{\partial \sigma'_{yx}}{\partial x} + \dfrac{\partial \sigma'_{yy}}{\partial y} + \dfrac{\partial \sigma'_{yz}}{\partial z} - \dfrac{\partial P}{\partial y} = 0 \\[2ex] \dfrac{\partial \sigma'_{zx}}{\partial x} + \dfrac{\partial \sigma'_{zy}}{\partial y} + \dfrac{\partial \sigma'_{zz}}{\partial z} - \dfrac{\partial P}{\partial z} - \rho g = 0 \end{array}\right.} \tag{1.14}$$

Momentum balance in terms of strain rate

$$\boxed{\left\{\begin{array}{l} 2\dfrac{\partial \mu \dot{\varepsilon}_{xx}}{\partial x} + 2\dfrac{\partial \mu \dot{\varepsilon}_{xy}}{\partial y} + 2\dfrac{\partial \mu \dot{\varepsilon}_{xz}}{\partial z} - \dfrac{\partial P}{\partial x} = 0 \\[2ex] 2\dfrac{\partial \mu \dot{\varepsilon}_{yx}}{\partial x} + 2\dfrac{\partial \mu \dot{\varepsilon}_{yy}}{\partial y} + 2\dfrac{\partial \mu \dot{\varepsilon}_{yz}}{\partial z} - \dfrac{\partial P}{\partial y} = 0 \\[2ex] 2\dfrac{\partial \mu \dot{\varepsilon}_{zx}}{\partial x} + 2\dfrac{\partial \mu \dot{\varepsilon}_{zy}}{\partial y} + 2\dfrac{\partial \mu \dot{\varepsilon}_{zz}}{\partial z} - \dfrac{\partial P}{\partial z} - \rho g = 0 \end{array}\right.} \tag{1.15}$$

1.4 Time evolution of ice thickness

The thickness of ice varies in time because of: (1) basal melting, (2) surface accumulation and (3) the flow of the glacier itself. The mass balance of an ice column of section $dxdy$ and height H between a time t and $t+dt$ is:

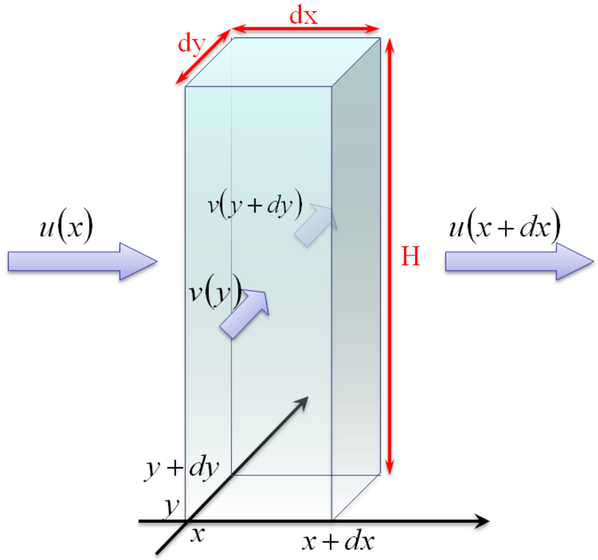


Figure 1.2: Mass balance of an ice column

$$m(t+dt)-m(t) = \underbrace{dy\int_{b(x)}^{s(x)}\rho u\left(x\right)dt\;dz + dx\int_{b(y)}^{s(y)}\rho v\left(y\right)dt\;dz}_{Input\quad mass} + \underbrace{\rho\dot{M}_s dxdydt}_{Accumulation}$$
$$\underbrace{-dy\int_{b(x+dx)}^{s(x+dx)}\rho u\left(x+dx\right)dt\;dz - dx\int_{b(y+dy)}^{s(y+dy)}\rho v\left(y+dy\right)dt\;dz}_{Output\quad mass} - \underbrace{\rho\dot{M}_b dxdydt}_{Melting} \tag{1.16}$$

where $\dot{M}_s$ is the local accumulation-ablation function at the surface ($m.a^{-1}$ ice equivalent, positive when accumulation) and $\dot{M}_b$ is the local melting rate at the base of the glacier ($m.a^{-1}$ ice equivalent positive when melting).
If one uses the relation of incompressibility:

$$\begin{aligned}\frac{m(t+dt)-m(t)}{\rho dxdydt} = \frac{\partial H}{\partial t} &= -\frac{1}{dx}\left(\int_{b(x+dx)}^{s(x+dx)} u\left(x+dx\right)dz - \int_{b(x)}^{s(x)} u\left(x\right)dz\right)\\ &\quad -\frac{1}{dy}\left(\int_{b(y+dy)}^{s(y+dy)} v\left(y+dy\right)dz - \int_{b(y)}^{s(y)} v\left(y\right)dz\right) + \dot{M}_s - \dot{M}_b\end{aligned}$$

We now introduce the depth-averaged velocity $H\overline{u} = \int_b^s u dz$ and $H\overline{v} = \int_b^s v dz$, and obtain:

$$\frac{\partial H}{\partial t} = -\frac{1}{dx}\left(\left(H\overline{u}\right)\left(x+dx\right) - \left(H\overline{u}\right)\left(x\right)\right) - \frac{1}{dy}\left(\left(H\overline{v}\right)\left(y+dy\right) - \left(H\overline{v}\right)\left(y\right)\right) + \dot{M}_s - \dot{M}_b \tag{1.17}$$

The evolution of ice thickness is then written as :

$$\boxed{\frac{\partial H}{\partial t} = -div\left(H\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right) + \dot{M}_s - \dot{M}_b} \tag{1.18}$$

1.5 Boundary Conditions

All the equations governing ice flow were described in the previous section. The mass conservation, the momentum balance and the constitutive relation of ice give a set of equations that describes completely the ice sheet system dynamics. But one needs boundary conditions to constrain these equations. There are many boundary conditions depending on the interface: ice/atmosphere boundary (air pressure), ice/ocean boundary (water pressure) and ice till boundary (friction). All these boundary conditions are described in this section.

1.5.1 Ice/Atmosphere boundary

The upper surface of the glacier can be viewed as a free surface (since the atmospheric pressure P_0 is negligible), the boundary condition is hence:

$$\sigma \overrightarrow{n} = \sigma' \overrightarrow{n} - P \overrightarrow{n} = P_0 \overrightarrow{n} \simeq \overrightarrow{0} \tag{1.19}$$

The upper surface of the glacier has the following equation : $z - s\,(x, y, t) = 0$. The unit normal vector pointing outward from the glacier $\overrightarrow{n}$ is the gradient of the previous equation:

$$\overrightarrow{n} = \frac{1}{\left(\left(\frac{\partial s}{\partial x} \right)^2 + \left(\frac{\partial s}{\partial y} \right)^2 + 1 \right)^{1/2}} \begin{pmatrix} -\frac{\partial s}{\partial x} \\ -\frac{\partial s}{\partial y} \\ 1 \end{pmatrix} \tag{1.20}$$

1.5.2 Ice front boundary

Water applies a pressure on the ice front equal to $P_w = -\rho_w g z$:

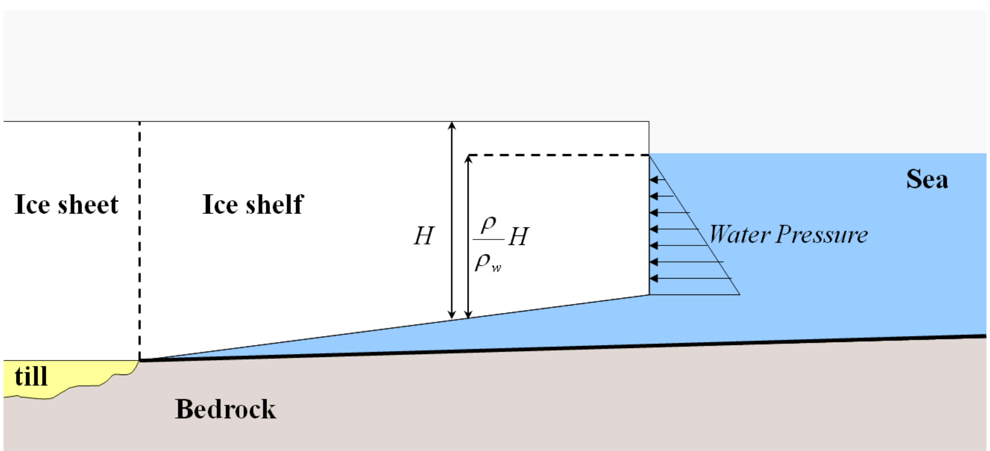


Figure 1.3: Water pressure on the front of an ice shelf

The boundary condition is thus:

$$\sigma \overrightarrow{n} = \sigma' \overrightarrow{n} - P \overrightarrow{n} = -P_w \overrightarrow{n} \tag{1.21}$$

1.5.3 Ice/Water boundary

The exact Ice/water boundary condition is a Neumann condition[2]. Unfortunately, if one uses the incompressibility equation to deduce the vertical velocity (e.g. Pattyn), a Dirichlet boundary condition[3] is required. The only way to provide a Dirichlet boundary condition is to assume hydrostatic equilibrium. This gives an expression of the basal vertical velocity.

1.5.3.1 Neumann boundary condition

The lower surface of the ice shelf in contact with water is subject to the pressure of the water $P_w = -\rho_w g b(x,y)$ where ρ_w is the water density and $b(x,y)$ the glacier base:

$$\sigma \overrightarrow{n} = \sigma' \overrightarrow{n} - P\,[I]\,\overrightarrow{n} = -P_w \overrightarrow{n} \tag{1.22}$$

When there is a hydrostatic equilibrium, the water and ice pressure are equal, the boundary condition becomes:

$$\sigma' \overrightarrow{n} = \overrightarrow{0} \tag{1.23}$$

The lower surface of the ice shelf verifies: $z - b\,(x,y,t) = 0$. The unit normal vector pointing outward $\overrightarrow{n}$ is the opposite of the gradient of the previous equation:

$$\overrightarrow{n} = \frac{1}{\left(\left(\frac{\partial b}{\partial x}\right)^2 + \left(\frac{\partial b}{\partial y}\right)^2 + 1\right)^{1/2}} \begin{pmatrix} \frac{\partial b}{\partial x} \\ \frac{\partial b}{\partial y} \\ -1 \end{pmatrix} \tag{1.24}$$

1.5.3.2 Dirichlet boundary condition assuming hydrostatic equilibrium

Using the Lagrangian derivative, the vertical basal velocity is:

$$w_b = \frac{db}{dt} = \frac{\partial b}{\partial t} + u_b \frac{\partial b}{\partial x} + v_b \frac{\partial b}{\partial y} \tag{1.25}$$

The ice shelf lies on sea water. We do not have any kinematic information if we do not make any assumption. We suppose as a first order approximation that there is a hydrostatic equilibrium. This imposes:

$$\rho_i H = -\rho_w b \tag{1.26}$$

where H is the ice thickness and b the vertical position of the bed ($b < 0$). If one differentiates this equation, given that ρ_i and ρ_w are constant, one finds :

$$\frac{\partial b}{\partial t} = -\frac{\rho_i}{\rho_w} \frac{\partial H}{\partial t} \tag{1.27}$$

The mass balance of an ice column gives the local ice thickness evolution:

$$\frac{\partial H}{\partial t} = -div\left(H \begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix}\right) + \dot{M}_s - \dot{M}_b \tag{1.28}$$

[2] A Neumann boundary condition specifies the values that the derivative of a solution is to take on the boundary of the domain

[3] A Dirichlet boundary condition specifies the values a solution needs to take on the boundary of the domain

We then use the equation (1.28) to evaluate the evolution of the ice thickness with time. With no accumulation and melting, the basal vertical velocity (1.25) is:

$$w_b = u_b \frac{\partial b}{\partial x} + v_b \frac{\partial b}{\partial y} - \frac{\rho_i}{\rho_w} \left(-div \left(H \bar{u} \right) \right) \tag{1.29}$$

Now let us consider the influence of accumulation and melting. If an accumulation of thickness h appears on an ice shelf in hydrostatic equilibrium, it will sink by a height equal to $\rho_i/\rho_w h$:

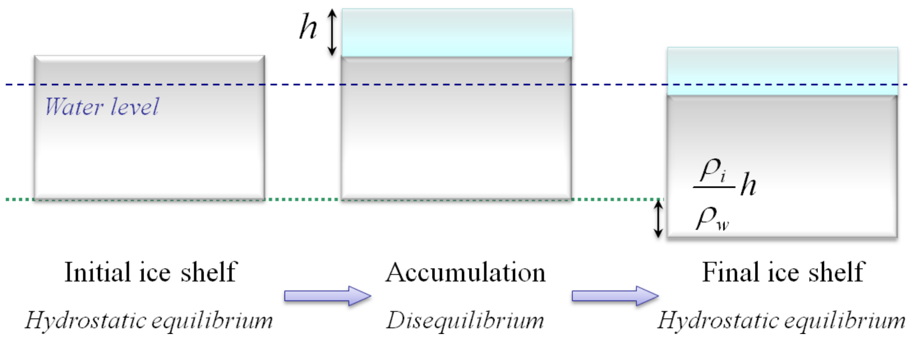


Figure 1.4: Evolution of an ice shelf subjected to accumulation

Same thing for melting: if a thickness h melts at the ice shelf base in hydrostatic equilibrium, it will sink by a height equal to $(1 - \rho_i/\rho_w)h$:

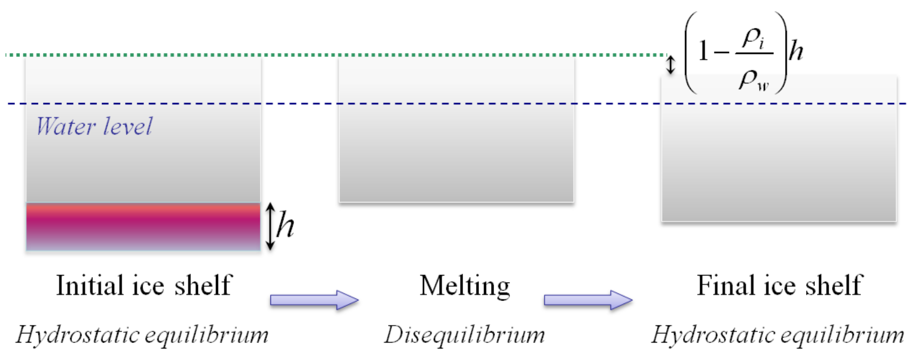


Figure 1.5: Evolution of an ice shelf subjected to melting

One uses the derivative of the previous equations with respect to time. If there is accumulation at the ice surface, the ice will sink into water with a velocity equal to:

$$w_b = -\frac{\rho_i}{\rho_w} \dot{M}_a \tag{1.30}$$

If ice melts at the bottom, the ice will sink with a velocity:

$$w_b = -\left(1 - \frac{\rho_i}{\rho_w}\right) \dot{M}_b \tag{1.31}$$

Eventually, the basal vertical velocity at ice/water interface is:

$$\boxed{w_b = u_b \frac{\partial b}{\partial x} + v_b \frac{\partial b}{\partial y} - \frac{\rho_i}{\rho_w} \left(-div \left(H \left[\begin{array}{c} \bar{u} \\ \bar{v} \end{array} \right] \right) \right) - \frac{\rho_i}{\rho_w} \dot{M}_a - \left(1 - \frac{\rho_i}{\rho_w}\right) \dot{M}_b} \tag{1.32}$$

1.5.4 Ice/Till interface

The Ice/Till interface is constrained by two boundary conditions. The first one is a Neumann boundary condition describing basal drag (tangent to the interface) and the other one is a Dirichlet boundary condition constraining the vertical velocity.

1.5.4.1 Dynamic boundary condition

The basal friction given by Paterson[10] (p151) is:

$$\|\overrightarrow{u_b}\| = K N_{eff}^{-q} \|\overrightarrow{\tau_b}\|^p \tag{1.33}$$

- $\overrightarrow{u_b}$ is the velocity component, parallel to the bedrock surface $z = b(x, y)$ $(\overrightarrow{v} = \overrightarrow{u_b} + (\overrightarrow{v}.\overrightarrow{n})\,\overrightarrow{n})$
- $N_{eff} = g(\rho H(x, y) + \rho_w b(x, y))$ is the effective pressure at the base
- $\overrightarrow{\tau_b}$ is the friction stress component, parallel to the bedrock surface $(\sigma\,\overrightarrow{n} = \overrightarrow{\tau_b} + (\sigma\,(\overrightarrow{n}).\overrightarrow{n})\,\overrightarrow{n})$
- K, q and p are constants

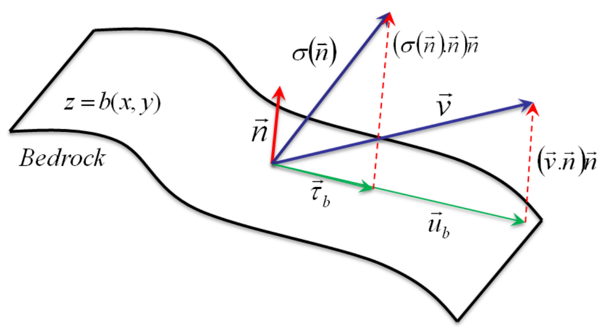


Figure 1.6: Ice/Till interface geometry

This friction law may be written in vector form:

$$\overrightarrow{\tau_b} \quad = \quad -K^2 N_{eff}^r \|\overrightarrow{v}\|^{s-1} \overrightarrow{u_b} \quad \overset{def}{=} \quad -\alpha^2 \overrightarrow{u_b} \tag{1.34}$$

with $r = q/p$ and $s = 1/p$. We use K^2 to be sure that this quantity is positive (ie. stress opposes the motion). The unit normal vector $\overrightarrow{n}$ is:

$$\overrightarrow{n} = \frac{1}{\left(\left(\dfrac{\partial b}{\partial x}\right)^2 + \left(\dfrac{\partial z_b}{\partial y}\right)^2 + 1 \right)^{1/2}} \begin{pmatrix} -\dfrac{\partial b}{\partial x} \\ -\dfrac{\partial b}{\partial y} \\ +1 \end{pmatrix} = \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} \tag{1.35}$$

Several approximations can be made depending on the slope. They are despcribed in section 1.5.5.

1.5.4.2 Kinematic boundary condition

There is also a kinematic boundary condition which constraints the vertical component of the glacier velocity w_b at its base. The following scheme (Fig.1.7) represents a point M on the lower surface of the glacier at a time t and $t + dt$ in the velocity plane (plane $(\overrightarrow{v}, \overrightarrow{e_z})$). The accumulation/ablation is not represented.

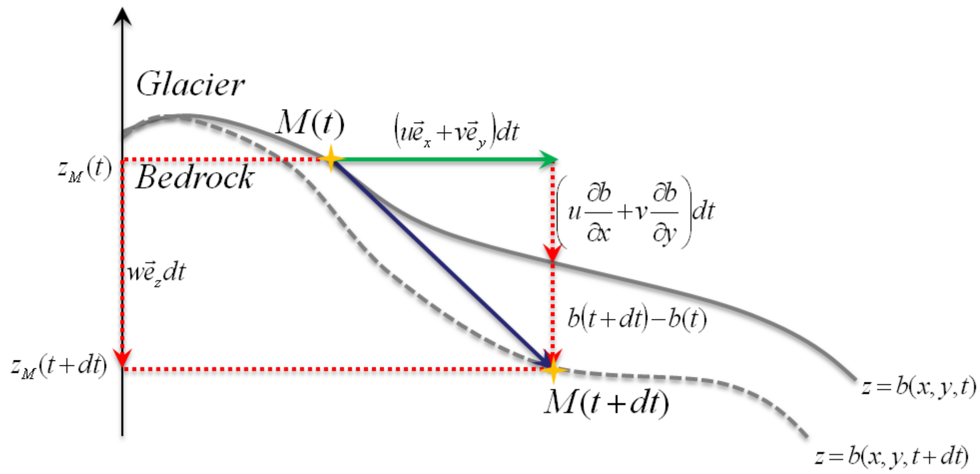


Figure 1.7: vertical displacement of the glacier base

The vertical displacement of M is due to:

- The bedrock motion (isostatic adjustment): $b(t + dt) - b(t)$
- The velocity of the glacier (the glacier is sliding and M follows the bedrock geometry)

Therefore we have the following equation:

$$z_M(t + dt) - z_M(t) = w_M dt = \left(u\frac{\partial b}{\partial x} + v\frac{\partial b}{\partial y}\right) dt + b(t + dt) - b(t) \tag{1.36}$$

And then one must add the melting rate $\dot{M}_b$ at the base:

$$z_M(t + dt) - z_M(t) = w_M dt = \left(u_b\frac{\partial b}{\partial x} + v_b\frac{\partial b}{\partial y}\right) dt + b(t + dt) - b(t) - \dot{M}_b dt \tag{1.37}$$

That gives us a constraint[4] for the vertical component of the glacier velocity at its base[5] w_b:

$$\boxed{w_b = \frac{\partial b}{\partial t} + u_b\frac{\partial b}{\partial x} + v_b\frac{\partial b}{\partial y} - \dot{M}_b} \tag{1.38}$$

[4] usually the term $\frac{\partial b}{\partial t}$ is ignored

[5] The expression of w_b can also be found using the same approach as the ice/water boundary condition. The equation (1.25) is still true without melting or accumulation. Since ice is grounded, the accumulation $\dot{M}_s$ has no effect on w_b, whereas basal melting leads to a thinning of the ice sheet at a velocity equal to $-\dot{M}_b$. If one adds this velocity to equation (1.25), one obtains the exact same equation as demonstrated here

1.5.5 Friction simplifications

1.5.5.1 Development of the boundary condition

We have the following equations:

$$\overrightarrow{\tau_b} = -\alpha^2 u_b = -\alpha^2 \left(\overrightarrow{u} - \left(\overrightarrow{u}.\overrightarrow{n}\right)\overrightarrow{n}\right) = -\alpha^2 \begin{pmatrix} u\left(1-{n_x}^2\right) - vn_xn_y - wn_xn_z \\ -un_xn_y + v\left(1-{n_y}^2\right) - wn_yn_z \\ -un_xn_z - vn_yn_z + w\left(1-{n_z}^2\right) \end{pmatrix} \tag{1.39}$$

Finally, the exact boundary condition:

$$\overrightarrow{F} = \sigma\overrightarrow{n} = \overrightarrow{\tau_b} + \sigma_{nn}\overrightarrow{n} = \begin{pmatrix} -\alpha^2 u\left(1-{n_x}^2\right) + \alpha^2 vn_xn_y + \alpha^2 wn_xn_z + \sigma_{nn}n_x \\ \alpha^2 un_xn_y - \alpha^2 v\left(1-{n_y}^2\right) + \alpha^2 wn_yn_z + \sigma_{nn}n_y \\ \alpha^2 un_xn_z + \alpha^2 vn_yn_z - \alpha^2 w\left(1-{n_z}^2\right) + \sigma_{nn}n_z \end{pmatrix} \tag{1.40}$$

with:

$$\sigma_{nn} \stackrel{def}{=} \sigma\overrightarrow{n}.\overrightarrow{n} = \sigma_{xx}{n_x}^2 + \sigma_{yy}{n_y}^2 + \sigma_{zz}{n_z}^2 + 2\sigma_{xy}n_xn_y + 2\sigma_{xz}n_zn_x + 2\sigma_{yz}n_zn_y \tag{1.41}$$

1.5.5.2 Zero order

We can use a development of near zero order in n_x and n_y condering that the other terms have roughly the same order of magnitude[6]. We consider that the normal vector is such as $n_x = n_y = 0$ and $n_z = -1$. So we have $\sigma_{nn} = \sigma_{zz} + \mathcal{O}(n_x, n_y)$

So the previous force becomes:

$$\overrightarrow{F} = \begin{pmatrix} -\alpha^2 u + \mathcal{O}(n_x, n_y) \\ -\alpha^2 v + \mathcal{O}(n_x, n_y) \\ \sigma_{zz} + \mathcal{O}(n_x, n_y) \end{pmatrix} \tag{1.42}$$

1.5.5.3 First order

This relation doesn't consider the slope of the ice sheet base. That's why we will now use a first order development of the boundary condition to consider this slope. We neglect the terms composed with a product of n_x or n_y. This imposes:

$$\begin{aligned} {n_z}^2 &= 1 - {n_x}^2 - {n_y}^2 = 1 + \mathcal{O}(n_x^2, n_y^2) \\ \sigma_{nn} &= \sigma_{zz} + 2\sigma_{xz}n_zn_x + 2\sigma_{yz}n_zn_y + \mathcal{O}(n_x^2, n_y^2) \end{aligned} \tag{1.43}$$

The boundary condition becomes:

$$\overrightarrow{F} = \begin{pmatrix} -\alpha^2 u + \alpha^2 wn_xn_z + \sigma_{zz}n_x + \mathcal{O}(n_x^2, n_y^2) \\ -\alpha^2 v + \alpha^2 wn_yn_z + \sigma_{zz}n_y + \mathcal{O}(n_x^2, n_y^2) \\ \alpha^2 un_xn_z + \alpha^2 vn_yn_z + \sigma_{zz}n_z + 2\sigma_{xz}n_x + 2\sigma_{yz}n_y + \mathcal{O}(n_x^2, n_y^2) \end{pmatrix} \tag{1.44}$$

[6] An adimensionnalization study not detailed here shows that the results of the following developments are consistent

1.6 Simplified ice flow models

This section presents the three ice flow models that are implemented in *ISSM*, from the most complex to the simplest. They all use the set of equations described in the first part of this chapter (mass balance, momentum equilibrium and constitutive relation) but have different levels of assumptions to simplify this initial set of equations. The idea is that the simpler, the faster in terms of computational time, but the less precise in terms of physics.

1.6.1 Full Stokes Model

The full Stokes model consists in resolving equations (1.15) without approximation. It is a 3d incompressible ice flow model. The pressure P becomes a variable like the velocity (u, v, w). The additional fourth equation comes from incompressibility (1.9). The equations are:

$$\boxed{\begin{array}{c} \left\{ \begin{array}{l} \dfrac{\partial}{\partial x}\left(2\mu\dfrac{\partial u}{\partial x}\right) + \dfrac{\partial}{\partial y}\left(\mu\dfrac{\partial u}{\partial y} + \mu\dfrac{\partial v}{\partial x}\right) + \dfrac{\partial}{\partial z}\left(\mu\dfrac{\partial u}{\partial z} + \mu\dfrac{\partial w}{\partial x}\right) - \dfrac{\partial P}{\partial x} = 0 \\ \dfrac{\partial}{\partial x}\left(\mu\dfrac{\partial u}{\partial y} + \mu\dfrac{\partial v}{\partial x}\right) + \dfrac{\partial}{\partial y}\left(2\mu\dfrac{\partial v}{\partial y}\right) + \dfrac{\partial}{\partial z}\left(\mu\dfrac{\partial v}{\partial z} + \mu\dfrac{\partial w}{\partial y}\right) - \dfrac{\partial P}{\partial y} = 0 \\ \dfrac{\partial}{\partial x}\left(\mu\dfrac{\partial u}{\partial z} + \mu\dfrac{\partial w}{\partial x}\right) + \dfrac{\partial}{\partial y}\left(\mu\dfrac{\partial v}{\partial z} + \mu\dfrac{\partial w}{\partial y}\right) + \dfrac{\partial}{\partial z}\left(2\mu\dfrac{\partial w}{\partial z}\right) - \dfrac{\partial P}{\partial z} - \rho g = 0 \end{array} \right. \\ \\ \dfrac{\partial u}{\partial x} + \dfrac{\partial v}{\partial y} + \dfrac{\partial w}{\partial z} = 0 \end{array}} \tag{1.45}$$

1.6.2 Pattyn's Higher-order 3d Model

Pattyn's model is described in Pattyn's 2003 paper *A new three-dimensional higher-order thermomechanical ice sheet model* [11]. It is a 3d model in which the vertical velocity is deduced from the horizontal components of velocity. The two main assumptions of this model are (1) the horizontal gradients of the vertical velocity are small compared to the vertical gradient of the horizontal velocity, and (2) the hydrostatic approximation in the vertical. This translates to:

- $\dfrac{\partial w}{\partial x} \ll \dfrac{\partial u}{\partial z}$
- $\dfrac{\partial w}{\partial y} \ll \dfrac{\partial v}{\partial z}$
- $\dfrac{\partial \sigma_{xz}}{\partial x} \ll \dfrac{\partial \sigma_{zz}}{\partial z}$
- $\dfrac{\partial \sigma_{yz}}{\partial y} \ll \dfrac{\partial \sigma_{zz}}{\partial z}$

With the second assumption, the third equation of (1.45) is reduced to:

$$\frac{\partial}{\partial z}\left(2\mu\frac{\partial w}{\partial z}\right) - \frac{\partial P}{\partial z} - \rho g = 0 \tag{1.46}$$

If we integrate this equation from z to the surface s, we obtain:

$$P(s) - P(z) = 2\,\mu\frac{\partial w}{\partial z}\bigg|_s - 2\,\mu\frac{\partial w}{\partial z}\bigg|_z - \rho g\,(s - z) \tag{1.47}$$

The boundary conditions at the ice/atmosphere interface (1.19) impose:

$$\sigma_{zz}(s) = 2\,\mu \frac{\partial w}{\partial z}\bigg|_s - P(s) \simeq 0 \tag{1.48}$$

Finally the pressure is:

$$P(z) = 2\mu \frac{\partial w}{\partial z} + \rho g\,(s - z) \tag{1.49}$$

With this approximation, full Stokes equations (1.45) become:

$$\begin{cases} \dfrac{\partial}{\partial x}\left(2\mu\dfrac{\partial u}{\partial x}\right) + \dfrac{\partial}{\partial y}\left(\mu\dfrac{\partial u}{\partial y} + \mu\dfrac{\partial v}{\partial x}\right) + \dfrac{\partial}{\partial z}\left(\mu\dfrac{\partial u}{\partial z} - \mu\dfrac{\partial w}{\partial x}\right) = \rho g\dfrac{\partial s}{\partial x} \\[2ex] \dfrac{\partial}{\partial x}\left(\mu\dfrac{\partial u}{\partial y} + \mu\dfrac{\partial v}{\partial x}\right) + \dfrac{\partial}{\partial y}\left(2\mu\dfrac{\partial v}{\partial y}\right) + \dfrac{\partial}{\partial z}\left(\mu\dfrac{\partial v}{\partial z} - \mu\dfrac{\partial w}{\partial y}\right) = \rho g\dfrac{\partial s}{\partial y} \\[2ex] P(z) = 2\mu\dfrac{\partial w}{\partial z} + \rho g\,(s - z) \end{cases} \tag{1.50}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0$$

The second assumption simplifies the first two equations:

$$\begin{cases} \dfrac{\partial}{\partial x}\left(2\mu\dfrac{\partial u}{\partial x}\right) + \dfrac{\partial}{\partial y}\left(\mu\dfrac{\partial u}{\partial y} + \mu\dfrac{\partial v}{\partial x}\right) + \dfrac{\partial}{\partial z}\left(\mu\dfrac{\partial u}{\partial z}\right) = \rho g\dfrac{\partial s}{\partial x} \\[2ex] \dfrac{\partial}{\partial x}\left(\mu\dfrac{\partial u}{\partial y} + \mu\dfrac{\partial v}{\partial x}\right) + \dfrac{\partial}{\partial y}\left(2\mu\dfrac{\partial v}{\partial y}\right) + \dfrac{\partial}{\partial z}\left(\mu\dfrac{\partial v}{\partial z}\right) = \rho g\dfrac{\partial s}{\partial y} \\[2ex] P(z) = 2\mu\dfrac{\partial w}{\partial z} + \rho g\,(s - z) \end{cases} \tag{1.51}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0$$

The vertical component of the velocity is deduced from the incompressibility equation[7] that is integrated from b to z. The equations of Pattyn's model become :

$$\boxed{\begin{cases} \dfrac{\partial}{\partial x}\left(4\mu\dfrac{\partial u}{\partial x} + 2\mu\dfrac{\partial v}{\partial y}\right) + \dfrac{\partial}{\partial y}\left(\mu\dfrac{\partial u}{\partial y} + \mu\dfrac{\partial v}{\partial x}\right) + \dfrac{\partial}{\partial z}\left(\mu\dfrac{\partial u}{\partial z}\right) = \rho g\dfrac{\partial s}{\partial x} \\[2ex] \dfrac{\partial}{\partial x}\left(\mu\dfrac{\partial u}{\partial y} + \mu\dfrac{\partial v}{\partial x}\right) + \dfrac{\partial}{\partial y}\left(4\mu\dfrac{\partial v}{\partial y} + 2\mu\dfrac{\partial u}{\partial x}\right) + \dfrac{\partial}{\partial z}\left(\mu\dfrac{\partial v}{\partial z}\right) = \rho g\dfrac{\partial s}{\partial y} \\[2ex] w\,(x,y,z) = w_b\,(x,y,z) - \displaystyle\int_{b(x,y)}^{z} \dfrac{\partial u}{\partial x} + \dfrac{\partial v}{\partial y} dz' \end{cases}} \tag{1.52}$$

[7]For ice shelves, we will assume hydrostatic equilibrium so that we can use a Dirichlet boundary condition at the ice shelf base for the vertical velocity computation (See section 1.5.3)

1.6.3 MacAyeal's Shelfy-stream 2d model

MacAyeal's model is described in Douglas MacAyeal's 1989 paper *Large-scale Ice Flow Over a Viscous Basal Sediment* [7]. It is a 2d model, the horizontal components of the velocity are vertically integrated. The main assumption is that the basal drag associated with deforming basal sediment does not induce significant vertical gradients of the horizontal velocity. This drove him to make the following assumptions:

- $\dfrac{\partial u}{\partial z} = 0$
- $\dfrac{\partial v}{\partial z} = 0$
- $\dot{\varepsilon}_{xz} = 0$
- $\dot{\varepsilon}_{yz} = 0$
- bedrock slope assumed to be negligible

These assumptions are stronger than Pattyn's. Therefore, we can use Pattyn's higher order model equations (1.52) as a starting point. If we use the first two assumptions, the first equations of Pattyn's model become:

$$\left\{ \begin{aligned} &\frac{\partial}{\partial x}\left(4\mu\frac{\partial u}{\partial x} + 2\mu\frac{\partial v}{\partial y}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right) = \rho g\frac{\partial s}{\partial x} \\ &\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial y}\left(4\mu\frac{\partial v}{\partial y} + 2\mu\frac{\partial u}{\partial x}\right) = \rho g\frac{\partial s}{\partial y} \end{aligned} \right. \tag{1.53}$$

One can integrate these equations from the bed $b(x,y)$ to the surface $s(x,y)$:

$$\left\{ \begin{aligned} &\int_{b(x,y)}^{s(x,y)} \left(\frac{\partial}{\partial x}\left(4\mu\frac{\partial u}{\partial x} + 2\mu\frac{\partial v}{\partial y}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial z}\right)\right) dz = \rho g H\frac{\partial s}{\partial x} \\ &\int_{b(x,y)}^{s(x,y)} \left(\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial y}\left(4\mu\frac{\partial v}{\partial y} + 2\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial z}\left(\mu\frac{\partial v}{\partial z}\right)\right) dz = \rho g H\frac{\partial s}{\partial y} \end{aligned} \right. \tag{1.54}$$

We can use Leibniz integral rule[8] which gives:

$$\begin{aligned} &\frac{\partial}{\partial x}\int_{b(x,y)}^{s(x,y)} \left(4\mu\frac{\partial u}{\partial x} + 2\mu\frac{\partial v}{\partial y}\right) dz + \frac{\partial}{\partial y}\int_{b(x,y)}^{s(x,y)} \left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right) dz \\ &\qquad - \sigma'_{xx}(s)\frac{\partial s}{\partial x} - \sigma'_{xy}(s)\frac{\partial s}{\partial x} + \sigma'_{xz}(s) + \sigma'_{xx}(b)\frac{\partial b}{\partial x} + \sigma'_{xy}(b)\frac{\partial b}{\partial x} - \sigma'_{xz}(b) = \rho g H\frac{\partial s}{\partial x} \end{aligned} \tag{1.55}$$

$$\begin{aligned} &\frac{\partial}{\partial x}\int_{b(x,y)}^{s(x,y)} \left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right) dz + \frac{\partial}{\partial y}\int_{b(x,y)}^{s(x,y)} \left(4\mu\frac{\partial v}{\partial y} + 2\mu\frac{\partial u}{\partial x}\right) dz \\ &\qquad - \sigma'_{yx}(s)\frac{\partial s}{\partial x} - \sigma'_{yy}(s)\frac{\partial s}{\partial x} + \sigma'_{yz}(s) + \sigma'_{yx}(b)\frac{\partial b}{\partial x} + \sigma'_{yy}(b)\frac{\partial b}{\partial x} - \sigma'_{yz}(b) = \rho g H\frac{\partial s}{\partial y} \end{aligned} \tag{1.56}$$

[8] For a given scalar quantity $f(x,z)$, Leibniz integral rule is written:

$$\frac{\partial}{\partial x}\int_{\alpha(x)}^{\beta(x)} f(x,z)dz = \int_{\alpha(x)}^{\beta(x)} \frac{\partial}{\partial x} f(x,z)dz + f(x,\beta(x))\frac{\partial \beta(x)}{\partial x} - f(x,\alpha(x))\frac{\partial \alpha(x)}{\partial x}$$

In our case, we use:

$$\int_{\alpha(x)}^{\beta(x)} \frac{\partial}{\partial x} f(x,z)dz = \frac{\partial}{\partial x}\int_{\alpha(x)}^{\beta(x)} f(x,z)dz - f(x,\beta(x))\frac{\partial \beta(x)}{\partial x} + f(x,\alpha(x))\frac{\partial \alpha(x)}{\partial x}$$

The boundary conditions at the ice/atmosphere interface (1.19) imposes:

$$\sigma'(s)\overrightarrow{n} = \overrightarrow{0} \tag{1.57}$$

To the zero order (horizontal bedrock), the boundary conditions at the ice/till interface (1.42) is:

$$\begin{aligned} (\sigma'(b)\overrightarrow{n})_x &= \alpha^2 u \\ (\sigma'(b)\overrightarrow{n})_y &= \alpha^2 v \end{aligned} \tag{1.58}$$

Equation (1.55) and (1.56) become:

$$\left\{ \begin{aligned} &\frac{\partial}{\partial x}\int_{b(x,y)}^{s(x,y)}\left(4\mu\frac{\partial u}{\partial x}+2\mu\frac{\partial v}{\partial y}\right)dz+\frac{\partial}{\partial y}\int_{b(x,y)}^{s(x,y)}\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)dz=\rho g H\frac{\partial s}{\partial x}+\alpha^2 u \\ &\frac{\partial}{\partial x}\int_{b(x,y)}^{s(x,y)}\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)dz+\frac{\partial}{\partial y}\int_{b(x,y)}^{s(x,y)}\left(4\mu\frac{\partial v}{\partial y}+2\mu\frac{\partial u}{\partial x}\right)dz=\rho g H\frac{\partial s}{\partial y}+\alpha^2 v \end{aligned} \right. \tag{1.59}$$

MacAyeal's assumptions imply that velocities (u and v) do not depend on z: there is no need to use Leibniz integral law here. We define the depth averaged viscosity as: $H\overline{\mu} = \int_b^s \mu dz$ so that finally we obtain:

$$\boxed{\left\{ \begin{aligned} &\frac{\partial}{\partial x}\left(4H\bar{\mu}\frac{\partial u}{\partial x}+2H\bar{\mu}\frac{\partial v}{\partial y}\right)+\frac{\partial}{\partial y}\left(H\bar{\mu}\frac{\partial u}{\partial y}+H\bar{\mu}\frac{\partial v}{\partial x}\right)=\rho g H\frac{\partial s}{\partial x}+\alpha^2 u \\ &\frac{\partial}{\partial y}\left(4H\bar{\mu}\frac{\partial v}{\partial y}+2H\bar{\mu}\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial x}\left(H\bar{\mu}\frac{\partial u}{\partial y}+H\bar{\mu}\frac{\partial v}{\partial x}\right)=\rho g H\frac{\partial s}{\partial y}+\alpha^2 v \end{aligned} \right.} \tag{1.60}$$

1.7 Thermodynamic model

A thermal model is essential to predict an ice sheet system's evolution since it affects the flow. The viscosity parameter B is highly temperature-dependent. Temperature also affects melting, which in turn affects the glacier geometry and sliding. It is hence needed in a comprehensive time evolution model.

1.7.1 Local equation

The most general energy balance (Hooke, 2005 [4] p117, Paterson, 1994 [10] p224) equation includes a conduction term and a source term (here due to viscous heating):

$$\frac{d\left(\rho c T\right)}{dt} = div\left(k\,\overrightarrow{grad}\,T\right) + \Phi \tag{1.61}$$

where:

- c ice heat capacity ($Jkg^{-1}K^{-1}$)
- T ice temperature (K)
- k ice heat conductivity ($Wm^{-1}K^{-1}$)
- Φ viscous heating due to ice flow (Wm^{-3})

Advection appears when one takes the Eulerian version of the previous equations:

$$\frac{d\left(\rho c T\right)}{dt} = \frac{\partial\left(\rho c T\right)}{\partial t} + \overrightarrow{v}.\overrightarrow{grad}\left(\rho c T\right) \tag{1.62}$$

this gives:

$$\frac{\partial\left(\rho c T\right)}{\partial t} = -\overrightarrow{v}.\overrightarrow{grad}\left(\rho c T\right) + div\left(k\,\overrightarrow{grad}\,T\right) + \Phi \tag{1.63}$$

1.7.2 Simplifications

In $ISSM$, several assumptions are employed. The first one is that (1) the heat capacity and conductivity are constant, and (2) spatial and thermal dependence of k and c are neglected. These are classical assumptions (Hooke, 2005 [4] p117). Using the ice incompressibility, equation (1.63) becomes:

$$\frac{\partial T}{\partial t} = -\overrightarrow{v}.\overrightarrow{grad}\left(T\right) + \frac{k}{\rho c}\Delta T + \frac{\Phi}{\rho c} \tag{1.64}$$

The local heat transfer is thus a result of horizontal and vertical advection, conduction[9], and internal deformation heating:

$$\boxed{\frac{\partial T}{\partial t} = -\left(u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y} + w\frac{\partial T}{\partial z}\right) + \frac{k}{\rho c}\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2}\right) + \frac{\Phi}{\rho c}} \tag{1.65}$$

1.7.3 Internal deformation heating

The viscous heating due to ice flow is

$$\Phi = Tr\left(\sigma\dot{\varepsilon}\right) \tag{1.66}$$

One can use the deviatoric stress tensor and use the incompressibility:

[9] In many thermal models, the horizontal advection is neglected (Paterson, 1994 [10] p216; Hooke, 2005 [4] p118). Horizontal conduction is often neglected as well (Hulbe, 1999 [5]; Hooke, 2005 [4] p118). These assumptions are not employed here.

$$\Phi = Tr\left(\sigma'\dot{\varepsilon} - P\dot{\varepsilon}\right) = Tr\left(\sigma'\dot{\varepsilon}\right) - P\,Tr\left(\dot{\varepsilon}\right) = Tr\left(\sigma'\dot{\varepsilon}\right) \tag{1.67}$$

We do not use this equation in $ISSM$. We use the effective strain rate $\dot{\varepsilon}_e$ and effective deviatoric stress σ_e defined in (1.2) to obtain:

$$\begin{aligned}\Phi^2 &= \left(Tr\left(\sigma'\dot{\varepsilon}\right)\right)^2 = \left(\sum_{i,j}\sigma'_{ij}\dot{\varepsilon}_{ji}\right)^2 = \left(\sum_{i,j}\sigma'_{ij}\dot{\varepsilon}_{ij}\right)^2 = \left(\sum_{i,j}\sigma'_{ij}\dot{\varepsilon}_{ij}\right)\left(\sum_{k,l}\sigma'_{kl}\dot{\varepsilon}_{kl}\right)\\ &= \left(\sum_{i,j}2\mu\dot{\varepsilon}_{ij}\dot{\varepsilon}_{ij}\right)\left(\sum_{k,l}\sigma'_{kl}\frac{1}{2\mu}\sigma'_{kl}\right) = \sum_{i,j}\dot{\varepsilon}_{ij}^2\sum_{kl}{\sigma'_{kl}}^2 = 2\dot{\varepsilon}_e^2\;\;2{\sigma'_e}^2\end{aligned} \tag{1.68}$$

In conclusion, $ISSM$ uses the following equation:

$$\Phi = 2\dot{\varepsilon}_e\sigma'_e = 4\mu\dot{\varepsilon}_e^2 \tag{1.69}$$

1.7.4 Thermal model boundary conditions

Ice/Atmosphere interface

On the upper surface, the temperature is imposed as the mean annual air temperature at the surface, T_s.

Ice/Water boundary

On the Ice/Water interface, there is an imposed flux, which depends on the temperature difference between the ice shelf surface T_b and the ocean $T = T_{pmp}$:

$$k\left(\overrightarrow{grad}\,T\right)\Big|_b.\overrightarrow{n} \simeq -k\left.\frac{\partial T}{\partial z}\right|_b = -\rho_w c_{pM}\gamma\left(T_b - T_{pmp}\right) \tag{1.70}$$

With:

- $\overrightarrow{n} = (n_x, n_y, n_z)$ normal vector pointing outward
- c_{pM} mixed layer (see Holland, 1999 [3] p5) specific heat capacity $(Jkg^{-1}K^{-1})$
- γ thermal exchange velocity (ms^{-1})
- $T_{pmp} = 273.15 - \beta P$ pressure melting point (K)(See Paterson [10] p212)

If the ocean temperature is above the ice shelf base temperature, the flux is positive (heat goes from the ocean to the ice shelf), whereas when the ice shelf base is warmer than the ocean, the flux is negative (heat goes from the ice shelf to the ocean).

Ice/Till boundary

The basal thermal boundary condition is supplied by a geothermal flux G at the ice sheet bottom and the heat due to basal friction. When the pressure melting point is reached, an additional flux is generated:

$$k\left(\overrightarrow{grad}\,T\right)\Big|_b.\overrightarrow{n} \simeq \underbrace{-k\left.\frac{\partial T}{\partial z}\right|_b}_{Conduction} = \underbrace{G + \overrightarrow{\tau}_b.\overrightarrow{u}_b}_{Energy\ input} + \underbrace{\rho L\dot{M}_b}_{Melting} \tag{1.71}$$

Where $\overrightarrow{n} = (n_x, n_y, n_z)$ is the unit normal vector pointing outward and L is the specific latent heat of fusion. The basal temperature in the ice mass is kept at the pressure melting point $T_{pmp} = 273.15 - \beta P$ (See Paterson [10] p212) whenever it is reached (Pattyn, 2003 [11]). In other words, the boundary condition at the ice/till interface is an imposed flux. When this imposed flux generates ice temperatures above the pressure melting point, the boundary condition is changed into an imposed temperature $T_b = T_{pmp}$.

1.7.5 Transformation of the basal boundary condition

At the base, we impose a heat flux, and the temperature cannot exceed T_{pmp}. If the temperature reaches the pressure melting point, melting occurs. All these conditions can be summarized by the following equations:

$$\begin{cases} -k \left.\dfrac{\partial T}{\partial z}\right|_b = G + \overrightarrow{\tau}_b . \overrightarrow{u}_b - \rho L \dot{M}_b \\ \dot{M}_b \left(T - T_{pmp}\right) = 0 \\ T - T_{pmp} \leq 0 \\ \dot{M}_b \geq 0 \end{cases} \tag{1.72}$$

The last three equations can be represented by the following graph:

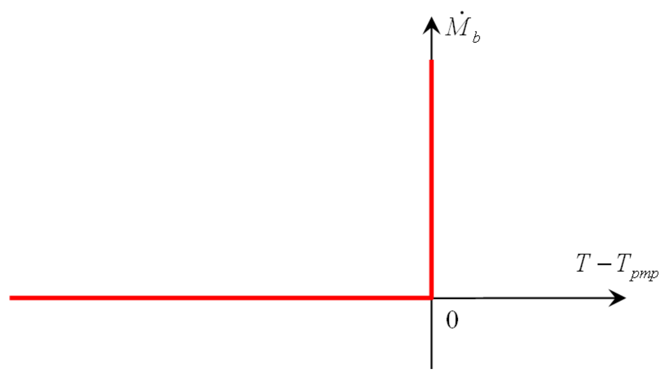


Figure 1.8: $\dot{M}_b$ as a function of $T - T_{pmp}$

These equations are not used in $ISSM$ since they are introducing nonlinearities. To avoid that, we use a penalty method exactly as in contact mechanics (See Chapter 5 for more details).

Conclusion

We used the mass conservation, the momentum balance and Glen's flow law to deduce the system of equations that is governing ice flow. This system of equations constitutes the full Stokes equations. This model unfortunately requires huge computational resources in terms of computer memory and execution time: it is a 3d model that solves 4 unknowns (u, v, w, P). We also presented two other models: MacAyeal's shelfy-stream model and Frank Pattyn's higher-order model. Frank Pattyn uses still a 3d mesh but decorellates the horizontal velocity components from the vertical component. There are hence only 2 unknowns (u, v). Doug MacAyeal went further by vertically integrating the equations, neglecting vertical shear. These three models give three different refinement levels and require three different computational powers, which is priceless to model a continent such as Antarctica since one can use different models on different areas depending on the accuracy needed and the phenomena involved.

Chapter 2

Dynamic Models Finite Element Formulation

ISSM uses finite element method to solve the equations described in the first chapter. This chapter explains how we built a finite element model for each solution. MacAyeal's model formulation is very detailed for those who are not very familiar with finite element method. For more details about finite element method, the reader is invited to consult Zienkiewicz *The Finite Element Method, Fourth Edition*, 1994 [13].

2.1 MacAyeal's Shelfy-stream model

2.1.1 Geometry and notations

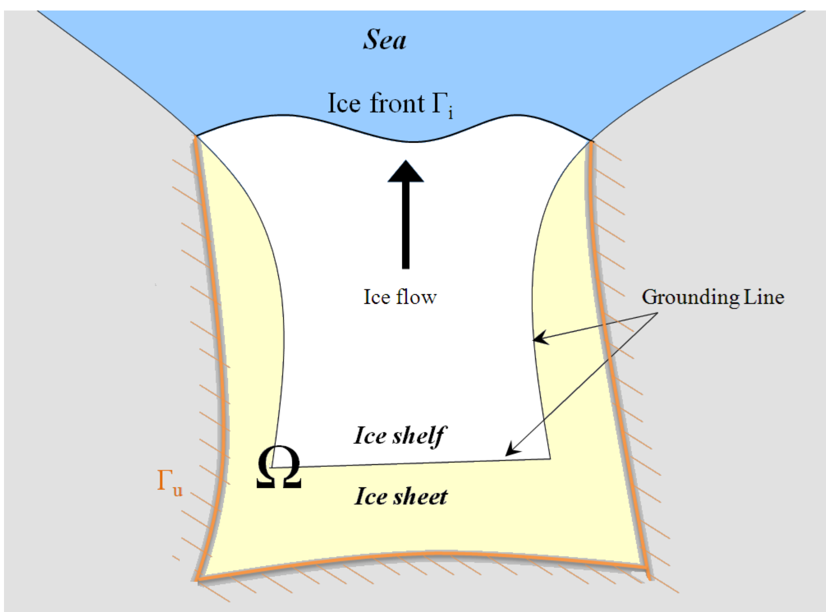


Figure 2.1: Geometry and notations

We adopt the following notations:

- Ω is the open surface that constitutes the entire glacier system
- $\partial\Omega$ are points located at the boundary
- Γ are points located at the ice front

– $\Gamma_u = \partial\Omega \backslash \Gamma$ are all other points located at the boundary

2.1.2 Equations

The equations of MacAyeal's Shelfy-stream model are:

$$\left\{ \begin{array}{l} \dfrac{\partial}{\partial x}\left(2H\mu\left(2\dfrac{\partial u}{\partial x}+\dfrac{\partial v}{\partial y}\right)\right)+\dfrac{\partial}{\partial y}\left(H\mu\left(\dfrac{\partial u}{\partial y}+\dfrac{\partial v}{\partial x}\right)\right)-\rho g H\dfrac{\partial s}{\partial x}+\tau_x=0 \\ \\ \dfrac{\partial}{\partial y}\left(2H\mu\left(2\dfrac{\partial v}{\partial y}+\dfrac{\partial u}{\partial x}\right)\right)+\dfrac{\partial}{\partial x}\left(H\mu\left(\dfrac{\partial u}{\partial y}+\dfrac{\partial v}{\partial x}\right)\right)-\rho g H\dfrac{\partial s}{\partial y}+\tau_y=0 \end{array} \right. \tag{2.1}$$

At the ice front, the boundary condition is (1.21):

$$\sigma'\overrightarrow{n} = (P - P_w)\,\overrightarrow{n} \tag{2.2}$$

where:

- $P = \rho g\,(s - z)$ is the ice pressure
- $P = -\rho_w g z$ is the water pressure ($z \leq 0$)

MacAyeal's model uses vericaly integrated quantities, we must apply the verticaly-integrated form of the previous equation which is:

$$\sigma'\overrightarrow{n} = \left(\int_b^s \rho g\,(s-z)\,dz + \int_b^0 \rho_w g z dz\right)\overrightarrow{n} = \left(\frac{1}{2}\rho g H^2 - \frac{1}{2}\rho_w g b^2\right)\overrightarrow{n} \tag{2.3}$$

The boundary conditions are: a dynamic boundary condition (Neumann) at the ice front, a friction law on the ice sheet, and single point constraint (Dirichlet) on the other borders of the domain Ω. It is more convenient to consider these specified velocities as nill. At the end of the process, the velocity field will be updated to take these constraints into account.

$$\left\{ \begin{array}{ll} \text{Ice Front } \Gamma_i: & \sigma'\overrightarrow{n} = F\overrightarrow{n} = \left(\dfrac{1}{2}\rho g H^2 - \dfrac{1}{2}\rho_w g b^2\right)\overrightarrow{n} \\ \\ \text{Other borders } \Gamma_u: & u = v = 0 \end{array} \right. \tag{2.4}$$

2.1.3 Weak Formulation

Now that we have all the equations and boundary conditions, we write the weak formulation to solve these equations using finite element. We define a kinematically admissible velocity field[1][2] $\overrightarrow{\phi}$:

$$\overrightarrow{\phi}(x,y) \in \{\left(H^1(\Omega \cup \partial\Omega)\right)^2 \quad \backslash \quad \overrightarrow{\phi}(x,y) = \overrightarrow{0} \qquad (x,y) \in \Gamma_u\} \tag{2.6}$$

For any function of this virtual field, we take the scalar product of the equations (2.1) with $\overrightarrow{\phi} = (\phi_x, 0)$ and $\overrightarrow{\phi} = (0, \phi_y)$ that are 2 other kinematically admissible velocity fields, and integrate these products over the domain. Indeed, if $\overrightarrow{\phi}$ is a kinematically admissible velocity field, then $\overrightarrow{\phi_1} = (\phi_x, 0)$ and $\overrightarrow{\phi_2} = (0, \phi_y)$ are also two kinematically admissible velocity fields. This gives us:

$$\iint_\Omega \left[\frac{\partial}{\partial x}\left(2H\mu\left(2\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)\right)\phi_x + \frac{\partial}{\partial y}\left(H\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right)\phi_x - \rho g H\frac{\partial s}{\partial x}\phi_x + \tau_x\phi_x\right] d\Omega = 0$$

$$\iint_\Omega \left[\frac{\partial}{\partial y}\left(2H\mu\left(2\frac{\partial v}{\partial y} + \frac{\partial u}{\partial x}\right)\right)\phi_y + \frac{\partial}{\partial x}\left(H\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right)\phi_y - \rho g H\frac{\partial s}{\partial y}\phi_y + \tau_y\phi_y\right] d\Omega = 0$$

One intergrates by parts the two first terms in order to merge in the force on the ice front Γ:

$$\begin{aligned}\iint_\Omega 2H\mu\left(2\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)\frac{\partial \phi_x}{\partial x} + H\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\frac{\partial \phi_x}{\partial y} - \tau_x\phi_x d\Omega \\ = \int_\Gamma F n_x \phi_x d\Gamma - \iint_\Omega \rho g H \frac{\partial s}{\partial x}\phi_x d\Omega\end{aligned} \tag{2.7}$$

$$\begin{aligned}\iint_\Omega 2H\mu\left(2\frac{\partial v}{\partial y} + \frac{\partial u}{\partial x}\right)\frac{\partial \phi_y}{\partial x} + H\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\frac{\partial \phi_y}{\partial y} - \tau_y\phi_y d\Omega \\ = \int_\Gamma F n_y \phi_y d\Gamma - \iint_\Omega \rho g H \frac{\partial s}{\partial y}\phi_y d\Omega\end{aligned} \tag{2.8}$$

2.1.4 Finite element discretisation, Galerkin method

The domain is meshed using triangles. There are a *nel* elements and *nods* grids. We use the nodal functions of each grid, which are continuous on the domain and affine on each element. We now project the unknows on the nodal functions:

$$\begin{gathered}(\phi_i)_{1 \le i \le nods} \\ u(x,y) = \sum_{i=1}^{nods} u_i \phi_i(x,y) \\ v(x,y) = \sum_{i=1}^{nods} v_i \phi_i(x,y)\end{gathered} \tag{2.9}$$

We build a vector U that holds all the unknowns of the velocity field:

[1] $H^1(\Omega \cup \partial\Omega)$ is a Sobolev space that contains all the square-integrable functions defined on $\Omega \cup \partial\Omega$ that have a square-integrable first derivative

$$\int_{\Omega\cup\partial\Omega} \phi^2 d\Omega < \infty \text{ and } \int_{\Omega\cup\partial\Omega} \left(\frac{\partial \phi}{\partial x_i}\right)^2 d\Omega < \infty \tag{2.5}$$

[2] $\overrightarrow{\phi}$ is a 2d function and each of its component (ϕ_x, ϕ_y) is in $H^1(\Omega \cup \partial\Omega)$. Therefore $\overrightarrow{\phi} \in \left(H^1(\Omega \cup \partial\Omega)\right)^2$

$$U = \begin{bmatrix} u_1 \\ v_1 \\ ... \\ u_{nods} \\ v_{nods} \end{bmatrix} \tag{2.10}$$

With this new field and with $\tau_x = -\alpha^2 u$, $\tau_y = -\alpha^2 v$, we have:

$$\forall 1 \leq i \leq nods$$

$$\sum_{j=1}^{nods} \iint_{\Omega} 2H\mu \left(2u_j \frac{\partial \phi_j}{\partial x} + v_j \frac{\partial \phi_j}{\partial y}\right) \frac{\partial \phi_i}{\partial x} + H\mu \left(u_j \frac{\partial \phi_j}{\partial y} + v_j \frac{\partial \phi_j}{\partial x}\right) \frac{\partial \phi_i}{\partial y} + \alpha^2 u_j \phi_j \phi_i d\Omega$$
$$= \int_{\Gamma} F n_x \phi_i d\Gamma - \iint_{\Omega} \rho g H \frac{\partial s}{\partial x} \phi_i d\Omega \tag{2.11}$$

$$\sum_{j=1}^{nods} \iint_{\Omega} 2H\mu \left(2v_j \frac{\partial \phi_j}{\partial y} + u_j \frac{\partial \phi_j}{\partial x}\right) \frac{\partial \phi_i}{\partial y} + H\mu \left(u_j \frac{\partial \phi_j}{\partial y} + v_j \frac{\partial \phi_j}{\partial x}\right) \frac{\partial \phi_i}{\partial x} + \alpha^2 v_j \phi_j \phi_i d\Omega$$
$$= \int_{\Gamma} F n_y \phi_i d\Gamma - \iint_{\Omega} \rho g H \frac{\partial s}{\partial y} \phi_i d\Omega \tag{2.12}$$

By rearranging terms:

$$\forall 1 \leq i \leq nods$$

$$\sum_{j=1}^{nods} \iint_{\Omega} \left(4H\mu \frac{\partial \phi_j}{\partial r} \frac{\partial \phi_i}{\partial r} + H\mu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial y} + \alpha^2 \phi_j \phi_i\right) u_j + \left(2H\mu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial x} + H\mu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial y}\right) v_j d\Omega$$
$$= \int_{\Gamma} F n_x \phi_i d\Gamma - \iint_{\Omega} \rho g H \frac{\partial s}{\partial x} \phi_i d\Omega \tag{2.13}$$

$$\sum_{j=1}^{nods} \iint_{\Omega} \left(2H\mu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial y} + H\mu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial x}\right) u_j + \left(4H\mu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial y} + H\mu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial x} + \alpha^2 \phi_j \phi_i\right) v_j d\Omega$$
$$= \int_{\Gamma} F n_y \phi_i d\Gamma - \iint_{\Omega} \rho g H \frac{\partial s}{\partial y} \phi_i d\Omega \tag{2.14}$$

We see the stiffness matrix K and load vector F appearing, so that the previous equation can be written in matrix form:

$$[K]\, U = F \tag{2.15}$$

or

$$\begin{bmatrix} & \vdots & \vdots & \\ \cdots & K_{2i-1,2j-1} & K_{2i-1,2j} & \cdots \\ \cdots & K_{2i,2j-1} & K_{2i,2j} & \cdots \\ & \vdots & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ u_j \\ v_j \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ F_{2j-1} \\ F_{2j} \\ \vdots \end{bmatrix} \tag{2.16}$$

We have the following expressions:

$$
\begin{aligned}
K_{2i-1,2j-1} &= \iint_\Omega \left(4H\mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x} + H\mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y} + \alpha^2 \phi_j \phi_i \right) d\Omega \\
K_{2i-1,2j} &= \iint_\Omega \left(2H\mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial x} + H\mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial y} \right) d\Omega \\
K_{2i,2j-1} &= \iint_\Omega \left(2H\mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial y} + H\mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial x} \right) d\Omega \\
K_{2i,2j} &= \iint_\Omega \left(4H\mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y} + H\mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x} + \alpha^2 \phi_j \phi_i \right) d\Omega \\
F_{2i-1} &= \int_\Gamma F n_x \phi_i d\Gamma - \iint_\Omega \rho g H \frac{\partial s}{\partial x} \phi_i d\Omega \\
F_{2i} &= \int_\Gamma F n_y \phi_i d\Gamma - \iint_\Omega \rho g H \frac{\partial s}{\partial y} \phi_i d\Omega
\end{aligned}
\tag{2.17}
$$

2.1.5 Decomposition over the elements

The integrals over Ω can be divided in several integrals over each elements. Remembering that the mesh contains *nel* elements,equations (2.13) and (2.14) become:

$$\forall 1 \leq i \leq nods$$

$$\sum_{E=1}^{nel}\left(\sum_{j=1}^{nods}\iint_E\left(4H\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}+H\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}+\alpha^2\phi_j\phi_i\right)u_j+\left(2H\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x}+H\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}\right)v_j dS\right)$$
$$=\sum_{E=1}^{nel}\left(\int_{E\cup\Gamma}Fn_x\phi_i dl-\iint_E\rho gH\frac{\partial s}{\partial x}\phi_i dS\right) \quad (2.18)$$

$$\sum_{E=1}^{nel}\left(\sum_{j=1}^{nods}\iint_E\left(2H\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}+H\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x}\right)u_j+\left(4H\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}+H\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}+\alpha^2\phi_j\phi_i\right)v_j dS\right)$$
$$=\sum_{E=1}^{nel}\left(\int_{E\cup\Gamma}Fn_y\phi_i dl-\iint_E\rho gH\frac{\partial s}{\partial y}\phi_i dS\right) \quad (2.19)$$

To compute the matrices K and F, instead of calculating the integral over Ω directly, we will first integrate over each element E of the mesh to obtain several *Elementary Matrices*, that will be assembled to get the final Stiffness matrix and load vector during the assembly process.

The nodal functions are built so that $\forall 1 \leq i \leq nods$:

$$\phi_i(N_j) = \delta_{ij} \quad (2.20)$$

where δ_{ij} is the Kronecker symbol. The nodal function is equal to 1 at a node and 0 on every other nodes. So for one element E, the only nodal functions that are useful in (2.7) and (2.8) (ie: do not give 0=0) are the nodal functions corresponding to the vertexes of the element. Since the elements are triangle, there are only 3 nodal functions that need to be taken into account.

One element E gives 6 equations: we take 3 nodal functions for $\phi_i(x, y)$ (the nodal functions corresponding to the vertexes of E), and each ϕ_i gives 2 equations (one for the x-component, and one for the y-component). Therefore, an elementary stiffness matrix K_E will be 6×6, and the elementary load vector will be of size 6.

2.1.6 Reference element and numerical integration

Rather than calculating these integrals for each element, one uses a *Reference Element* $\hat{E}$, and a transformation φ that transforms the reference element to the element E (See Fig.2.2):

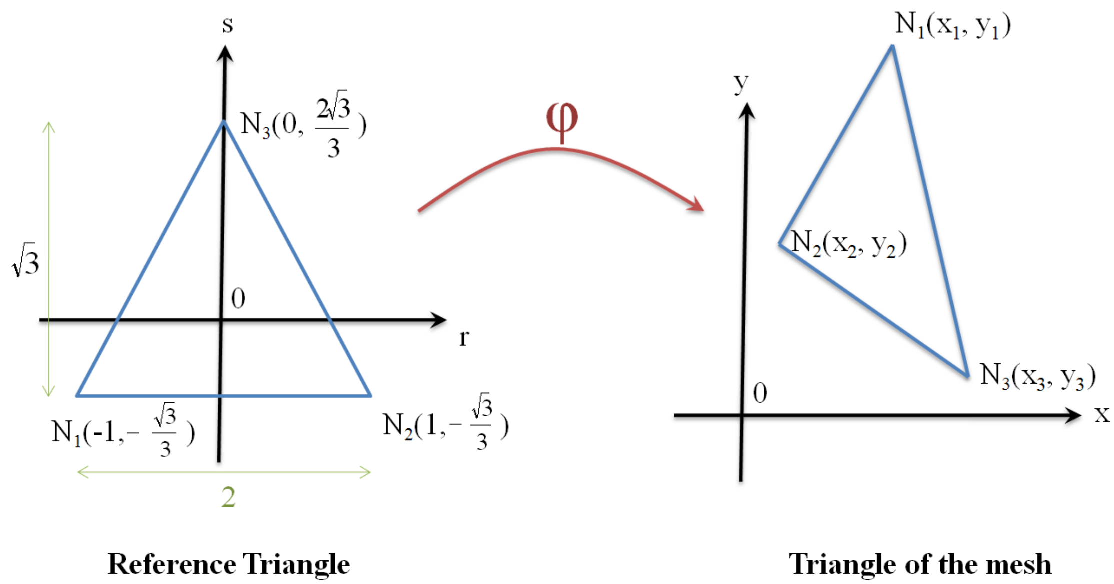


Figure 2.2: Transformation from the reference triangle, to the current element of the mesh

For any function $f(x, y)$, the transformation verifies:

$$\iint_E f(x,y)dS = \iint_{\hat{E}} \hat{f}(r,s)|J|d\hat{S} \tag{2.21}$$

Where J is the Jacobian determinant of the φ. This Jacobian is calculated in section A.2 and is constant over the element:

$$J = \begin{bmatrix} \frac{1}{2}(x_2 - x_1) & \frac{\sqrt{3}}{6}(2x_3 - x_1 - x_2) \\ \frac{1}{2}(y_2 - y_1) & \frac{\sqrt{3}}{6}(2y_3 - y_1 - y_2) \end{bmatrix} \tag{2.22}$$

Since $ISSM$ is a Matlab code, the integrals cannot be calculated directly. The numerical integration use a gaussian quadrature to approximate the integral as a weighted sum of function values at specified points (gaussian points) within the domain:

$$\iint_E f(x,y)dS = \iint_{\hat{E}} \hat{f}(r,s)|J|d\hat{S} \simeq \sum_{g=1}^{n} w_g \hat{f}(r_g, s_g) \tag{2.23}$$

where w_g are the quadrature rules weights and (r_g, s_g) are the evaluation points, $g = 1, 2..., n$. The gauss quadrature is such that it makes the computed integral exact for all polynomials of a certain degree. All the integrals that appear in the stiffness matrix and load vector involve polynomials since nodal function are polynomials of degree 1. It is then easy to evaluate the degree of the polynomial to integrate (usually 2), and then find the gauss quadrature that gives an exact computed integral.

The integration points are given with their area coordinates (See Fig.2.3):

$$(r_g, s_g) \equiv \left(\frac{A_1}{A_1 + A_2 + A_3}, \frac{A_2}{A_1 + A_2 + A_3}, \frac{A_3}{A_1 + A_2 + A_3}\right) \tag{2.24}$$

We use this system of coordinates because they have the following property:

$$1 \leq k \leq 3 \qquad \frac{A_k}{A_1 + A_2 + A_3} = L_k(r_i, s_i) \tag{2.25}$$

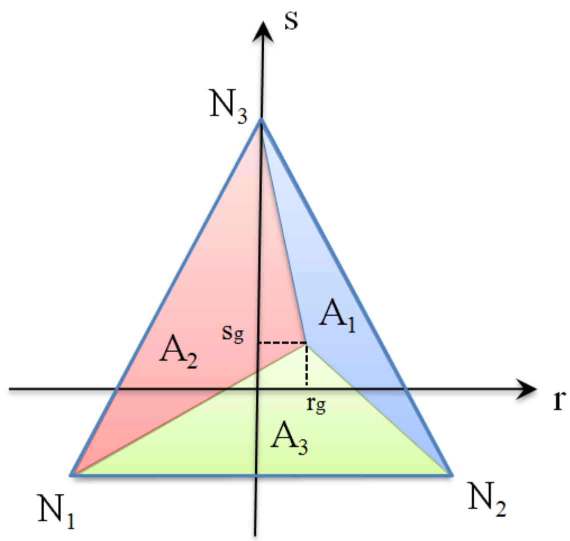


Figure 2.3: Area coordinates

The number of gaussian points depends on the order of integration. The order of integration can be computed from the polynomial degree p that needs to be integrated. The formula is:

$$order = \frac{p+1}{2} \tag{2.26}$$

Here, the degree of the polynome under the integral is 2 (the drag implies the calculation of $\phi_j \phi_i$ whose degree is 2). The order of integration is hence 2 (the order must be an integer). For this order, 3 gaussian points are required:

Gaussian point	1	2	3
Weight	0.577	0.577	0.577
Area coordinates	$\frac{2}{3}$ $\frac{1}{6}$ $\frac{1}{6}$	$\frac{1}{6}$ $\frac{2}{3}$ $\frac{1}{6}$	$\frac{1}{6}$ $\frac{1}{6}$ $\frac{2}{3}$

2.1.7 Elementary stiffness matrix

To compute the elementary stiffness matrix, we first remove the basal drag, that will create a first elementary stiffness matrix $K1$, the basal drag will constitute another stiffness matrix $K2$. The elementary stiffness matrix will be the sum of $K1$ and $K2$ ($K_E = K1 + K2$).

2.1.7.1 Without basal drag

This matrix has the following shape:

$$1 \leq i \leq 3 \text{ and } 1 \leq j \leq 3$$

$$\begin{aligned}
K1_{2i-1,2j-1} &= \iint_E \left(4H\mu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial x} + H\mu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial y} \right) d\Omega \\
K1_{2i-1,2j} &= \iint_E \left(2H\mu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial x} + H\mu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial y} \right) d\Omega \\
K1_{2i,2j-1} &= \iint_E \left(2H\mu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial y} + H\mu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial x} \right) d\Omega \\
K1_{2i,2j} &= \iint_E \left(4H\mu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial y} + H\mu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial x} \right) d\Omega
\end{aligned} \tag{2.27}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix become:

$$1 \leq i \leq 3 \text{ and } 1 \leq j \leq 3$$

$$\begin{aligned}
K1_{2i-1,2j-1} &= \iint_{\hat{E}} \left(4H\mu \frac{\partial L_j}{\partial \varphi_x(r,s)} \frac{\partial L_i}{\partial \varphi_x(r,s)} + H\mu \frac{\partial L_j}{\partial \varphi_y(r,s)} \frac{\partial L_i}{\partial \varphi_y(r,s)} \right) |J| d\hat{S} \\
&\simeq \sum_{g=1}^{3} w_g \left(4H\mu \frac{\partial L_j}{\partial \varphi_x(r,s)} \frac{\partial L_i}{\partial \varphi_x(r,s)} + H\mu \frac{\partial L_j}{\partial \varphi_y(r,s)} \frac{\partial L_i}{\partial \varphi_y(r,s)} \right) |J| \\
K1_{2i-1,2j} &= \iint_{\hat{E}} \left(2H\mu \frac{\partial L_j}{\partial \varphi_y(r,s)} \frac{\partial L_i}{\partial \varphi_x(r,s)} + H\mu \frac{\partial L_j}{\partial \varphi_x(r,s)} \frac{\partial L_i}{\partial \varphi_y(r,s)} \right) |J| d\hat{S} \\
&\simeq \sum_{g=1}^{3} w_g \left(2H\mu \frac{\partial L_j}{\partial \varphi_y(r,s)} \frac{\partial L_i}{\partial \varphi_x(r,s)} + H\mu \frac{\partial L_j}{\partial \varphi_x(r,s)} \frac{\partial L_i}{\partial \varphi_y(r,s)} \right) |J| \\
K1_{2i,2j-1} &= \iint_{\hat{E}} \left(2H\mu \frac{\partial L_j}{\partial \varphi_x(r,s)} \frac{\partial L_i}{\partial \varphi_y(r,s)} + H\mu \frac{\partial L_j}{\partial \varphi_y(r,s)} \frac{\partial L_i}{\partial \varphi_x(r,s)} \right) |J| d\hat{S} \\
&\simeq \sum_{g=1}^{3} w_g \left(2H\mu \frac{\partial L_j}{\partial \varphi_x(r,s)} \frac{\partial L_i}{\partial \varphi_y(r,s)} + H\mu \frac{\partial L_j}{\partial \varphi_y(r,s)} \frac{\partial L_i}{\partial \varphi_x(r,s)} \right) |J| \\
K1_{2i,2j} &= \iint_{\hat{E}} \left(4H\mu \frac{\partial L_j}{\partial \varphi_y(r,s)} \frac{\partial L_i}{\partial \varphi_y(r,s)} + H\mu \frac{\partial L_j}{\partial \varphi_x(r,s)} \frac{\partial L_i}{\partial \varphi_x(r,s)} \right) |J| d\hat{S} \\
&\simeq \sum_{g=1}^{3} w_g \left(4H\mu \frac{\partial L_j}{\partial \varphi_y(r,s)} \frac{\partial L_i}{\partial \varphi_y(r,s)} + H\mu \frac{\partial L_j}{\partial \varphi_x(r,s)} \frac{\partial L_i}{\partial \varphi_x(r,s)} \right) |J|
\end{aligned}$$

This elementary stiffness matrix is calculated with three matrices: $K1 = \sum_{g=1}^{3} [B]^T [D] [B']$ as follows[3]:

[3] $[I_n]$ is a common notation for an identity matrix of size $n \times n$

$$D = 2w_g H\mu|J|\,[I_3]$$

$$[B]^T = \begin{bmatrix} \dfrac{\partial L_1}{\partial\varphi_x(r,s)} & 0 & \dfrac{1}{2}\dfrac{\partial L_1}{\partial\varphi_y(r,s)} \\ 0 & \dfrac{\partial L_1}{\partial\varphi_y(r,s)} & \dfrac{1}{2}\dfrac{\partial L_1}{\partial\varphi_x(r,s)} \\ \dfrac{\partial L_2}{\partial\varphi_x(r,s)} & 0 & \dfrac{1}{2}\dfrac{\partial L_2}{\partial\varphi_y(r,s)} \\ 0 & \dfrac{\partial L_2}{\partial\varphi_y(r,s)} & \dfrac{1}{2}\dfrac{\partial L_2}{\partial\varphi_x(r,s)} \\ \dfrac{\partial L_3}{\partial\varphi_x(r,s)} & 0 & \dfrac{1}{2}\dfrac{\partial L_3}{\partial\varphi_y(r,s)} \\ 0 & \dfrac{\partial L_3}{\partial\varphi_y(r,s)} & \dfrac{1}{2}\dfrac{\partial L_3}{\partial\varphi_x(r,s)} \end{bmatrix} \tag{2.28}$$

$$[B'] = \begin{bmatrix} 2\dfrac{\partial L_1}{\partial\varphi_x(r,s)} & \dfrac{\partial L_1}{\partial\varphi_y(r,s)} & 2\dfrac{\partial L_2}{\partial\varphi_x(r,s)} & \dfrac{\partial L_2}{\partial\varphi_y(r,s)} & 2\dfrac{\partial L_3}{\partial\varphi_x(r,s)} & \dfrac{\partial L_3}{\partial\varphi_y(r,s)} \\ \dfrac{\partial L_1}{\partial\varphi_x(r,s)} & 2\dfrac{\partial L_1}{\partial\varphi_y(r,s)} & \dfrac{\partial L_2}{\partial\varphi_x(r,s)} & 2\dfrac{\partial L_2}{\partial\varphi_y(r,s)} & \dfrac{\partial L_3}{\partial\varphi_x(r,s)} & 2\dfrac{\partial L_3}{\partial\varphi_y(r,s)} \\ \dfrac{\partial L_1}{\partial\varphi_y(r,s)} & \dfrac{\partial L_1}{\partial\varphi_x(r,s)} & \dfrac{\partial L_2}{\partial\varphi_y(r,s)} & \dfrac{\partial L_2}{\partial\varphi_x(r,s)} & \dfrac{\partial L_3}{\partial\varphi_y(r,s)} & \dfrac{\partial L_3}{\partial\varphi_x(r,s)} \end{bmatrix} \tag{2.29}$$

NB: in $ISSM$ code, the factor 2 in the D matrix does not exist since the constitutive relation used is $\sigma' = \mu\varepsilon$ instead of $\sigma' = 2\mu\varepsilon$. This changes the viscosity definition that must be multiplied by 2.

2.1.7.2 Computation of nodal function derivatives

The computation of $\dfrac{\partial L_i}{\partial\varphi_x(r,s)}$ and $\dfrac{\partial L_i}{\partial\varphi_y(r,s)}$ is done using the Jacobian of φ. We have:

$$\begin{bmatrix} \dfrac{\partial L_i}{\partial\varphi_x(r,s)} \\ \dfrac{\partial L_i}{\partial\varphi_y(r,s)} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial r}{\partial\varphi_x(r,s)} & \dfrac{\partial s}{\partial\varphi_x(r,s)} \\ \dfrac{\partial r}{\partial\varphi_y(r,s)} & \dfrac{\partial s}{\partial\varphi_y(r,s)} \end{bmatrix} \begin{bmatrix} \dfrac{\partial L_i}{\partial r} \\ \dfrac{\partial L_i}{\partial s} \end{bmatrix} \tag{2.30}$$

The Jacobian of φ is:

$$[J_\varphi] = \begin{bmatrix} \dfrac{\partial\varphi_x}{\partial r} & \dfrac{\partial\varphi_x}{\partial s} \\ \dfrac{\partial\varphi_y}{\partial r} & \dfrac{\partial\varphi_y}{\partial s} \end{bmatrix} \tag{2.31}$$

We have: $[J_\varphi]^{-1} = \left[J_{\varphi^{-1}}\right]$. This gives us:

$$[J_\varphi]^{-1} = \begin{bmatrix} \dfrac{\partial r}{\partial\varphi_x} & \dfrac{\partial r}{\partial\varphi_y} \\ \dfrac{\partial s}{\partial\varphi_x} & \dfrac{\partial s}{\partial\varphi_y} \end{bmatrix} \tag{2.32}$$

And finally, we obtain:

$$\begin{bmatrix} \dfrac{\partial L_i}{\partial \varphi_x(r,s)} \\ \\ \dfrac{\partial L_i}{\partial \varphi_y(r,s)} \end{bmatrix} = \left([J]^{-1}\right)^T \begin{bmatrix} \dfrac{\partial L_i}{\partial r} \\ \\ \dfrac{\partial L_i}{\partial s} \end{bmatrix} \tag{2.33}$$

2.1.7.3 Basal drag

This matrix has the following shape:

$$\begin{aligned} K_{2i-1,2j-1} &= \iint_E \alpha^2 \phi_j \phi_i d\Omega \\ K_{2i-1,2j} &= 0 \\ K_{2i,2j-1} &= 0 \\ K_{2i,2j} &= \iint_E \alpha^2 \phi_j \phi_i d\Omega \end{aligned} \tag{2.34}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix become:

$$\begin{aligned} K_{2i-1,2j-1} &= \iint_{\hat{E}} \alpha^2 L_j(r,s) L_i(r,s) |J| d\hat{S} &&\simeq \sum_{g=1}^{3} w_g \alpha^2 L_j(r_g,s_g) L_i(r_g,s_g) |J| \\ K_{2i-1,2j} &= 0 \\ K_{2i,2j-1} &= 0 \\ K_{2i,2j} &= \iint_{\hat{E}} \alpha^2 L_j(r,s) L_i(r,s) |J| d\hat{S} &&\simeq \sum_{g=1}^{3} w_g \alpha^2 L_j(r_g,s_g) L_i(r_g,s_g) |J| \end{aligned} \tag{2.35}$$

The values $L_i(r_g, s_g)$ are equal to the component of the area coordinates (ie. for example $L_1(r_2, s_2) = 1/6$). This matrix is calculated with three matrices: $K1 = \sum_{g=1}^{3} [L]^T [D_{drag}][L]$ as follows:

$$[L] = \begin{bmatrix} L_1 & 0 & L_2 & 0 & L_3 & 0 \\ 0 & L_1 & 0 & L_2 & 0 & L_3 \end{bmatrix} \tag{2.36}$$

and $D = w_g \alpha^2 |J| \, [I_2]$

2.1.8 Elementary load vector

To compute the elementary load vector, we first calculate the force due to the driving stress only, andthen we and to this vector the force on the ice front due to the water pressure.

2.1.8.1 Driving stress

The elementary load vector due to driving stress has the following shape:

$$\begin{aligned} F1_{2i-1} &= -\iint_E \rho g H \frac{\partial s}{\partial x} \phi_i d\Omega \\ F1_{2i} &= -\iint_E \rho g H \frac{\partial s}{\partial y} \phi_i dS \end{aligned} \tag{2.37}$$

The slopes $\frac{\partial s}{\partial x}$ and $\frac{\partial s}{\partial y}$ are constant on the element. Using the reference element $\hat{E}$ and the gaussian points, this vector become:

$$\begin{aligned} F1_{2i-1} &= -\iint_{\hat{E}} \rho g H \frac{\partial s}{\partial x} L_i(r,s)|J| d\hat{S} & \simeq \sum_{g=1}^{3} -w_g \rho g H \frac{\partial s}{\partial x} L_i(r_g, s_g)|J| \\ F1_{2i} &= -\iint_{\hat{E}} \rho g H \frac{\partial s}{\partial y} L_i(r,s)|J| d\hat{S} & \simeq \sum_{g=1}^{3} -w_g \rho g H \frac{\partial s}{\partial y} L_i(r_g, s_g)|J| \end{aligned} \tag{2.38}$$

2.1.8.2 Ice front

The load on the ice front have the following shape:

$$\begin{aligned} F2_{2i-1} &= \int_{E\cup\Gamma} \left(\frac{1}{2}\rho g H^2 - \frac{1}{2}\rho_w g b^2\right) n_x \phi_i dl \\ F2_{2i} &= \int_{E\cup\Gamma} \left(\frac{1}{2}\rho g H^2 - \frac{1}{2}\rho_w g b^2\right) n_y \phi_i dl \end{aligned} \tag{2.39}$$

Here, the reference element are segment. Using these reference elements $\hat{E}$ and the gaussian points, this vector become:

$$\begin{aligned} F2_{2i-1} &= \int_{\hat{E}\cup\Gamma} \left(\frac{1}{2}\rho g H^2 - \frac{1}{2}\rho_w g b^2\right) n_x L_i |J| d\hat{l} & \simeq \sum_{g=1}^{2} w_g \left(\frac{1}{2}\rho g H(r_g)^2 - \frac{1}{2}\rho_w g b(r_g)^2\right) n_x L_i(r_g)|J| \\ F2_{2i} &= \int_{\hat{E}\cup\Gamma} \left(\frac{1}{2}\rho g H^2 - \frac{1}{2}\rho_w g b^2\right) n_y L_i |J| d\hat{l} & \simeq \sum_{g=1}^{2} w_g \left(\frac{1}{2}\rho g H(r_g)^2 - \frac{1}{2}\rho_w g b(r_g)^2\right) n_y L_i(r_g)|J| \end{aligned}$$

2.1.9 Assembly

Once an element stiffness matrix or a load vector is computed, it is plugged into the global stiffness matrix or global load vector. That step requires the knowledge of the global degrees of freedom. Indeed, each component of the element stiffness matrix or load vector is initially indexed by local degrees of freedom but each one of these local degrees of freedom is linked to a global degree of freedom. Once an elementary stiffness matrix is computed, each term is added to the corresponding term in the global stiffness matrix depending on the global degrees of freedom involved.

Let us illustrate this process with Fig.2.4, there are 6 local degrees of freedom for a given triangle of the mesh (3 nodes and two degrees of freedom per node: one for x and another for y). The local degree of freedom number 5 correspond to a global degree of freedom number 809, and the local degree of freedom number 1 is the global degree of freedom number 12. Therefore, the component $Ke_{1,5}$ of the element stiffness matrix must be added to $K_{12,809}$ of the global stiffness matrix.

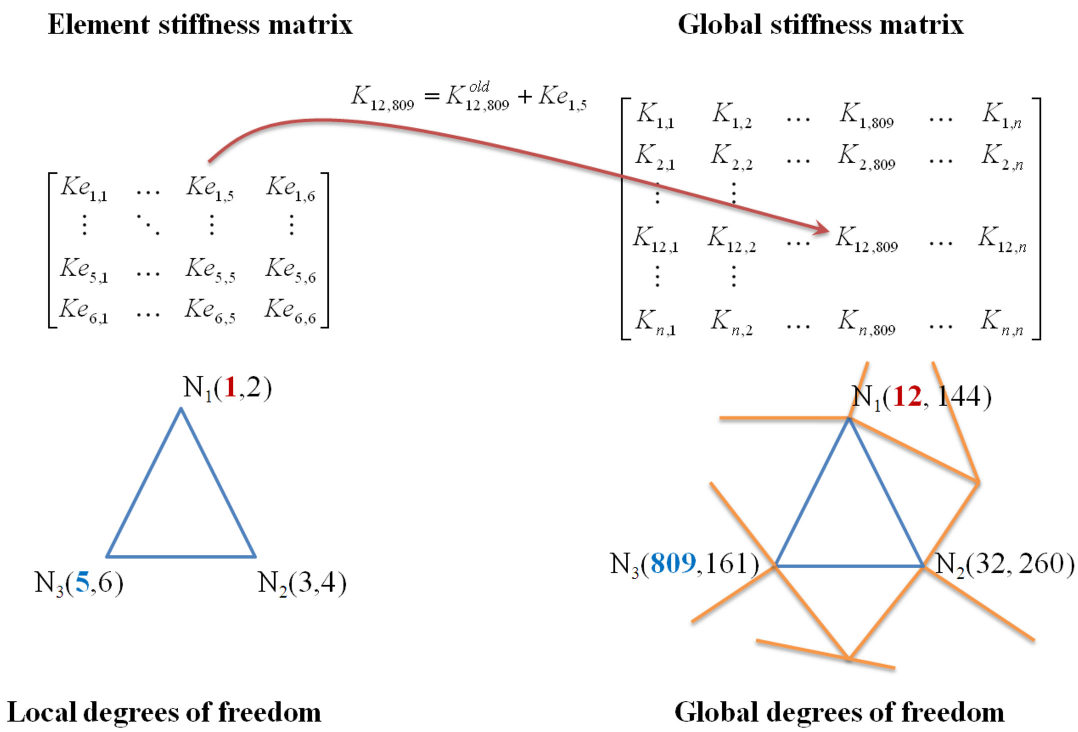


Figure 2.4: Assembly process

After the assembly process, one obtains a global stiffness matrix whose size is $2nods \times 2nods$, but to build the weak formulation, we made the assumption that ϕ was kinematically admissible:

$$\overrightarrow{\phi}(x,y) \in \{\left(H^1(\Omega \cup \partial\Omega)\right)^2 \quad \backslash \quad \overrightarrow{\phi}(x,y) = 0 \qquad (x,y) \in \Gamma_u\} \tag{2.40}$$

Therefore, all the nodal functions of the nods located on Γ_u that we used to build the global stiffness matrix are not kinematically admissible. We must reduce the global stiffness matrix K_{gg} to the final stiffness matrix K_{ff} by erasing all the rows and columns related to these nodal functions.

2.1.10 Resolution

We saw that the velocity on each node could be calculated with $U = K^{-1}F$, but we assumed that the specified velocity U_s of the nodes located on Γ_u was 0. To take into account the fact that the constraint can be different than 0, the velocity field is calculated as follows:

$$U_f = [K_{ff}]^{-1} (F + K_{fs}U_s) \tag{2.41}$$

Where f stands for "free" and s for "single point constraint". Then, we must add the single point constraints to the vector U_f in order to have the velocity of all the nodes of the domain U_g.

2.1.11 Summary

The computation of the velocity field is not linear since the viscosity depends directly on the solution. The following algorithm is used: an initial viscosity is assumed. Each elementary stiffness matrix and load vector is computed using gauss quadratures. They are then plugged into the global stiffness matrix and global load vector. The global stiffness matrix and load vector are reduced depending on the constraints (The velocity of a constrained node is not computed, the corresponding line and column in the stiffness matrix and load vector must be deleted). Then the velocity field is computed and compared to the one of the previous iteration (for the first iteration, there is no convergence test, a second iteration must be launched). If the convergence criterion is not fulfilled (ei. the results are too different from one another), a new iteration is computed, using the velocity field of the previous iteration to compute the viscosity.

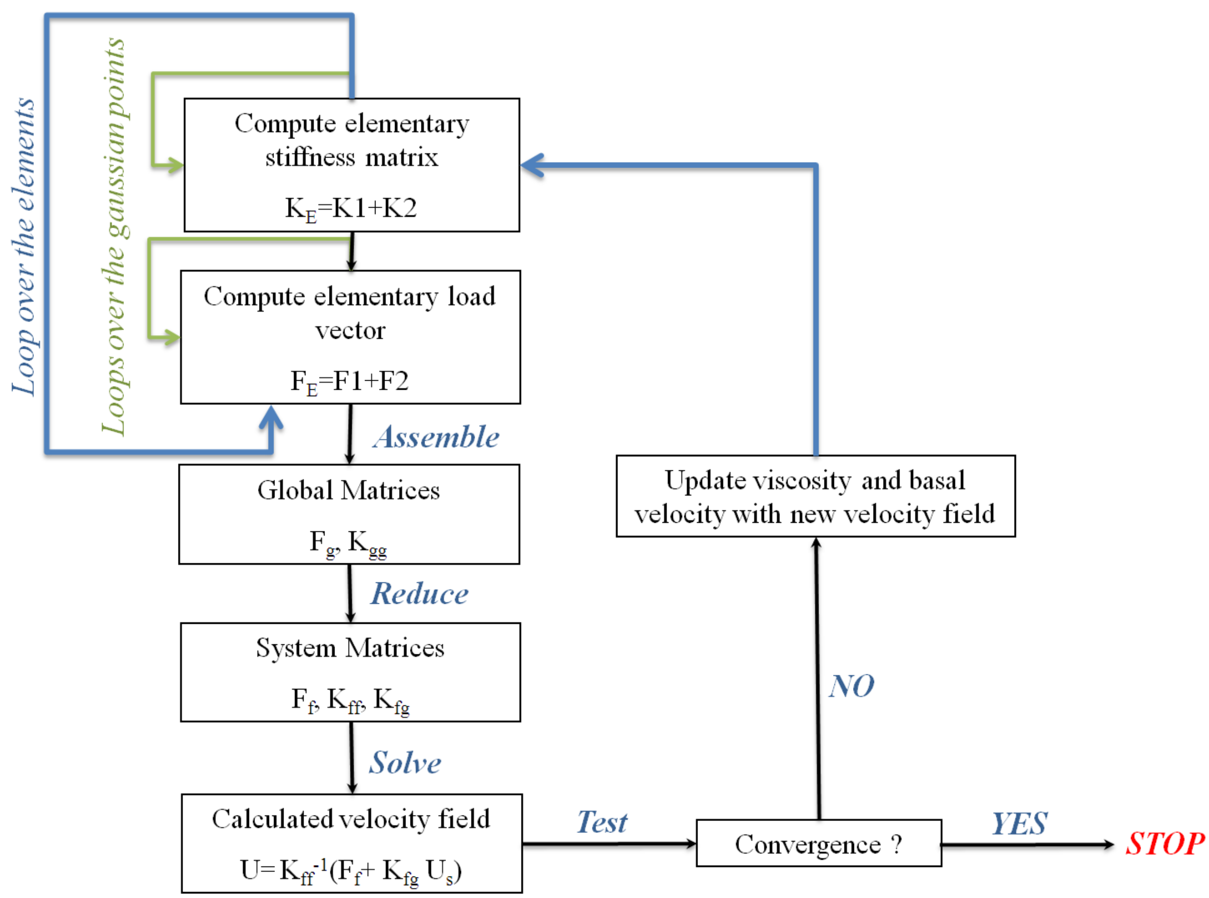


Figure 2.5: Schematic algorithm used by *ISSM*

2.2 Pattyn's Higher-order model

2.2.1 Geometry and notations

The glacier system is 3d. With the assumptions detailed in the first chapter, we only solve the problem for the horizontal components of velocity and deduce the vertical term from them using the incompressibility.

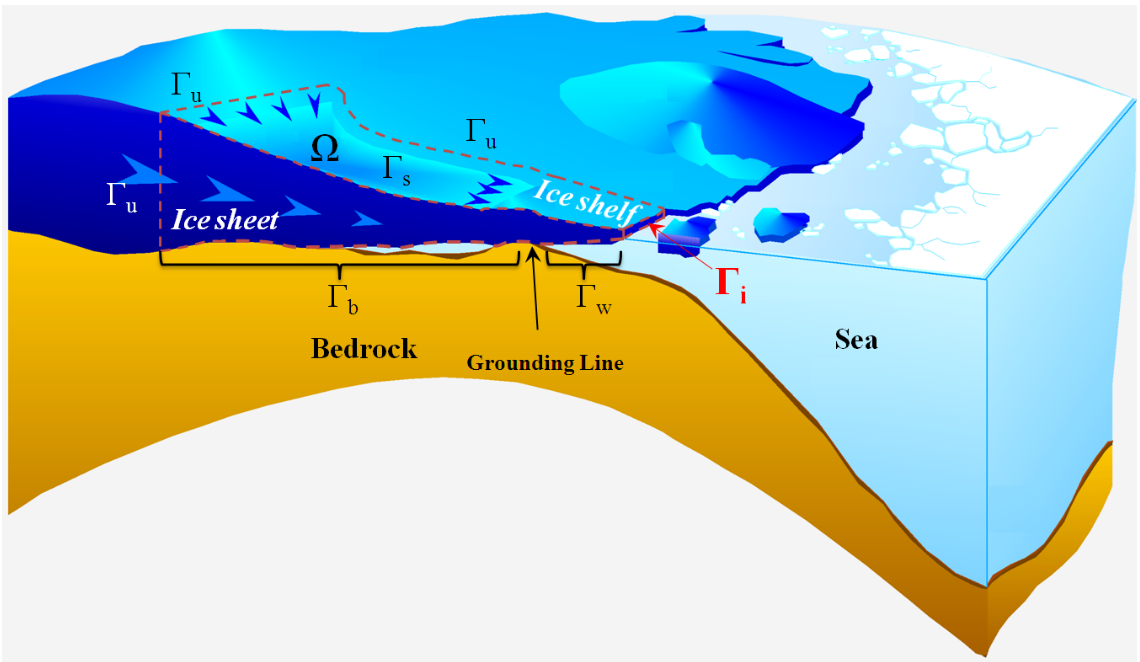


Figure 2.6: Geometry and notations

We adopt the following notations:

- Ω is the open surface that constitutes the entire glacier system
- $\partial\Omega$ are the points located at the boundary
- Γ_i are points at the ice front (Ice/Water interface)
- Γ_s are points at the surface of the glacier (Ice/Atmosphere interface $z = s$)
- Γ_b are points at the interface between glacier and bedrock (Ice/Till interface $z = b$)
- Γ_w are points at the interface between glacier and water (Ice/Water interface $z = b$)
- $\Gamma_u = \partial\Omega \backslash (\Gamma_i \cup \Gamma_s \cup \Gamma_b \cup \Gamma_w)$ are all other points on the boundary

2.2.2 Equations

The equations of Pattyn's model are (1.52):

$$\left\{ \begin{array}{l} \frac{\partial}{\partial x}\left(4\mu\frac{\partial u}{\partial x} + 2\mu\frac{\partial v}{\partial y}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial z}\right) = \rho g\frac{\partial s}{\partial x} \\ \frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial y}\left(4\mu\frac{\partial v}{\partial y} + 2\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial z}\left(\mu\frac{\partial v}{\partial z}\right) = \rho g\frac{\partial s}{\partial y} \end{array} \right. \tag{2.42}$$

The boundary conditions are: a dynamic boundary condition (Neumann) at the ice front, a friction law on the ice sheet, and single point constraint (Dirichlet) on the other borders of the domain Ω. It is more convenient to consider these specified velocities as nill. At the end of the process, the velocity field will be updated to take these constraints into account.

$$\begin{cases} \text{Ice Front } \Gamma_i : & \sigma' \overrightarrow{n} = F_i \overrightarrow{n} = \left(\rho_i g \left(s - z\right) + \rho_w g \min\left(0, z\right)\right) \overrightarrow{n} \\ \text{Upper surface } \Gamma_s : & \sigma' \left(\overrightarrow{n}\right) = 0 \\ \text{Ice sheet base } \Gamma_b : & \left(\sigma' \left(\overrightarrow{n}\right)\right)_x = -K^2 N_{eff}^r \|\overrightarrow{v}\|^{s-1} u = -\alpha^2 u \\ & \left(\sigma' \left(\overrightarrow{n}\right)\right)_y = -K^2 N_{eff}^r \|\overrightarrow{v}\|^{s-1} v = -\alpha^2 v \\ \text{Ice shelf base } \Gamma_w : & \sigma' \overrightarrow{n} = F_w \overrightarrow{n} = g \left(\rho_i h(x,y) + \rho_w g b(x,y)\right) \overrightarrow{n} \\ \text{Other borders } \Gamma_u : & u = v = 0 \end{cases} \tag{2.43}$$

For the boundary condition on the ice sheet base, we use $r = q/p$ and $s = 1/p$ as explained in the first chapter (**??**)

2.2.3 Weak Formulation

Now that we have all the equations and boundary conditions, we write the weak formulation to solve these equations using finite element. We define a kinematically admissible velocity field $\overrightarrow{\phi}$:

$$\overrightarrow{\phi}(x,y) \in \{\left(H^1(\Omega \cup \partial\Omega)\right)^2 \quad \backslash \quad \overrightarrow{\phi}(x,y,z) = 0 \qquad (x,y,z) \in \Gamma_u\} \tag{2.44}$$

For any function of this virtual field, we take the scalar product of the equations with $\overrightarrow{\phi} = (\phi_x, 0)$ and $\overrightarrow{\phi} = (0, \phi_y)$ that are 2 other kinematically admissible velocity fields, and integrate these products over the domain to obtain:

$$\iiint_\Omega \left[\frac{\partial}{\partial x}\left(4\mu\frac{\partial u}{\partial x} + 2\mu\frac{\partial v}{\partial y}\right)\phi_x + \frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right)\phi_x + \frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial z}\right)\phi_x - \rho g\frac{\partial s}{\partial x}\phi_x\right] d\Omega = 0$$

$$\iiint_\Omega \left[\frac{\partial}{\partial y}\left(4\mu\frac{\partial v}{\partial y} + 2\mu\frac{\partial u}{\partial x}\right)\phi_y + \frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right)\phi_y + \frac{\partial}{\partial z}\left(\mu\frac{\partial v}{\partial z}\right)\phi_y - \rho g\frac{\partial s}{\partial y}\phi_y\right] d\Omega = 0$$

One intergrates by parts the first three terms in order to merge in the force on the ice front Γ_i, the interface with the bedrock Γ_b (ice sheet base) and the interface with the water Γ_w (ice shelf base):

$$\begin{aligned}\iiint_\Omega \left(4\mu\frac{\partial u}{\partial x} + 2\mu\frac{\partial v}{\partial y}\right)\frac{\partial \phi_x}{\partial x} + \left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right)\frac{\partial \phi_x}{\partial y} + \mu\frac{\partial u}{\partial z}\frac{\partial \phi_x}{\partial z} d\Omega& \\ = \iint_{\Gamma_i \cup \Gamma_b \cup \Gamma_w} F n_x \phi_x d\Gamma - \iiint_\Omega \rho g \frac{\partial s}{\partial x}\phi_x d\Omega&\end{aligned} \tag{2.45}$$

$$\begin{aligned}\iiint_\Omega \left(4\mu\frac{\partial v}{\partial y} + 2\mu\frac{\partial u}{\partial x}\right)\frac{\partial \phi_y}{\partial y} + \left(\mu\frac{\partial v}{\partial x} + \mu\frac{\partial u}{\partial y}\right)\frac{\partial \phi_y}{\partial x} + \mu\frac{\partial v}{\partial z}\frac{\partial \phi_y}{\partial z} d\Omega& \\ = \iint_{\Gamma_i \cup \Gamma_b \cup \Gamma_w} F n_y \phi_y d\Gamma - \iiint_\Omega \rho g \frac{\partial s}{\partial y}\phi_y d\Omega&\end{aligned} \tag{2.46}$$

It is easier to put the basal friction at the left hand side of the equation since it is velocity dependant. The equations become:

$$\begin{aligned}\iiint_\Omega \left(4\mu\frac{\partial u}{\partial x} + 2\mu\frac{\partial v}{\partial y}\right)\frac{\partial \phi_x}{\partial x} + \left(\mu\frac{\partial u}{\partial y} + \mu\frac{\partial v}{\partial x}\right)\frac{\partial \phi_x}{\partial y} + \mu\frac{\partial u}{\partial z}\frac{\partial \phi_x}{\partial z} d\Omega + \iint_{\Gamma_b} \alpha^2 u \phi_x d\Gamma& \\ = \iint_{\Gamma_i} F_i n_x \phi_x d\Gamma + \iint_{\Gamma_w} F_w n_x \phi_y x\Gamma - \iiint_\Omega \rho g \frac{\partial s}{\partial x}\phi_x d\Omega&\end{aligned} \tag{2.47}$$

$$\begin{aligned}\iiint_\Omega \left(4\mu\frac{\partial v}{\partial y} + 2\mu\frac{\partial u}{\partial x}\right)\frac{\partial \phi_y}{\partial y} + \left(\mu\frac{\partial v}{\partial x} + \mu\frac{\partial u}{\partial y}\right)\frac{\partial \phi_y}{\partial x} + \mu\frac{\partial v}{\partial z}\frac{\partial \phi_y}{\partial z} d\Omega + \iint_{\Gamma_b} \alpha^2 v \phi_y d\Gamma& \\ = \iint_{\Gamma_i} F_i n_y \phi_y d\Gamma + \iint_{\Gamma_w} F_w n_y \phi_y d\Gamma - \iiint_\Omega \rho g \frac{\partial s}{\partial y}\phi_y d\Omega&\end{aligned} \tag{2.48}$$

2.2.4 Finite element discretisation, Galerkin method

The domain is meshed using pentahedrons. There are a *nel* elements and *nods* grids. We use the nodal functions of each grid, which are continuous on the domain and affine on each element. We now project the unknows u and v on the nodal functions:

$$(\phi_i)_{1\leq i\leq nods}$$

$$\begin{aligned} u(x,y,z) &= \sum_{i=1}^{nods} u_i\phi_i(x,y,z) \\ v(x,y,z) &= \sum_{i=1}^{nods} v_i\phi_i(x,y,z) \end{aligned} \tag{2.49}$$

We build a vector U that holds all the unknows of the velocity field:

$$U = \begin{bmatrix} u_1 \\ v_1 \\ \dots \\ u_{nods} \\ v_{nods} \end{bmatrix} \tag{2.50}$$

With this new field, we replace the component of the virtual function $\overrightarrow{\phi}$ by the nodal functions:

$$\forall 1 \leq i \leq nods$$

$$\begin{aligned} \sum_{j=1}^{nods} \iiint_\Omega \left(4\mu u_j \frac{\partial\phi_j}{\partial x} + 2\mu v_j \frac{\partial\phi_j}{\partial y}\right)\frac{\partial\phi_i}{\partial x} + \left(\mu u_j \frac{\partial\phi_j}{\partial y} + \mu v_j \frac{\partial\phi_j}{\partial x}\right)\frac{\partial\phi_i}{\partial y} + \mu u_j \frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z} d\Omega \\ + \iint_{\Gamma_b} \alpha^2 u_j\phi_j\phi_i d\Gamma = \iint_{\Gamma_i} F_i n_x \phi_i d\Gamma + \iint_{\Gamma_w} F_w n_x \phi_i d\Gamma - \iiint_\Omega \rho g \frac{\partial s}{\partial x}\phi_i d\Omega \end{aligned} \tag{2.51}$$

$$\begin{aligned} \sum_{j=1}^{nods} \iiint_\Omega \left(\mu u_j \frac{\partial\phi_j}{\partial y} + \mu v_j \frac{\partial\phi_j}{\partial x}\right)\frac{\partial\phi_i}{\partial x} + \left(4\mu v_j \frac{\partial\phi_j}{\partial y} + 2\mu u_j \frac{\partial\phi_i}{\partial x}\right)\frac{\partial\phi_i}{\partial y} + \mu v_j \frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z} d\Omega \\ + \iint_{\Gamma_b} \alpha^2 v_j\phi_j\phi_i d\Gamma = \iint_{\Gamma_i} F_i n_y \phi_i d\Gamma + \iint_{\Gamma_w} F_w n_y \phi_i d\Gamma - \iiint_\Omega \rho g \frac{\partial s}{\partial y}\phi_i d\Omega \end{aligned} \tag{2.52}$$

By rearranging terms:

$$\forall 1 \leq i \leq nods$$

$$\begin{aligned} \sum_{j=1}^{nods} \iiint_\Omega \left[4\mu \frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x} + \mu \frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y} + \mu \frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}\right] u_j + \left[2\mu \frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x} + \mu \frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}\right] v_j d\Omega \\ + \iint_{\Gamma_b} \alpha^2 u_j\phi_j\phi_i d\Gamma = \iint_{\Gamma_i} F_i n_x \phi_i d\Gamma + \iint_{\Gamma_w} F_w n_x \phi_i d\Gamma - \iiint_\Omega \rho g \frac{\partial s}{\partial x}\phi_i d\Omega \end{aligned} \tag{2.53}$$

$$\begin{aligned} \sum_{j=1}^{nods} \iiint_\Omega \left[\mu \frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x} + 2\mu \frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}\right] u_j + \left[\mu \frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x} + 4\mu \frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y} + \mu \frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}\right] v_j d\Omega \\ + \iint_{\Gamma_b} \alpha^2 v_j\phi_j\phi_i d\Gamma = \iint_{\Gamma_i} F_i n_y \phi_i d\Gamma + \iint_{\Gamma_w} F_w n_y \phi_i d\Gamma - \iiint_\Omega \rho g \frac{\partial s}{\partial y}\phi_i d\Omega \end{aligned} \tag{2.54}$$

We see the stiffness matrix K and load vector F appearing, so that the previous equations can be written as a matrix form:

$$[K]\,U = F \tag{2.55}$$

We use the following index:

$$\begin{bmatrix} & \vdots & \vdots & \\ \cdots & K_{2i-1,2j-1} & K_{2i-1,2j} & \cdots \\ \cdots & K_{2i,2j-1} & K_{2i,2j} & \cdots \\ & \vdots & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ u_j \\ v_j \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ F_{2j-1} \\ F_{2j} \\ \vdots \end{bmatrix} \tag{2.56}$$

And we have the following expressions for the stiffness matrix and the load vector :

$$\begin{aligned}
K_{2i-1,2j-1} &= \iiint_\Omega 4\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x} + \mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y} + \mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}d\Omega + \iint_{\Gamma_b}\alpha^2 u_j\phi_j\phi_i d\Gamma \\
K_{2i-1,2j} &= \iiint_\Omega 2\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x} + \mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}d\Omega \\
K_{2i,2j-1} &= \iiint_\Omega 2\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y} + \mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x}d\Omega \\
K_{2i,2i} &= \iiint_\Omega \mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x} + 4\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y} + \mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}d\Omega + \iint_{\Gamma_b}\alpha^2 v_j\phi_j\phi_i d\Gamma \\
F_{2i-1} &= \iint_{\Gamma_i} F_i n_x\phi_i d\Gamma + \iint_{\Gamma_w} F_w n_x\phi_i d\Gamma - \iiint_\Omega \rho g\frac{\partial s}{\partial x}\phi_i d\Omega \\
F_{2i} &= \iint_{\Gamma_i} F_i n_y\phi_i d\Gamma + \iint_{\Gamma_w} F_w n_y\phi_i d\Gamma - \iiint_\Omega \rho g\frac{\partial s}{\partial y}\phi_i d\Omega
\end{aligned} \tag{2.57}$$

2.2.5 Decomposition over the elements and reference element

The integrals over Ω can be divided in several integrals over each elements. To compute the matrices K and F, instead of calculating the integral over Ω directly, we will first calculate the integrales over each element E of the mesh, that will give us several *Element Matrices*, that will be plugged into the global matrices to get the final Stiffness matrix and load vector.

One element E gives 12 equations: we can take 6 nodal functions for $\phi_i(x, y, z)$ (the nodal functions corresponding to the vertexes of E, which is an pentahedron), each ϕ_i gives 2 equations (one for the x-component and one for the y-component). Therefore, an elementary stiffness matrix K_E will be 12×12, and the elementary load vector will be of size 19.

Exactly as in MacAyeal's model, rather than calculating these integrals for each element, one uses a *Reference Element* $\hat{E}$, and a transformation φ that transforms the reference element to the element E (See Fig.2.7):

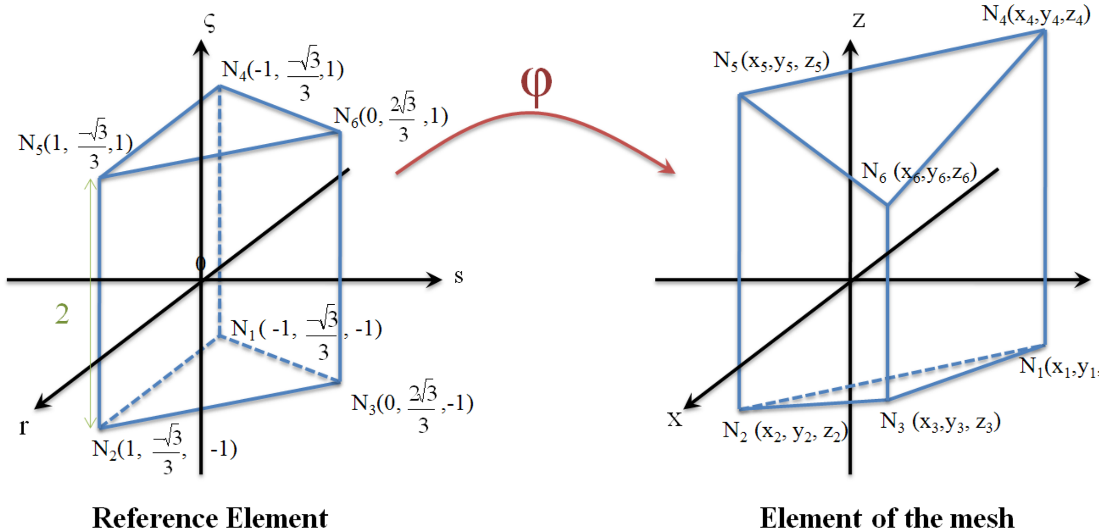


Figure 2.7: Transformation from the reference element, to the current element of the pentahedral mesh

For any function $f(x,y,z)$, the transformation verifies:

$$\iiint_E f(x,y,z)dV = \iiint_{\hat{E}} \hat{f}(r,s,\zeta)|J|d\hat{V} \tag{2.58}$$

Where J is the Jacobian determinant of the φ function. This Jacobian is calculated in section A.5:

$$J = \begin{bmatrix} \frac{\partial \varphi_x}{\partial r} & \frac{\partial \varphi_x}{\partial s} & \frac{\partial \varphi_x}{\partial \zeta} \\ \frac{\partial \varphi_y}{\partial r} & \frac{\partial \varphi_y}{\partial s} & \frac{\partial \varphi_y}{\partial \zeta} \\ \frac{\partial \varphi_z}{\partial r} & \frac{\partial \varphi_z}{\partial s} & \frac{\partial \varphi_z}{\partial \zeta} \end{bmatrix} \tag{2.59}$$

with:

$$\begin{aligned}
J_{11} &= \frac{1}{4}(x_1 - x_2 - x_4 + x_5) + \frac{1}{4}(-x_1 + x_2 - x_4 + x_5)\zeta \\
J_{21} &= \frac{1}{4}(y_1 - y_2 - y_4 + y_5) + \frac{1}{4}(-y_1 + y_2 - y_4 + y_5)\zeta \\
J_{31} &= \frac{1}{4}(z_1 - z_2 - z_4 + z_5) + \frac{1}{4}(-z_1 + z_2 - z_4 + z_5)\zeta \\
J_{12} &= \frac{\sqrt{3}}{12}(x_1 + x_2 - 2x_3 - x_4 - x_5 + 2x_6) + \frac{\sqrt{3}}{12}(-x_1 - x_2 + 2x_3 - x_4 - x_5 + 2x_6)\zeta \\
J_{22} &= \frac{\sqrt{3}}{12}(y_1 + y_2 - 2y_3 - y_4 - y_5 + 2y_6) + \frac{\sqrt{3}}{12}(-y_1 - y_2 + 2y_3 - y_4 - y_5 + 2y_6)\zeta \\
J_{32} &= \frac{\sqrt{3}}{12}(z_1 + z_2 - 2z_3 - z_4 - z_5 + 2z_6) + \frac{\sqrt{3}}{12}(-z_1 - z_2 + 2z_3 - z_4 - z_5 + 2z_6)\zeta
\end{aligned} \tag{2.60}$$

$$J_{31} = \frac{1}{4}\left(-x_1 + x_2 - x_4 + x_5\right) r + \frac{\sqrt{3}}{12}\left(-x_1 - x_2 + 2x_3 - x_4 - x_5 + 2x_6\right) s$$

$$J_{32} = \frac{1}{4}\left(-y_1 + y_2 - y_4 + y_5\right) r + \frac{\sqrt{3}}{12}\left(-y_1 - y_2 + 2y_3 - y_4 - y_5 + 2y_6\right) s$$

$$J_{33} = \frac{1}{4}\left(-z_1 + z_2 - z_4 + z_5\right) r + \frac{\sqrt{3}}{12}\left(-z_1 - z_2 + 2z_3 - z_4 - z_5 + 2z_6\right) s$$

Now, we have all the tools to calculate the integrals over the reference element instead of the element itself. Since $ISSM$ is a Matlab code, the integrals cannot be calculated directly. The numerical integration uses a gaussian quadrature to approximate the integral as a weighted sum of function values at specified points (gaussian points) within the domain:

$$\iiint_E f(x,y,z)dV = \iint_{\hat{E}} \hat{f}(r,s,\zeta)|J|d\hat{V} \simeq \sum_{g=1}^{n} W_g \hat{f}(r_g, s_g, \zeta_g) \tag{2.61}$$

where W_i are the quadrature rules weights and (r_g, s_g, ζ_g) are the evaluation points, $g = 1, 2..., n$. The integration points are given with their area coordinates:

$$(r_g, s_g, \zeta_g) \equiv (\frac{A_1}{A}, \frac{A_2}{A}, \frac{A_3}{A}, \zeta_g) \tag{2.62}$$

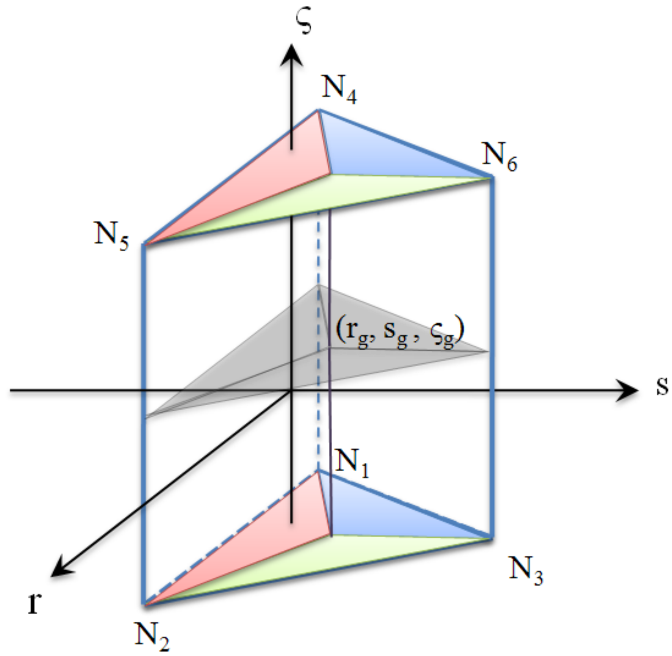


Figure 2.8: Area coordinates

The number of gaussian points depends on the order of integration. The order of integration can be computed from the polynomial degree p that needs to be integrated. The formula is:

$$order = \frac{p+1}{2} \tag{2.63}$$

Here, the degree of the polynome under the integral is 2 (the drag implies the calculation of $\phi_j \phi_i$ whose degree is 2). The order of integration is hence 2 (the order must be an integer). For this order, 6 gaussian points are required (See in $ISSM$ code $GaussPenta.m$ for more details).

2.2.6 Elementary stiffness matrix

To compute the elementary stiffness matrix, we first remove the basal drag, that will create a first elementary stiffness matrix $K1$, the basal drag will constitute another stiffness matrix $K2$. The elementary stiffness matrix will be the sum of $K1$ and $K2$ ($K_E = K1 + K2$).

2.2.6.1 Without basal drag

This matrix has the following shape:

$$1 \leq i \leq 6 \text{ and } 1 \leq j \leq 6$$

$$\begin{aligned}
K1_{2i-1,2j-1} &= \iiint_E \left(4\mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x} + \mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y} + \mu \frac{\partial \phi_j}{\partial z}\frac{\partial \phi_i}{\partial z} \right) d\Omega \\
K1_{2i-1,2j} &= \iiint_E \left(2\mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial x} + \mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial y} \right) d\Omega \\
K1_{2i,2j-1} &= \iiint_E \left(2\mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial y} + \mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial x} \right) d\Omega \\
K1_{2i,2j} &= \iiint_E \left(4\mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y} + \mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x} + \mu \frac{\partial \phi_j}{\partial z}\frac{\partial \phi_i}{\partial z} \right) d\Omega
\end{aligned} \tag{2.64}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix becomes:

$$1 \leq i \leq 6 \text{ and } 1 \leq j \leq 6$$

$$\begin{aligned}
K1_{2i-1,2j-1} &= \iiint_{\hat{E}} \left(4\mu \frac{\partial L_j}{\partial \varphi_x}\frac{\partial L_i}{\partial \varphi_x} + \mu \frac{\partial L_j}{\partial \varphi_y}\frac{\partial L_i}{\partial \varphi_y} + \mu \frac{\partial L_j}{\partial \varphi_z}\frac{\partial L_i}{\partial \varphi_z} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{6} W_g \left(4\mu \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g + \mu \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g + \mu \left.\frac{\partial L_j}{\partial \varphi_z}\right|_g \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \right) |J_g| \\
K1_{2i-1,2j} &= \iiint_{\hat{E}} \left(2\mu \frac{\partial L_j}{\partial \varphi_y}\frac{\partial L_i}{\partial \varphi_x} + \mu \frac{\partial L_j}{\partial \varphi_x}\frac{\partial L_i}{\partial \varphi_y} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{6} W_g \left(2\mu \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g + \mu \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g \right) |J_g| \\
K1_{2i,2j-1} &= \iiint_{\hat{E}} \left(2\mu \frac{\partial L_j}{\partial \varphi_x}\frac{\partial L_i}{\partial \varphi_y} + \mu \frac{\partial L_j}{\partial \varphi_y}\frac{\partial L_i}{\partial \varphi_x} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{6} W_g \left(2\mu \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g + \mu \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g \right) |J_g| \\
K1_{2i,2j} &= \iiint_{\hat{E}} \left(4\mu \frac{\partial L_j}{\partial \varphi_y}\frac{\partial L_i}{\partial \varphi_y} + \mu \frac{\partial L_j}{\partial \varphi_x}\frac{\partial L_i}{\partial \varphi_x} + \mu \frac{\partial L_j}{\partial \varphi_z}\frac{\partial L_i}{\partial \varphi_z} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{6} W_g \left(4\mu \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g + \mu \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g + \mu \left.\frac{\partial L_j}{\partial \varphi_z}\right|_g \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \right) |J_g|
\end{aligned}$$

This elementary stiffness matrix is calculated with three matrices: $K1 = \sum_{g=1}^{6} [B]^T [D][B']$ as follows:

$$[B] = \begin{bmatrix} B_1 & B_2 & B_3 & B_4 & B_5 & B_6 \end{bmatrix} \quad and \quad [B'] = \begin{bmatrix} B'_1 & B'_2 & B'_3 & B'_4 & B'_5 & B'_6 \end{bmatrix} \tag{2.65}$$

with:

$$[B_i] = \begin{bmatrix} \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & 0 \\ 0 & \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g \\ \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_x}\right|_g \\ \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & 0 \\ 0 & \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \end{bmatrix} \qquad [B'_i] = \begin{bmatrix} 2\left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g \\ \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & 2\left.\frac{\partial L_i}{\partial \varphi_y}\right|_g \\ \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g \\ \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & 0 \\ 0 & \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \end{bmatrix} \tag{2.66}$$

and $D = W_g 2\mu |J_g| \, [I_5]$

NB: in $ISSM$ code, the factor 2 in the D matrix does not exist since the constitutive relation used is $\sigma' = \mu\varepsilon$ instead of $\sigma' = 2\mu\varepsilon$. This changes the viscosity definition that must be multiplied by 2.

2.2.6.2 Basal drag

Basal drag get involved only for the nods on the ice/bedrock interface. Thus, only surface integrals are required. The elements are triangles and there are only 3 nodal functions to take into account. Therefore, the stiffness matrix that stands for basal drag is 6×6.
The basal drag matrix has the following shape:

$$1 \leq i \leq 3 \text{ and } 1 \leq j \leq 3$$

$$\begin{aligned} K_{2i-1,2j-1} &= \iint_{E\cap\Gamma_b} \alpha^2 \phi_j \phi_i dS \\ K_{2i-1,2j} &= 0 \\ K_{2i,2j-1} &= 0 \\ K_{2i,2j} &= \iint_{E\cap\Gamma_b} \alpha^2 \phi_j \phi_i dS \end{aligned} \tag{2.67}$$

This time, the reference element are triangles, as in MacAyeal's model. Using the reference element $\hat{E}$ and the gaussian points, this matrix become:

$$
\begin{aligned}
K_{2i-1,2j-1} &= \iint_{\hat{E}\cap\hat{\Gamma_b}} \alpha^2 L_j(r,s)L_i(r,s)|J|d\hat{S} &&\simeq \sum_{g=1}^{3} W_g\alpha^2 L_j(r_g,s_g)L_i(r_g,s_g)|J_g| \\
K_{2i-1,2j} &= 0 \\
K_{2i,2j-1} &= 0 \\
K_{2i,2j} &= \iint_{\hat{E}\cap\hat{\Gamma_b}} \alpha^2 L_j(r,s)L_i(r,s)|J|d\hat{S} &&\simeq \sum_{g=1}^{3} W_g\alpha^2 L_j(r_g,s_g)L_i(r_g,s_g)|J_g|
\end{aligned}
\tag{2.68}
$$

This matrix is calculated with three matrices: $K1 = \sum_{g=1}^{3}[L]^T[D_{drag}][L]$ as follows:

$$
[L] = \begin{bmatrix} L_1(r_g,s_g) & 0 & L_2(r_g,s_g) & 0 & L_3(r_g,s_g) & 0 \\ 0 & L_1(r_g,s_g) & 0 & L_2(r_g,s_g) & 0 & L_3(r_g,s_g) \end{bmatrix}
\tag{2.69}
$$

and $[D_{drag}] = W_g\alpha^2|J_g|\,[I_6]$

2.2.7 Elementary load vector

To compute the elementary load vector, we first calculate the force due to the driving stress only, the water pressure only, and we then add these two terms.

2.2.7.1 Driving stress

The elementary load vector due to driving stress has the following shape:

$$
\begin{aligned}
F1_{2i-1} &= -\iiint_E \rho g \frac{\partial s}{\partial x}\phi_i d\Omega \\
F1_{2i} &= -\iiint_E \rho g \frac{\partial s}{\partial y}\phi_i d\Omega
\end{aligned}
\tag{2.70}
$$

Using the reference element $\hat{E}$ and the gaussian points, this vector become:

$$
\begin{aligned}
F1_{2i-1} &= -\iiint_{\hat{E}} \rho g \frac{\partial s}{\partial x} L_i(r,s,\zeta)|J|d\hat{\Omega} &&\simeq \sum_{g=1}^{6} -W_g\rho g\frac{\partial s}{\partial x} L_i(r_g,s_g,\zeta_g)|J_g| \\
F1_{2i} &= -\iiint_{\hat{E}} \rho g \frac{\partial s}{\partial y} L_i(r,s,\zeta)|J|d\hat{\Omega} &&\simeq \sum_{g=1}^{6} -W_g\rho g\frac{\partial s}{\partial y} L_i(r_g,s_g,\zeta_g)|J_g|
\end{aligned}
\tag{2.71}
$$

2.2.7.2 Ice front

The load on the ice front involves only the nodes on the ice/sea interface. As with basal drag, only surface integrals are required. The elements are quadrangle (See sectionA.3). The load on the ice front have the following shape:

$$
\begin{aligned}
F2_{2i-1} &= \iint_{E\cap\Gamma_i} F_i n_x \phi_i dS \\
F2_{2i} &= \iint_{E\cap\Gamma_i} F_i n_y \phi_i dS
\end{aligned}
\tag{2.72}
$$

Using the reference element $\hat{E}$ and the gaussian points, this vector become:

$$
\begin{aligned}
F2_{2i-1} &= \iint_{\hat{E}\cup\hat{\Gamma}_i} \hat{F}_i n_x L_i |J| d\hat{S} &\simeq \sum_{g=1}^{4} W_g \left(\rho_i g\left(s-z\right) + \rho_w g \min\left(0,z\right)\right) n_x L_i(r_g,s_g,\zeta_g)|J_g| \\
F2_{2i} &= \iint_{\hat{E}\cup\hat{\Gamma}_i} \hat{F}_i n_y L_i |J| d\hat{S} &\simeq \sum_{g=1}^{4} W_g \left(\rho_i g\left(s-z\right) + \rho_w g \min\left(0,z\right)\right) n_y L_i(r_g,s_g,\zeta_g)|J_g|
\end{aligned}
\tag{2.73}
$$

2.2.7.3 Ice/water interface

The load on the ice/water interface involves only the nodes at the base on the iceshelf. As with basal drag and ice front, only surface integrals are required. The elements are triangles. The load on the ice shelf base have the following shape:

$$
\begin{aligned}
F3_{2i-1} &= \iint_{E\cap\Gamma_w} F_w n_x \phi_i dS \\
F3_{2i} &= \iint_{E\cap\Gamma_w} F_w n_y \phi_i dS
\end{aligned}
\tag{2.74}
$$

Using the reference element $\hat{E}$ and the gaussian points, this vector become:

$$
\begin{aligned}
F3_{2i-1} &= \iint_{\hat{E}\cup\hat{\Gamma}_w} \hat{F_w} n_x L_i |J| d\hat{S} &\simeq \sum_{g=1}^{3} W_g g \left(\rho_i h(r_g,s_g) + \rho_w b(r_g,s_g)\right) n_x L_i(r_g,s_g)|J_g| \\
F3_{2i} &= \iint_{\hat{E}\cup\hat{\Gamma}_w} \hat{F_w} n_y L_i |J| d\hat{S} &\simeq \sum_{g=1}^{3} W_g g \left(\rho_i h(r_g,s_g) + \rho_w b(r_g,s_g)\right) n_y L_i(r_g,s_g)|J_g|
\end{aligned}
\tag{2.75}
$$

2.3 Vertical Velocity computation

2.3.1 Equations

In the first chapter, we saw that we used the incompressibility to calculate the vertical component of the velocity:

$$\frac{\partial w}{\partial z} = -\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} \tag{2.76}$$

The boundary condition on the lower surface is $w = w_b$ with (see (1.32)):

$$w_b = u\frac{\partial b}{\partial x} + v\frac{\partial b}{\partial y} - \frac{\rho_w}{\rho_i}\left(-div\left(H\begin{bmatrix}\bar{u}\\ \bar{v}\end{bmatrix}\right)\right) \tag{2.77}$$

2.3.2 Basal velocity formulation

2.3.2.1 Basal velocity weak formulation

As usual, We define a kinematically admissible velocity field ϕ:

$$\phi(x,y,z) \in \{H^1(\Omega \cup \partial\Omega)\} \tag{2.78}$$

If one integrates the product of the equation (2.76) by a kinematically admissible velocity field:

$$\begin{aligned}\iint_{\Gamma_b\cup\Gamma_w} w_b\phi d\Gamma &= \iint_{\Gamma_b\cup\Gamma_w} u\frac{\partial b}{\partial x}\phi + v\frac{\partial b}{\partial y}\phi d\Gamma + \iint_{\Gamma_b} -\dot{M}_b\phi d\Gamma \\ &+ \iint_{\Gamma_w} -\frac{\rho_w}{\rho_i}\left(-div\left(H\begin{bmatrix}\bar{u}\\ \bar{v}\end{bmatrix}\right)\right)\phi - \frac{\rho_w}{\rho_i}\dot{M}_a\phi + \left(\frac{\rho_w}{\rho_i}-1\right)\dot{M}_b\phi d\Gamma\end{aligned} \tag{2.79}$$

2.3.2.2 Elementary stiffness matrix and load vector

Following the same method as in the previous situations, we obtain the stiffness matrix as follows:

$$\begin{aligned}K_{i,j} &= \iint_{\Gamma_b\cup\Gamma_w} \phi_j\phi_i d\Gamma \\ F_i &= \iint_{\Gamma_b\cup\Gamma_w} u\frac{\partial b}{\partial x}\phi_i + v\frac{\partial b}{\partial y}\phi_i d\Gamma + \iint_{\Gamma_b} -\dot{M}_b\phi_i d\Gamma \\ &+ \iint_{\Gamma_w} -\frac{\rho_w}{\rho_i}\left(-div\left(H\begin{bmatrix}\bar{u}\\ \bar{v}\end{bmatrix}\right)\right)\phi_i - \frac{\rho_w}{\rho_i}\dot{M}_a\phi_i + \left(\frac{\rho_w}{\rho_i}-1\right)\dot{M}_b\phi_i d\Gamma\end{aligned} \tag{2.80}$$

We now decompose these two matrices over the reference elements and use gaussian points for the integration :

$$\begin{aligned}&1 \le i \le 6 \text{ and } 1 \le j \le 6 \\ K_{i,j} &= \iint_{\hat{E}} L_j L_i |J| d\hat{\Gamma} \\ &\simeq \sum_{g=1}^{3} W_g L_j(g) L_i(g) |J_g|\end{aligned} \tag{2.81}$$

This elementary stiffness matrix is calculated with three matrices: $K1 = \sum_{g=1}^{3}[B]^T[D][B]$ as follows:

$$[B] = \left[\begin{array}{ccc} B_1 & B_2 & B_3 \end{array}\right] \quad \text{and} \quad D = W_g|J_g| \tag{2.82}$$

with:

$$B_i = L_i(g) \tag{2.83}$$

$$1 \leq i \leq 6$$

$$\begin{aligned}
F_i &= \iint_{\hat{E}} u\frac{\partial b}{\partial x}L_i + v\frac{\partial b}{\partial y}L_i|J|d\hat{\Gamma} + \iint_{\hat{E}\cap\Gamma_b} -\dot{M}_b L_i|J|d\Gamma \\
&+ \iint_{\hat{E}\cap\Gamma_w} -\frac{\rho_w}{\rho_i}\left(-div\left(H\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right)\right)L_i - \frac{\rho_w}{\rho_i}\dot{M}_a L_i + \left(\frac{\rho_w}{\rho_i}-1\right)\dot{M}_b L_i|J|d\Gamma \\
&\simeq \sum_{g=1}^{3} u\frac{\partial b}{\partial x}L_i(g) + v\frac{\partial b}{\partial y}L_i(g)|J_g| + \underbrace{\sum_{g=1}^{3} -\dot{M}_b L_i(g)|J_g|)}_{\text{ice shelf}} \\
&+ \sum_{g=1}^{3} -\frac{\rho_w}{\rho_i}\left(-div\left(H\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right)\right)L_i(g) - \frac{\rho_w}{\rho_i}\dot{M}_a L_i(g) + \underbrace{\left(\frac{\rho_w}{\rho_i}-1\right)\dot{M}_b L_i(g)|J_g|}_{\text{ice sheet}}
\end{aligned} \tag{2.84}$$

2.3.3 Vertical velocity weak formulation

As usual, We define a kinematically admissible velocity field ϕ:

$$\phi(x,y,z) \in \{H^1(\Omega \cup \partial\Omega) \quad \backslash \quad \phi(x,y,z) = 0 \qquad (x,y,z) \in \Gamma_b\} \tag{2.85}$$

If one integrates the product of the equation (2.76) by a kinematically admissible velocity field:

$$\iiint_\Omega \frac{\partial w}{\partial z}\phi d\Omega = -\iiint_\Omega \frac{\partial u}{\partial x}\phi + \frac{\partial v}{\partial y}\phi d\Omega \tag{2.86}$$

One can integrate by parts the left handside of the previous equation:

$$\iiint_\Omega w\frac{\partial \phi}{\partial z}d\Omega = \iiint_\Omega \frac{\partial u}{\partial x}\phi + \frac{\partial v}{\partial y}\phi d\Omega + \iiint_{\partial\Omega} w\phi n_z dS \tag{2.87}$$

Where $\overrightarrow{n} = (n_x, n_y, n_z)$ is the unit normal vector pointing outward from the glacier. Since we work on a vertically extruded mesh, n_z is equal to zero except on the upper and lower surface $\Gamma_b \cup \Gamma_s$. The function ϕ is cinematically admissible and is hence equal to zero on the lower surface Γ_b:

$$\iiint_\Omega w\frac{\partial \phi}{\partial z}d\Omega = \iiint_\Omega \frac{\partial u}{\partial x}\phi + \frac{\partial v}{\partial y}\phi d\Omega + \iint_{\Gamma_s} w\phi n_z dS \tag{2.88}$$

The term $\iint_{\Gamma_s} w\phi n_z dS$ is unknown. Therefore, we put it in the left handside of the weak formulation:

$$\iiint_\Omega w\frac{\partial \phi}{\partial z}d\Omega - \iint_{\Gamma_s} w\phi n_z dS = \iiint_\Omega \frac{\partial u}{\partial x}\phi + \frac{\partial v}{\partial y}\phi d\Omega \tag{2.89}$$

2.3.4 Finite element discretisation, Galerkin method

The domain is meshed using pentahedrons. There are a *nel* elements and *nods* grids. We use the nodal functions of each grid, which are continuous on the domain and affine on each element. We now project the unknown w on the nodal functions:

$$(\phi_i)_{1\leq i\leq nods}$$

$$w\left(x,y,z\right)=\sum_{i=1}^{nods} w_i\phi_i\left(x,y,z\right) \tag{2.90}$$

We build a vector W that holds all the unknows of the velocity field:

$$U=\left[\begin{array}{c} w_1 \\ ... \\ w_{nods} \end{array}\right] \tag{2.91}$$

With this new field, we replace the component of the virtual function ϕ by the nodal functions:

$$\forall 1\leq i\leq nods$$

$$\sum_{j=1}^{nods}\iiint_\Omega w_j\phi_j\frac{\partial\phi_i}{\partial z}d\Omega-\iint_{\Gamma_s} w_j\phi_j\phi_i n_z dS=\iiint_\Omega \frac{\partial u}{\partial x}\phi_i+\frac{\partial v}{\partial y}\phi_i d\Omega \tag{2.92}$$

We see the stiffness matrix K and load vector F appearing, so that the previous equations can be written as a system matrix :

$$\left[K\right]W=F \tag{2.93}$$

We use the following index:

$$\left[\begin{array}{ccc} & \vdots & \\ \cdots & K & \cdots \\ & \vdots & \end{array}\right]\left[\begin{array}{c}\vdots \\ w_j \\ \vdots\end{array}\right]=\left[\begin{array}{c}\vdots \\ F_j \\ \vdots\end{array}\right] \tag{2.94}$$

And we have the following expressions for the stiffness matrix and the load vector :

$$\begin{aligned} K_{i,j} &= \iiint_\Omega \phi_j\frac{\partial\phi_i}{\partial z}d\Omega-\iint_{\Gamma_s}\phi_j\phi_i n_z dS \\ F_i &= \iiint_\Omega \frac{\partial u}{\partial x}\phi_i+\frac{\partial v}{\partial y}\phi_i d\Omega \end{aligned} \tag{2.95}$$

2.3.5 Elementary stiffness matrix

To compute the elementary stiffness matrix, we first remove the integral over the upper surface, that will create a first elementary stiffness matrix $K1$, the upper surface integral will constitute another stiffness matrix $K2$. The elementary stiffness matrix will be the sum of $K1$ and $K2$ ($K_E=K1+K2$).

2.3.5.1 Without the upper surface

This matrix has the following shape:

$$1\leq i\leq 6 \text{ and } 1\leq j\leq 6$$

$$K1_{i,j}=\iiint_E \phi_j\frac{\partial\phi_i}{\partial z}dV \tag{2.96}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix becomes:

$$1 \leq i \leq 6 \text{ and } 1 \leq j \leq 6$$

$$\begin{aligned} K1_{i,j} &= \iiint_{\hat{E}} L_j \frac{\partial L_i}{\partial \varphi_z} |J| d\hat{V} \\ &\simeq \sum_{g=1}^{6} W_g L_j(g) \left. \frac{\partial L_i}{\partial \varphi_z} \right|_g |J_g| \end{aligned} \tag{2.97}$$

This elementary stiffness matrix is calculated with three matrices: $K1 = \sum_{g=1}^{6} [B]^T [D][B']$ as follows:

$$[B] = \begin{bmatrix} B_1 & B_2 & B_3 & B_4 & B_5 & B_6 \end{bmatrix} \quad \text{and} \quad [B'] = \begin{bmatrix} B'_1 & B'_2 & B'_3 & B'_4 & B'_5 & B'_6 \end{bmatrix} \tag{2.98}$$

with:

$$B_i = \left. \frac{\partial L_i}{\partial \varphi_z} \right|_g \qquad B'_i = L_i(g) \tag{2.99}$$

and $D = W_g |J_g|$

2.3.5.2 Upper surface integral

The upper surface involves only the three upper nods on the top of the upper pentahedrons of the mesh (which are the last three grids of the pentahedron grids lists). The elements are triangles and there are only 3 nodal functions to take into account. Therefore, this stiffness matrix is 3×3.
The basal drag matrix has the following shape:

$$4 \leq i \leq 6 \text{ and } 4 \leq j \leq 6$$

$$K2_{i,j} = -\iint_{E \cup \Gamma_s} \phi_j \phi_i n_z dS \tag{2.100}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix becomes:

$$4 \leq i \leq 6 \text{ and } 4 \leq j \leq 6$$

$$\begin{aligned} K2_{i,j} &= -\iiint_{\hat{E} \cup \Gamma_s} L_j(g) Li(g) |J| d\hat{S} \\ &\simeq -\sum_{g=1}^{6} W_g L_j(g) L_i(g) |J_g| \end{aligned} \tag{2.101}$$

This elementary stiffness matrix is calculated with three matrices: $K1 = \sum_{g=1}^{6} [L]^T [D][L]$ as follows:

$$[L] = \begin{bmatrix} L_4 & L_5 & L_6 \end{bmatrix} \tag{2.102}$$

with:

$$L_i = Li(g) \tag{2.103}$$

and $D = -W_g |J_g|$

2.3.6 Elementary load vector

The elementary load vector has the following shape:

$$F_i = \iiint_E \frac{\partial}{\partial x}\left(\sum_{k=1}^{6} u_k \phi_k\right)\phi_i + \frac{\partial}{\partial y}\left(\sum_{k=1}^{6} v_k \phi_k\right)\phi_i dV \tag{2.104}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix becomes:

$$1 \leq i \leq 6$$

$$\begin{aligned} F_i &= \iiint_{\hat{E}} \frac{\partial}{\partial \varphi_x}\left(\sum_{k=1}^{6} u_k L_k\right) L_i + \frac{\partial}{\partial \varphi_y}\left(\sum_{k=1}^{6} v_k L_k\right) L_i d\hat{V} \\ &\simeq \sum_{g=1}^{6}\left(\frac{\partial}{\partial \varphi_x}\left(\sum_{k=1}^{6} u_k L_k\right)\bigg|_g L_i(g) + \frac{\partial}{\partial \varphi_y}\left(\sum_{k=1}^{6} v_k L_k\right)\bigg|_g L_i(g)\right)|J_g| \end{aligned} \tag{2.105}$$

2.4 Full Stokes Model

2.4.1 Geometry and notations

The geometry is exactly the same as in Pattyn's model.

2.4.2 Equations

In the first chapter, we had:

$$\left\{\begin{array}{l}\frac{\partial}{\partial x}\left(2\mu\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial z}+\mu\frac{\partial w}{\partial x}\right)-\frac{\partial P}{\partial x}=0\\ \frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial y}\left(2\mu\frac{\partial v}{\partial y}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial v}{\partial z}+\mu\frac{\partial w}{\partial y}\right)-\frac{\partial P}{\partial y}=0\\ \frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial z}+\mu\frac{\partial w}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial z}+\mu\frac{\partial w}{\partial y}\right)+\frac{\partial}{\partial z}\left(2\mu\frac{\partial w}{\partial z}\right)-\frac{\partial P}{\partial z}-\rho g=0\end{array}\right. \tag{2.106}$$

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0$$

The boundary conditions are: a dynamic boundary condition (Neumann) at the ice front, a friction law on the ice sheet, and single point constraint (Dirichlet) on the other borders of the domain Ω. Exactly as in Pattyn's formulation, it is more convenient to consider these specified velocities as nill. For the vertical basal velocity, we use a multiple points constraint. This process, similar to a pemalty method, is explained in section 5.3.

$$\left\{\begin{array}{ll}\text{Ice Front } \Gamma_i: & \sigma\overrightarrow{n}=F_i\overrightarrow{n}=\rho_w g\min\left(0,z\right)\overrightarrow{n}\\ \text{Upper surface } \Gamma_s: & \sigma\left(\overrightarrow{n}\right)=0\\ \text{Ice sheet base } \Gamma_b: & \left(\sigma\left(\overrightarrow{n}\right)\right)_x+p\,n_x=-K^2N_{eff}^r\|\overrightarrow{v}\|^{s-1}u=-\alpha^2u\\ & \left(\sigma\left(\overrightarrow{n}\right)\right)_y+p\,n_y=-K^2N_{eff}^r\|\overrightarrow{v}\|^{s-1}v=-\alpha^2v\\ & w=0\\ \text{Ice shelf base } \Gamma_w: & \sigma\overrightarrow{n}=F_w\overrightarrow{n}=\rho_w g b(x,y)\overrightarrow{n}\\ \text{Other borders } \Gamma_u: & u=v=w=0\end{array}\right. \tag{2.107}$$

2.4.3 Weak Formulation

we write the weak formulation to solve these equations using finite element method. We define a kinematically admissible velocity field $\overrightarrow{\phi}$ and pressure field P^* such that:

$$\overrightarrow{\phi}(x,y,z)\in\{\left(H^1(\Omega\cup\partial\Omega)\right)^3 \quad\backslash\quad \overrightarrow{\phi}(x,y,z)=0 \qquad (x,y,z)\in\Gamma_u\} \tag{2.108}$$

$$P^*(x,y,z)\in H^1(\Omega\cup\partial\Omega) \tag{2.109}$$

For any function of these virtual fields, one can take the scalar product of the equations with $\overrightarrow{\phi}=(\phi_x,0,0)$, $\overrightarrow{\phi}=(0,\phi_y,0)$ and $\overrightarrow{\phi}=(0,0,\phi_z)$ that are 3 other kinematically admissible velocity fields. One can also multiply the last equation by the virtual pressure field and integrate this product over the domain. This gives us:

$$
\begin{aligned}
&\iiint_{\Omega}\left[\frac{\partial}{\partial x}\left(2\mu\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial z}+\mu\frac{\partial w}{\partial x}\right)-\frac{\partial P}{\partial x}\right]\phi_x d\Omega=0\\
&\iiint_{\Omega}\left[\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial y}\left(2\mu\frac{\partial v}{\partial y}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial v}{\partial z}+\mu\frac{\partial w}{\partial y}\right)-\frac{\partial P}{\partial y}\right]\phi_y d\Omega=0\\
&\iiint_{\Omega}\left[\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial z}+\mu\frac{\partial w}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial z}+\mu\frac{\partial w}{\partial y}\right)+\frac{\partial}{\partial z}\left(2\mu\frac{\partial w}{\partial z}\right)-\frac{\partial P}{\partial z}-\rho g\right]\phi_z d\Omega=0\\
&\iiint_{\Omega}\left[-\frac{\partial u}{\partial x}-\frac{\partial v}{\partial y}-\frac{\partial w}{\partial z}\right]P^* d\Omega=0
\end{aligned}
\tag{2.110}
$$

NB: We change the sign of the incompressibility equation because otherwise the stiffness matrix wouldn't be symmetric.
One intergrates by parts the first three equations in order to merge in the force on the ice front Γ_i and the basal friction Γ_b:

$$
\iiint_{\Omega}\left[\left(2\mu\frac{\partial u}{\partial x}\right)\frac{\partial \phi_x}{\partial x}+\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)\frac{\partial \phi_x}{\partial y}+\left(\mu\frac{\partial u}{\partial z}+\mu\frac{\partial w}{\partial x}\right)\frac{\partial \phi_x}{\partial z}-P\frac{\partial \phi_x}{\partial x}\right]d\Omega = \iint_{\Gamma_i\cup\Gamma_b} F n_x \phi_x d\Gamma \tag{2.111}
$$

$$
\iiint_{\Omega}\left[\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)\frac{\partial \phi_y}{\partial x}+\left(2\mu\frac{\partial v}{\partial y}\right)\frac{\partial \phi_y}{\partial y}+\left(\mu\frac{\partial v}{\partial z}+\mu\frac{\partial w}{\partial y}\right)\frac{\partial \phi_y}{\partial z}-P\frac{\partial \phi_y}{\partial y}\right]d\Omega = \iint_{\Gamma_i\cup\Gamma_b} F n_y \phi_y d\Gamma \tag{2.112}
$$

$$
\iiint_{\Omega}\left[\left(\mu\frac{\partial u}{\partial z}+\mu\frac{\partial w}{\partial x}\right)\frac{\partial \phi_z}{\partial x}+\left(\mu\frac{\partial v}{\partial z}+\mu\frac{\partial w}{\partial y}\right)\frac{\partial \phi_z}{\partial y}+\left(2\mu\frac{\partial w}{\partial z}\right)\frac{\partial \phi_z}{\partial z}-P\frac{\partial \phi_z}{\partial z}\right]d\Omega = \iint_{\Gamma_i} F n_z \phi_z d\Gamma - \iiint_{\Omega}\rho g \phi_z d\Omega \tag{2.113}
$$

It is easier to put the basal friction at the left hand side of the equation since it is velocity dependant. The first two equations become:

$$
\iiint_{\Omega}\left[\left(2\mu\frac{\partial u}{\partial x}\right)\frac{\partial \phi_x}{\partial x}+\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)\frac{\partial \phi_x}{\partial y}+\left(\mu\frac{\partial u}{\partial z}+\mu\frac{\partial w}{\partial x}\right)\frac{\partial \phi_x}{\partial z}-P\frac{\partial \phi_x}{\partial x}\right]d\Omega + \iint_{\Gamma_b}\alpha^2 u\phi_x d\Gamma = \iint_{\Gamma_i} F n_x \phi_x d\Gamma \tag{2.114}
$$

$$
\iiint_{\Omega}\left[\left(\mu\frac{\partial u}{\partial y}+\mu\frac{\partial v}{\partial x}\right)\frac{\partial \phi_y}{\partial x}+\left(2\mu\frac{\partial v}{\partial y}\right)\frac{\partial \phi_y}{\partial y}+\left(\mu\frac{\partial v}{\partial z}+\mu\frac{\partial w}{\partial y}\right)\frac{\partial \phi_y}{\partial z}-P\frac{\partial \phi_y}{\partial y}\right]d\Omega + \iint_{\Gamma_b}\alpha^2 v\phi_y d\Gamma = \iint_{\Gamma_i} F n_y \phi_y d\Gamma \tag{2.115}
$$

2.4.4 Finite element discretisation, Galerkin method

The domain is meshed using MINI elements (See A.6 for more details). The pressure is evaluated on the pentahedron vertices (*nods*), and the velocity is evaluated on the same vertices and also in a grid located in the middle of each element ($nods + nel$). We use the nodal functions of each grid, which are continuous on the domain.
We now project the velocities on the classical nodal functions ϕ_j and on the bubble function ϕ_{b_j}, and the pressur on the classical nodal functions:

$$
\begin{aligned}
u(x,y,z) &= \sum_{i=1}^{nods} u_i\phi_i(x,y,z) + \sum_{j=1}^{nels} u_j\phi_{b_j}(x,y,z) = \sum_{j=1}^{nods+nels} u_j\phi_j(x,y,z) \\
v(x,y,z) &= \sum_{i=1}^{nods} v_i\phi_i(x,y,z) + \sum_{j=1}^{nels} v_j\phi_{b_j}(x,y,z) = \sum_{j=1}^{nods+nels} v_j\phi_j(x,y,z) \\
w(x,y,z) &= \sum_{i=1}^{nods} w_i\phi_i(x,y,z) + \sum_{j=1}^{nels} w_j\phi_{b_j}(x,y,z) = \sum_{j=1}^{nods+nels} w_j\phi_j(x,y,z) \\
P(x,y,z) &= \sum_{k=1}^{nods} P_k\phi_k(x,y,z)
\end{aligned}
\tag{2.116}
$$

The nodal functions ϕ_i are bubble functions if $i > nods$. We build a vector W that holds all the unknows of the velocity and pressure field (for the sake of simplicity, this vector holds the speed components first and then the pressure of each element. This is not exactly what is done in ICE since the stiffness matrix would be badly conditioned): :

$$
W = \begin{bmatrix} u_1 \\ v_1 \\ w_1 \\ \vdots \\ w_{nods} \\ P_{nods} \\ \vdots \\ P_{nods+nel} \end{bmatrix}
\tag{2.117}
$$

With these new unknows, one can replace the components of the virtual function $\overrightarrow{\phi}$ by any nodal functions (classical or bubble):

$$\forall 1 \le i \le nods + nel$$

$$
\begin{aligned}
&\sum_{j=1}^{nods+nel} \iiint_\Omega \left(2\mu u_j \frac{\partial \phi_j}{\partial x}\right)\frac{\partial \phi_i}{\partial x} + \left(\mu u_j \frac{\partial \phi_j}{\partial y} + \mu v_j \frac{\partial \phi_j}{\partial x}\right)\frac{\partial \phi_i}{\partial y} + \left(\mu u_j \frac{\partial \phi_j}{\partial z} + \mu w_j \frac{\partial \phi_j}{\partial x}\right)\frac{\partial \phi_i}{\partial z} d\Omega \\
&+ \sum_{j=1}^{nods+nel} \iint_{\Gamma_b} \alpha^2 u_j\phi_j\phi_i d\Gamma + \sum_{k=1}^{nods} \iint_{\Gamma_b} P_k\phi_k\phi_i n_x d\Gamma - \sum_{k=1}^{nods} \iiint_\Omega P_k\phi_k \frac{\partial \phi_i}{\partial x} d\Omega = \iint_{\Gamma_i} F n_x \phi_i d\Gamma
\end{aligned}
\tag{2.118}
$$

$$
\begin{aligned}
&\sum_{j=1}^{nods+nel} \iiint_\Omega \left(\mu u_j \frac{\partial \phi_j}{\partial y} + \mu v_j \frac{\partial \phi_j}{\partial x}\right)\frac{\partial \phi_i}{\partial x} + \left(2\mu v_j \frac{\partial \phi_j}{\partial y}\right)\frac{\partial \phi_i}{\partial y} + \left(\mu v_j \frac{\partial \phi_j}{\partial z} + \mu w_j \frac{\partial \phi_j}{\partial y}\right)\frac{\partial \phi_i}{\partial z} d\Omega \\
&+ \sum_{j=1}^{nods+nel} \iint_{\Gamma_b} \alpha^2 v_j\phi_j\phi_i d\Gamma + \sum_{k=1}^{nods} \iint_{\Gamma_b} P_k\phi_k\phi_i n_y d\Gamma - \sum_{k=1}^{nods} \iiint_\Omega P_k\phi_k \frac{\partial \phi_i}{\partial y} d\Omega = \iint_{\Gamma_i} F n_y \phi_i d\Gamma
\end{aligned}
\tag{2.119}
$$

$$\sum_{j=1}^{nods+nel}\iiint_\Omega\left(\mu u_j\frac{\partial\phi_j}{\partial z}+\mu w_j\frac{\partial\phi_j}{\partial x}\right)\frac{\partial\phi_i}{\partial x}+\left(\mu v_j\frac{\partial\phi_j}{\partial z}+\mu w_j\frac{\partial\phi_j}{\partial y}\right)\frac{\partial\phi_i}{\partial y}+\left(2\mu w_j\frac{\partial\phi_j}{\partial z}\right)\frac{\partial\phi_i}{\partial z}d\Omega$$
$$-\sum_{k=1}^{nods}\iiint_\Omega P_k\phi_k\frac{\partial\phi_i}{\partial z}d\Omega=\iint_{\Gamma_i}Fn_z\phi_i d\Gamma-\iiint_\Omega\rho g\phi_i d\Omega \quad (2.120)$$

By rearranging terms:

$$\forall 1\le i\le nods+nel$$

$$\sum_{j=1}^{nods+nel}u_j\left[\iiint_\Omega 2\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}+\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}+\mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}d\Omega+\iint_{\Gamma_b}\alpha^2\phi_j\phi_i d\Gamma\right]+v_j\left[\iiint_\Omega\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}d\Omega\right]$$
$$+w_j\left[\iiint_\Omega\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial z}d\Omega\right]+\sum_{k=1}^{nods}P_k\left[\iint_{\Gamma_b}\phi_k\phi_i n_x d\Gamma-\iiint_\Omega\phi_k\frac{\partial\phi_i}{\partial x}d\Omega\right]=\iint_{\Gamma_i}Fn_x\phi_i d\Gamma \quad (2.121)$$

$$\sum_{j=1}^{nods+nel}u_j\left[\iiint_\Omega\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x}d\Omega\right]+v_j\left[\iiint_\Omega\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}+2\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}+\mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}d\Omega+\iint_{\Gamma_b}\alpha^2\phi_j\phi_i d\Gamma\right]$$
$$+w_j\left[\iiint_\Omega\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial z}d\Omega\right]+\sum_{k=1}^{nods}P_k\left[\iint_{\Gamma_b}\phi_k\phi_i n_y d\Gamma-\iiint_\Omega\phi_k\frac{\partial\phi_i}{\partial z}d\Omega\right]=\iint_{\Gamma_i}Fn_y\phi_i d\Gamma \quad (2.122)$$

$$\sum_{j=1}^{nods+nel}\iiint_\Omega u_j\left[\iiint_\Omega\mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial x}d\Omega\right]+v_j\left[\iiint_\Omega\mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial y}d\Omega\right]$$
$$+w_j\left[\iiint_\Omega\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}+\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}+2\mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}d\Omega\right]-\sum_{k=1}^{nods}P_k\iiint_\Omega\phi_k\frac{\partial\phi_i}{\partial y}d\Omega=\iint_{\Gamma_i}Fn_z\phi_i d\Gamma-\iiint_\Omega\rho g\phi_i d\Omega \quad (2.123)$$

And the incompressibility gives an additional equation, with $P^*=\phi_l\left(x,y,z\right)$ ($1\le l\le nods$, no bubble functions):

$$\forall 1\le l\le nods$$

$$\sum_{j=1}^{nods+nel}\iiint_\Omega\left[-\frac{\partial\phi_j}{\partial x}u_j-\frac{\partial\phi_j}{\partial y}v_j-\frac{\partial\phi_j}{\partial z}w_j\right]\phi_l d\Omega=0 \quad (2.124)$$

We see the stiffness matrix K and load vector F appearing, so that the previous equation can be written as follows:

$$\begin{bmatrix} & \vdots & \vdots & \vdots & & \vdots & \\ \cdots & K_{3i-2,3j-2} & K_{3i-2,3j-1} & K_{3i-2,3j} & \cdots & K_{3i-2,3nods+k} & \cdots \\ \cdots & K_{3i-1,3j-2} & K_{3i-1,3j-1} & K_{3i-1,3j} & \cdots & K_{3i-1,3nods+k} & \cdots \\ \cdots & K_{3i,3j-2} & K_{3i,3j-1} & K_{3i,3j} & \cdots & K_{3i,3nods+k} & \cdots \\ & \vdots & \vdots & \vdots & & \vdots & \\ \cdots & K_{3nods+l,3j-2} & K_{3nods+l,3j-1} & K_{3nods+l,3j} & \cdots & K_{3nods+l,3nods+k} & \cdots \\ & \vdots & \vdots & \vdots & & \vdots & \end{bmatrix}\begin{bmatrix}\vdots\\ u_j\\ v_j\\ w_j\\ \vdots\\ P_k\\ \vdots\end{bmatrix}=\begin{bmatrix}\vdots\\ F_{3j-2}\\ F_{3j-1}\\ F_{3j}\\ \vdots\\ F_{3nods+k}\\ \vdots\end{bmatrix}$$

$$
\begin{aligned}
K_{3i-2,3j-2} &= \iiint_\Omega 2\mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x} + \mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y} + \mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}d\Omega + \iint_{\Gamma_b}\alpha^2\phi_j\phi_i d\Gamma \\
K_{3i-2,3j-1} &= \iiint_\Omega \mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}d\Omega \\
K_{3i-2,3j} &= \iiint_\Omega \mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial z}d\Omega \\
K_{3i-2,3nods+k} &= \iint_{\Gamma_b}\phi_k\phi_i n_x d\Gamma + \iiint_\Omega -\phi_k\frac{\partial\phi_i}{\partial x}d\Omega \\
K_{3i-1,3j-2} &= \iiint_\Omega \mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x}d\Omega \\
K_{3i-1,3j-1} &= \iiint_\Omega \mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x} + 2\mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y} + \mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}d\Omega + \iint_{\Gamma_b}\alpha^2\phi_j\phi_i d\Gamma \\
K_{3i-1,3j} &= \iiint_\Omega \mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial z}d\Omega \\
K_{3i-1,3nods+k} &= \iint_{\Gamma_b}\phi_k\phi_i n_x d\Gamma + \iiint_\Omega -\phi_k\frac{\partial\phi_i}{\partial y}d\Omega \\
K_{3i,3j-2} &= \iiint_\Omega \mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial x}d\Omega \\
K_{3i,3j-1} &= \iiint_\Omega \mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial y}d\Omega \\
K_{3i,3j} &= \iiint_\Omega \mu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x} + \mu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y} + 2\mu\frac{\partial\phi_j}{\partial z}\frac{\partial\phi_i}{\partial z}d\Omega \\
K_{3i,3nods+k} &= \iiint_\Omega -\phi_k\frac{\partial\phi_i}{\partial z}d\Omega \\
K_{3nods+l,3j-2} &= \iiint_\Omega -\frac{\partial\phi_j}{\partial x}\phi_l d\Omega \\
K_{3nods+l,3j-1} &= \iiint_\Omega -\frac{\partial\phi_j}{\partial y}\phi_l d\Omega \\
K_{3nods+l,3j} &= \iiint_\Omega -\frac{\partial\phi_j}{\partial z}\phi_l d\Omega \\
K_{3nods+l,3nods+k} &= 0
\end{aligned}
\tag{2.125}
$$

And the load vector has the following expression:

$$
\begin{aligned}
F_{3i-2} &= \iint_{\Gamma_i} F n_x\phi_i d\Gamma \\
F_{3i-1} &= \iint_{\Gamma_i} F n_y\phi_i d\Gamma \\
F_{3i} &= \iint_{\Gamma_i} F n_z\phi_i d\Gamma - \iiint_\Omega \rho g\phi_i d\Omega \\
F_{3nods+k} &= 0
\end{aligned}
\tag{2.126}
$$

2.4.5 Decomposition over the elements and reference element

See the same section for Pattyn's model.

One element E gives 19 equations: we can take 6 classical nodal functions for $\phi_i(x, y, z)$ (the nodal functions corresponding to the vertexes of E, which is an pentahedron), each ϕ_i gives 4 equations (one for the x-component, one for the y-component, one for the z-component and one for the pressure) and there are 3 additional equations using the grid in the middle of the pentahedron (3 compenents of the velocity). Therefore, an elementary stiffness matrix K_E will be 27×27, and the elementary load vector will be of size 27.

2.4.6 Elementary stiffness matrix

To compute the elementary stiffness matrix, we first remove the basal drag, that will create a first elementary stiffness matrix $K1$, the basal drag will constitute another stiffness matrix $K2$. The elementary stiffness matrix will be the sum of $K1$ and $K2$ ($K_E = K1 + K2$).

2.4.6.1 Without basal drag

This matrix has the following shape:

$$1 \le i \le 7,\ 1 \le j \le 7,\ 1 \le k \le 6 \text{ and } 1 \le l \le 6$$

$$\begin{aligned}
K_{3i-2,3j-2} &= \iiint_E 2\mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x} + \mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y} + \mu \frac{\partial \phi_j}{\partial z}\frac{\partial \phi_i}{\partial z} d\Omega \\
K_{3i-2,3j-1} &= \iiint_E \mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial y} d\Omega \\
K_{3i-2,3j} &= \iiint_E \mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial z} d\Omega \\
K_{3i-2,k} &= \iiint_E -\phi_k \frac{\partial \phi_i}{\partial x} d\Omega
\end{aligned} \tag{2.127}$$

$$\begin{aligned}
K_{3i-1,3j-2} &= \iiint_E \mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial x} d\Omega \\
K_{3i-1,3j-1} &= \iiint_E \mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x} + 2\mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y} + \mu \frac{\partial \phi_j}{\partial z}\frac{\partial \phi_i}{\partial z} d\Omega \\
K_{3i-1,3j} &= \iiint_E \mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial z} d\Omega \\
K_{3i-1,k} &= \iiint_E -\phi_k \frac{\partial \phi_i}{\partial y} d\Omega
\end{aligned} \tag{2.128}$$

$$\begin{aligned}
K_{3i,3j-2} &= \iiint_E \mu \frac{\partial \phi_j}{\partial z}\frac{\partial \phi_i}{\partial x} d\Omega \\
K_{3i,3j-1} &= \iiint_E \mu \frac{\partial \phi_j}{\partial z}\frac{\partial \phi_i}{\partial y} d\Omega \\
K_{3i,3j} &= \iiint_E \mu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x} + \mu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y} + 2\mu \frac{\partial \phi_j}{\partial z}\frac{\partial \phi_i}{\partial z} d\Omega \\
K_{3i,k} &= \iiint_E -\frac{\partial \phi_i}{\partial z}\phi_k d\Omega \\
K_{l,3j-2} &= \iiint_E -\frac{\partial \phi_j}{\partial x}\phi_l d\Omega \\
K_{l,3j-1} &= \iiint_E -\frac{\partial \phi_j}{\partial y}\phi_l d\Omega \\
K_{l,3j} &= \iiint_E -\frac{\partial \phi_j}{\partial z}\phi_l d\Omega \\
K_{l,k} &= 0
\end{aligned} \tag{2.129}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix becomes:

$$1 \le i \le 7,\ 1 \le j \le 7,\ 1 \le k \le 6 \text{ and } 1 \le l \le 6$$

$$\begin{aligned}
K1_{3i-2,3j-2} &= \iiint_{\hat{E}} \left(2\mu \frac{\partial L_j}{\partial \varphi_x}\frac{\partial L_i}{\partial \varphi_x} + \mu \frac{\partial L_j}{\partial \varphi_y}\frac{\partial L_i}{\partial \varphi_y} + \mu \frac{\partial L_j}{\partial \varphi_z}\frac{\partial L_i}{\partial \varphi_z} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(2\mu \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g + \mu \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g + \mu \left.\frac{\partial L_j}{\partial \varphi_z}\right|_g \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \right) |J_g| \\
K1_{3i-2,3j-1} &= \iiint_{\hat{E}} \left(\mu \frac{\partial L_j}{\partial \varphi_x}\frac{\partial L_i}{\partial \varphi_y} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(\mu \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g \right) |J_g| \\
K1_{3i-2,3j} &= \iiint_{\hat{E}} \left(\mu \frac{\partial L_j}{\partial \varphi_x}\frac{\partial L_i}{\partial \varphi_z} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(\mu \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \right) |J_g| \\
K1_{3i-2,k} &= \iiint_{\hat{E}} \left(-L_k \frac{\partial L_i}{\partial \varphi_x} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(-L_k(g) \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g \right) |J_g| \\
K_{3i-1,3j-2} &= \iiint_{\hat{E}} \left(\mu \frac{\partial L_j}{\partial \varphi_y}\frac{\partial L_i}{\partial \varphi_x} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(\mu \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g \right) |J_g| \\
K_{3i-1,3j-1} &= \iiint_{\hat{E}} \left(\mu \frac{\partial L_j}{\partial \varphi_x}\frac{\partial L_i}{\partial \varphi_x} + 2\mu \frac{\partial L_j}{\partial \varphi_y}\frac{\partial L_i}{\partial \varphi_y} + \mu \frac{\partial L_j}{\partial \varphi_z}\frac{\partial L_i}{\partial \varphi_z} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(\mu \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g + 2\mu \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g + \mu \left.\frac{\partial L_j}{\partial \varphi_z}\right|_g \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \right) |J_g| \\
K_{3i-1,3j} &= \iiint_{\hat{E}} \left(\mu \frac{\partial L_j}{\partial \varphi_y}\frac{\partial L_i}{\partial \varphi_z} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(\mu \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \right) |J_g| \\
K_{3i-1,k} &= \iiint_{\hat{E}} \left(-L_k \frac{\partial L_i}{\partial \varphi_y} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(-L_k(g) \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g \right) |J_g|
\end{aligned}$$

$$
\begin{aligned}
K_{3i,3j-2} &= \iiint_{\hat{E}} \left(\mu \frac{\partial L_j}{\partial \varphi_z} \frac{\partial L_i}{\partial \varphi_x} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(\mu \left. \frac{\partial L_j}{\partial \varphi_z} \right|_g \left. \frac{\partial L_i}{\partial \varphi_x} \right|_g \right) |J_g| \\
K_{3i,3j-1} &= \iiint_{\hat{E}} \left(\mu \frac{\partial L_j}{\partial \varphi_z} \frac{\partial L_i}{\partial \varphi_y} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(\mu \left. \frac{\partial L_j}{\partial \varphi_z} \right|_g \left. \frac{\partial L_i}{\partial \varphi_y} \right|_g \right) |J_g| \\
K_{3i,3j} &= \iiint_{\hat{E}} \left(\mu \frac{\partial L_j}{\partial \varphi_x} \frac{\partial L_i}{\partial \varphi_x} + \mu \frac{\partial L_j}{\partial \varphi_y} \frac{\partial L_i}{\partial \varphi_y} + 2\mu \frac{\partial L_j}{\partial \varphi_z} \frac{\partial L_i}{\partial \varphi_z} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(\mu \left. \frac{\partial L_j}{\partial \varphi_x} \right|_g \left. \frac{\partial L_i}{\partial \varphi_x} \right|_g + \mu \left. \frac{\partial L_j}{\partial \varphi_y} \right|_g \left. \frac{\partial L_i}{\partial \varphi_y} \right|_g + 2\mu \left. \frac{\partial L_j}{\partial \varphi_z} \right|_g \left. \frac{\partial L_i}{\partial \varphi_z} \right|_g \right) |J_g| \\
K_{3i,k} &= \iiint_{\hat{E}} \left(-L_k \frac{\partial L_i}{\partial \varphi_z} \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(-L_k(g) \left. \frac{\partial L_i}{\partial \varphi_z} \right|_g \right) |J_g| \\
K_{l,3j-2} &= \iiint_{\hat{E}} \left(-\frac{\partial L_j}{\partial \varphi_x} L_l \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(- \left. \frac{\partial L_j}{\partial \varphi_x} \right|_g L_l(g) \right) |J_g| \\
K_{l,3j-1} &= \iiint_{\hat{E}} \left(-\frac{\partial L_j}{\partial \varphi_y} L_l \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(- \left. \frac{\partial L_j}{\partial \varphi_y} \right|_g L_l(g) \right) |J_g| \\
K_{l,3j} &= \iiint_{\hat{E}} \left(-\frac{\partial L_j}{\partial \varphi_z} L_l \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{3} W_g \left(- \left. \frac{\partial L_j}{\partial \varphi_z} \right|_g L_l(g) \right) |J_g| \\
K_{l,k} &= 0 \\
&\simeq \sum_{g=1}^{3} 0
\end{aligned}
$$

This elementary stiffness matrix is calculated with three matrices: $K1 = \sum_{g=1}^{3} [B]^T [D][B']$ as follows:

$$[B] = \begin{bmatrix} B_1 & B_2 & B_3 & B_4 & B_5 & B_6 & B_b \end{bmatrix} \quad \text{and} \quad [B'] = \begin{bmatrix} B'_1 & B'_2 & B'_3 & B'_4 & B'_5 & B'_6 & B_b \end{bmatrix}$$

with:

$$[B_i] = \begin{bmatrix} \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & 0 & 0 & 0 \\ 0 & \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & 0 & 0 \\ 0 & 0 & \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & 0 \\ \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & 0 & 0 \\ \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & 0 & \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & 0 \\ 0 & \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & \frac{1}{2}\left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & 0 \\ 0 & 0 & 0 & L_i(g) \\ \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & 0 \end{bmatrix} \quad \text{and} \quad [B_b] = \begin{bmatrix} \left.\frac{\partial L_b}{\partial \varphi_x}\right|_g & 0 & 0 \\ 0 & \left.\frac{\partial L_b}{\partial \varphi_y}\right|_g & 0 \\ 0 & 0 & \left.\frac{\partial L_b}{\partial \varphi_z}\right|_g \\ \frac{1}{2}\left.\frac{\partial L_b}{\partial \varphi_y}\right|_g & \frac{1}{2}\left.\frac{\partial L_b}{\partial \varphi_x}\right|_g & 0 \\ \frac{1}{2}\left.\frac{\partial L_b}{\partial \varphi_z}\right|_g & 0 & \frac{1}{2}\left.\frac{\partial L_b}{\partial \varphi_x}\right|_g \\ 0 & \frac{1}{2}\left.\frac{\partial L_b}{\partial \varphi_z}\right|_g & \frac{1}{2}\left.\frac{\partial L_b}{\partial \varphi_y}\right|_g \\ 0 & 0 & 0 \\ \left.\frac{\partial L_b}{\partial \varphi_x}\right|_g & \left.\frac{\partial L_b}{\partial \varphi_y}\right|_g & \left.\frac{\partial L_b}{\partial \varphi_z}\right|_g \end{bmatrix}$$

$$[B'_i] = \begin{bmatrix} \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & 0 & 0 & 0 \\ 0 & \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & 0 & 0 \\ 0 & 0 & \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & 0 \\ \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & 0 & 0 \\ \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & 0 & \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & 0 \\ 0 & \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & 0 \\ \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g & \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g & \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g & 0 \\ 0 & 0 & 0 & L_i(g) \end{bmatrix} \quad \text{and} \quad [B'_b] = \begin{bmatrix} \left.\frac{\partial L_b}{\partial \varphi_x}\right|_g & 0 & 0 \\ 0 & \left.\frac{\partial L_b}{\partial \varphi_y}\right|_g & 0 \\ 0 & 0 & \left.\frac{\partial L_b}{\partial \varphi_z}\right|_g \\ \left.\frac{\partial L_b}{\partial \varphi_y}\right|_g & \left.\frac{\partial L_b}{\partial \varphi_x}\right|_g & 0 \\ \left.\frac{\partial L_b}{\partial \varphi_z}\right|_g & 0 & \left.\frac{\partial L_b}{\partial \varphi_x}\right|_g \\ 0 & \left.\frac{\partial L_b}{\partial \varphi_z}\right|_g & \left.\frac{\partial L_b}{\partial \varphi_y}\right|_g \\ \left.\frac{\partial L_b}{\partial \varphi_x}\right|_g & \left.\frac{\partial L_b}{\partial \varphi_y}\right|_g & \left.\frac{\partial L_b}{\partial \varphi_z}\right|_g \\ 0 & 0 & 0 \end{bmatrix}$$

and

$$D = W_g|J_g| \begin{bmatrix} 2\mu & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2\mu & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2\mu & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2\mu & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{2.130}$$

NB: in $ISSM$ code, the factor 2 in the D matrix does not exist since the constitutive relation used is $\sigma' = \mu\varepsilon$ instead of $\sigma' = 2\mu\varepsilon$. This changes the viscosity definition that must be multiplied by 2.

2.4.6.2 Basal drag

As with Pattyn's, the elements are triangles and there are only 3 nodal functions to take into account. Therefore, the stiffness matrix that stands for basal drag is 9×9.
The basal drag stiffness matrix is such that:

$$\begin{aligned} & 1 \leq i \leq 3,\ 1 \leq j \leq 3 \text{ and } 1 \leq k \leq 3 \\ K_{3i-2,3j-2} &= \iint_{E\cap\Gamma_b} \alpha^2 \phi_j \phi_i d\Gamma \\ K_{3i-1,3j-1} &= \iint_{E\cap\Gamma_b} \alpha^2 \phi_j \phi_i d\Gamma \\ K_{3i-2,k} &= \iint_{E\cap\Gamma_b} \phi_k \phi_i n_x d\Gamma \\ K_{3i-1,k} &= \iint_{E\cap\Gamma_b} \phi_k \phi_i n_y d\Gamma \end{aligned} \tag{2.131}$$

The other terms are equal to zero.

Using the reference element $\hat{E}$ and the gaussian points, this matrix become:

$$\begin{aligned} K_{3i-2,3j-2} &= \iint_{\hat{E}\cap\hat{\Gamma_b}} \alpha^2 L_j(r,s) L_i(r,s)|J| d\hat{S} &\simeq \sum_{g=1}^{3} W_g \alpha^2 L_j(r_g,s_g) L_i(r_g,s_g)|J_g| \\ K_{3i-1,3j-1} &= \iint_{\hat{E}\cap\hat{\Gamma_b}} \alpha^2 L_j(r,s) L_i(r,s)|J| d\hat{S} &\simeq \sum_{g=1}^{3} W_g \alpha^2 L_j(r_g,s_g) L_i(r_g,s_g)|J_g| \\ K_{3i-2,k} &= \iint_{\hat{E}\cap\hat{\Gamma_b}} L_k(r,s) L_i(r,s)|J| n_x d\hat{S} &\simeq \sum_{g=1}^{3} W_g L_k(r_g,s_g) L_i(r_g,s_g)|J_g| n_x \\ K_{3i-1,k} &= \iint_{\hat{E}\cap\hat{\Gamma_b}} L_k(r,s) L_i(r,s)|J| n_y d\hat{S} &\simeq \sum_{g=1}^{3} W_g L_k(r_g,s_g) L_i(r_g,s_g)|J_g| n_y \end{aligned} \tag{2.132}$$

This matrix is calculated with three matrices: $K2 = \sum_{g=1}^{3} [L]^T [D_{drag}][L']$ as follows:

$$[L] = \begin{bmatrix} L_1 & L_2 & L_3 \end{bmatrix} \quad \text{and} \quad [L'] = \begin{bmatrix} L'_1 & L'_2 & L'_3 \end{bmatrix}$$

with:

$$[L_i] = \begin{bmatrix} L_i(r_g,s_g) & 0 & 0 & 0 \\ 0 & L_i(r_g,s_g) & 0 & 0 \\ L_i(r_g,s_g) & 0 & 0 & 0 \\ 0 & L_i(r_g,s_g) & 0 & 0 \end{bmatrix} \quad \text{and} \quad [L'_i] = \begin{bmatrix} L_i(r_g,s_g) & 0 & 0 & 0 \\ 0 & L_i(r_g,s_g) & 0 & 0 \\ 0 & 0 & 0 & L_i(r_g,s_g) \\ 0 & 0 & 0 & L_i(r_g,s_g) \end{bmatrix}$$

and

$$D = W_g |J_g| \begin{bmatrix} \alpha^2 & 0 & 0 & 0 \\ 0 & \alpha^2 & 0 & 0 \\ 0 & 0 & n_x & 0 \\ 0 & 0 & 0 & n_y \end{bmatrix} \tag{2.133}$$

2.4.7 Elementary load vector

To compute the elementary load vector, we first calculate the force due to the driving stress only, the water pressure only, and we then add these two terms.

2.4.7.1 Driving stress

The elementary load vector due to driving stress has the following shape:

$$F1_{3i} = -\iiint_{\Omega} \rho g \phi_i d\Omega \tag{2.134}$$

Using the reference element $\hat{E}$ and the gaussian points, this vector become:

$$F1_{3i} = -\iiint_{\hat{E}} \rho g L_i(r,s,\zeta)|J| d\hat{\Omega} \simeq \sum_{g=1}^{6} -W_g \rho g_i(r_g,s_g,\zeta_g)|J_g| \tag{2.135}$$

2.4.7.2 Ice front

The load on the ice front involves only the nodes on the ice/sea interface. As with basal drag, only surface integrals are required. The elements are quadrangle (See sectionA.3). The load on the ice front have the following shape:

$$\begin{aligned} F2_{3i-2} &= \iint_{E\cap\Gamma_i} F_i n_x \phi_i dS \\ F2_{3i-1} &= \iint_{E\cap\Gamma_i} F_i n_y \phi_i dS \\ F2_{3i} &= \iint_{E\cap\Gamma_i} F_i n_z \phi_i dS \end{aligned} \tag{2.136}$$

Using the reference element $\hat{E}$ and the gaussian points, this vector become:

$$
\begin{aligned}
F2_{3i-2} &= \iint_{\hat{E}\cup\hat{\Gamma}_i} \hat{F}_i n_x L_i |J| d\hat{S} &\simeq \sum_{g=1}^{4} W_g \rho_w g \min(0,z)\, n_x L_i(r_g,s_g,\zeta_g)|J_g| \\
F2_{3i-1} &= \iint_{\hat{E}\cup\hat{\Gamma}_i} \hat{F}_i n_y L_i |J| d\hat{S} &\simeq \sum_{g=1}^{4} W_g \rho_w g \min(0,z)\, n_y L_i(r_g,s_g,\zeta_g)|J_g| \\
F2_{3i} &= \iint_{\hat{E}\cup\hat{\Gamma}_i} \hat{F}_i n_z L_i |J| d\hat{S} &\simeq \sum_{g=1}^{4} W_g \rho_w g \min(0,z)\, n_z L_i(r_g,s_g,\zeta_g)|J_g|
\end{aligned}
\tag{2.137}
$$

2.4.7.3 Ice/water interface

The load on the ice/water interface involves only the nodes at the base on the iceshelf. As with basal drag and ice front, only surface integrals are required. The elements are triangles. The load on the ice shelf base have the following shape:

$$
\begin{aligned}
F3_{3i-2} &= \iint_{E\cap\Gamma_w} F_w n_x \phi_i dS \\
F3_{3i-1} &= \iint_{E\cap\Gamma_w} F_w n_y \phi_i dS \\
F3_{3i} &= \iint_{E\cap\Gamma_w} F_w n_z \phi_i dS
\end{aligned}
\tag{2.138}
$$

Using the reference element $\hat{E}$ and the gaussian points, this vector become:

$$
\begin{aligned}
F3_{2i-1} &= \iint_{\hat{E}\cup\hat{\Gamma}_w} \hat{F}_w n_x L_i |J| d\hat{S} &\simeq \sum_{g=1}^{3} W_g \rho_w g b(r_g,s_g) n_x L_i(r_g,s_g)|J_g| \\
F3_{2i} &= \iint_{\hat{E}\cup\hat{\Gamma}_w} \hat{F}_w n_y L_i |J| d\hat{S} &\simeq \sum_{g=1}^{3} W_g \rho_w g b(r_g,s_g) n_y L_i(r_g,s_g)|J_g| \\
F3_{2i} &= \iint_{\hat{E}\cup\hat{\Gamma}_w} \hat{F}_w n_y L_i |J| d\hat{S} &\simeq \sum_{g=1}^{3} W_g \rho_w g b(r_g,s_g) n_y L_i(r_g,s_g)|J_g|
\end{aligned}
\tag{2.139}
$$

2.4.8 Collapsing of the seventh grid

In order to use the same mesh for all the 3d models, we have to get rid of the grid located on the middle of the pentahedron. To do so, we have to collapse this grid and reduce the elementary matrix and load vector to the first six grids.

The elementary equation before collapsing is:

$$\left[\begin{array}{c|c} K_{ii} & K_{ib} \\ \hline K_{bi} & K_{bb} \end{array}\right] \left[\begin{array}{c} W_i \\ \hline W_b \end{array}\right] = \left[\begin{array}{c} F_i \\ \hline F_b \end{array}\right] \tag{2.140}$$

where index b relates to the seventh grid and index i refers to the first six grids. W contains all the unknows of the element, ie u, v, w, and p. So that W_i has 24 components and W_b has 3 components (there is no pressure on the seventh grid).

We separate this system of equations in two parts as follows:

$$\left\{ \begin{array}{l} K_{ii}W_i + K_{ib}W_b = F_i \\ \\ K_{bi}W_i + K_{bb}W_b = F_b \end{array} \right. \tag{2.141}$$

The second equation gives us $W_b = K_{bb}^{-1} \left(F_b - K_{bi}W_i\right)$ so that we can replace W_b on the first equation. We now have a linear system on the first six grids. So we do solve the system on these six grids and use the same mesh as in the other 3d computations.

The system to solve becomes:

$$\boxed{\left(K_{ii} - K_{ib}K_{bb}^{-1}K_{bi}\right) W_i = F_i - K_{ib}K_{bb}^{-1}F_b} \tag{2.142}$$

Chapter 3

Thermal Model Finite Element Formulation

The thermal equation (1.65) involves a transient term $\frac{\partial T}{\partial t}$. This term can be neglected if the glacier is considered in a steady state. If it is not the case, this term must be taken into account and finite differences time domain scheme is required.

3.1 Steady state

3.1.1 Geometry and notations

The geometry is exactly the same as in the 3d models. We adopt the following notations:

- Ω is the open surface that constitutes the entire glacier system
- $\partial\Omega$ are points located at the boundary
- Γ_b are points at the interface between glacier and bedrock (Ice/Till interface $z = b$)
- Γ_w are points at the interface between glacier and water.

3.1.2 Equations

For a steady state, one can neglect $\frac{\partial T}{\partial t}$. Given this assumption, the thermal equation (1.65) is reduced to:

$$u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y} + w\frac{\partial T}{\partial z} = \frac{k}{\rho c}\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2}\right) + \frac{\Phi}{\rho c} \tag{3.1}$$

For the temperature computation, the melting is not taken into account. When the pressure melting point is reached, one will use penalty method to constrain this point to this temperature. Thus, the boundary conditions are limited to: an imposed flux (Neumann) at the base of the grouded part, an imposed temperature (Dirichlet) on the lower surface of the floating part. The sides of the domain $\Gamma_u \cup \Gamma_i$ are treated as zero-flux surfaces. It is more convenient to consider the imposed temperatures as nill. At the end of the process, the temperature field will be updated to take these constraints into account.

$$\begin{cases} \text{Icesheet base } \Gamma_b: & k\left(\overrightarrow{grad}\, T\right)\Big|_b . \overrightarrow{n} = G + \overrightarrow{\tau}_b . \overrightarrow{u}_b \\ \\ \text{Iceshelf base } \Gamma_w: & k\left(\overrightarrow{grad}\, T\right)\Big|_b . \overrightarrow{n} = -\rho_w c_{pM}\gamma\left(T_b - T_{pmp}\right) \\ \\ \text{Other borders } \Gamma_u \cup \Gamma_i: & \left(\overrightarrow{grad}\, T\right) . \overrightarrow{n} = 0 \end{cases} \tag{3.2}$$

3.1.3 Weak Formulation

we write the weak formulation to solve the previous equation using finite element. One defines an admissible thermal field θ such that:

$$\theta(x,y,z) \in \{\left(H^1(\Omega \cup \partial\Omega)\right) \quad | \quad \theta(x,y,z) = 0 \qquad (x,y,z) \in \partial\Omega \setminus \Gamma_b\} \tag{3.3}$$

For any virtual field as described above:

$$\iiint_\Omega \left[u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y} + w\frac{\partial T}{\partial z} - \frac{k}{\rho c}\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2}\right) - \frac{\Phi}{\rho c}\right] \theta d\Omega = 0 \tag{3.4}$$

One intergrates by parts the conduction term in order to use the boundary conditions:

$$\iiint_\Omega \frac{k}{\rho c}\left(\frac{\partial T}{\partial x}\frac{\partial \theta}{\partial x} + \frac{\partial T}{\partial y}\frac{\partial \theta}{\partial y} + \frac{\partial T}{\partial z}\frac{\partial \theta}{\partial z}\right) d\Omega = \iiint_\Omega \left[-u\frac{\partial T}{\partial x} - v\frac{\partial T}{\partial y} - w\frac{\partial T}{\partial z} + \frac{\Phi}{\rho c}\right]\theta d\Omega$$

$$+ \iint_{\partial\Omega} \frac{k}{\rho c}\left(\frac{\partial T}{\partial x}\theta n_x + \frac{\partial T}{\partial y}\theta n_y + \frac{\partial T}{\partial z}\theta n_z\right) dS \tag{3.5}$$

Where $\overrightarrow{n} = (n_x, n_y, n_z)$ is the unit normal vector pointing outward from the glacier. Since θ is admissible, the last integrale is nill except on Γ_b, where one can use the boundary condition:

$$\iiint_\Omega \frac{k}{\rho c}\left(\frac{\partial T}{\partial x}\frac{\partial \theta}{\partial x} + \frac{\partial T}{\partial y}\frac{\partial \theta}{\partial y} + \frac{\partial T}{\partial z}\frac{\partial \theta}{\partial z}\right) d\Omega = \iiint_\Omega \left[-u\frac{\partial T}{\partial x} - v\frac{\partial T}{\partial y} - w\frac{\partial T}{\partial z} + \frac{\Phi}{\rho c}\right]\theta d\Omega$$

$$+ \iint_{\Gamma_w} \frac{\rho_w c_{pM} \gamma}{\rho c}\left(T_{pmp} - T\right)\theta dS + \iint_{\Gamma_b} \frac{1}{\rho c}\left(G + \overrightarrow{\tau}_b . \overrightarrow{u}_b\right)\theta dS \tag{3.6}$$

By rearranging terms:

$$\iiint_\Omega \frac{k}{\rho c}\left(\frac{\partial T}{\partial x}\frac{\partial \theta}{\partial x} + \frac{\partial T}{\partial y}\frac{\partial \theta}{\partial y} + \frac{\partial T}{\partial z}\frac{\partial \theta}{\partial z}\right) d\Omega + \iiint_\Omega u\frac{\partial T}{\partial x}\theta + v\frac{\partial T}{\partial y}\theta + w\frac{\partial T}{\partial z}\theta d\Omega$$

$$= \iiint_\Omega \frac{\Phi}{\rho c}\theta d\Omega + \iint_{\Gamma_w} \frac{\rho_w c_{pM} \gamma}{\rho c}\left(T_{pmp} - T\right)\theta dS + \iint_{\Gamma_b} \frac{1}{\rho c}\left(G + \overrightarrow{\tau}_b . \overrightarrow{u}_b\right)\theta dS \tag{3.7}$$

3.1.4 Finite element discretisation, Galerkin method

The domain is meshed using pentahedrons. There are a *nel* elements and *nods* grids. We use the nodal functions of each grid, which are continuous on the domain and affine on each element. We now project the unknown on the nodal functions:

$$(\phi_i)_{1 \leq i \leq nods}$$

$$T\left(x,y,z\right) = \sum_{i=1}^{nods} T_i \theta_i \left(x,y,z\right) \tag{3.8}$$

We build a vector T that holds all the unknowns:

$$T = \begin{bmatrix} T_1 \\ \ldots \\ T_{nods} \end{bmatrix} \tag{3.9}$$

With this new field, we have the following equations:

$$\forall 1 \leq i \leq nods$$

$$\sum_{j=1}^{nods} \iiint_\Omega \frac{k}{\rho c}\left(\frac{\partial T_j\theta_j}{\partial x}\frac{\partial \theta_i}{\partial x} + \frac{\partial T_j\theta_j}{\partial y}\frac{\partial \theta_i}{\partial y} + \frac{\partial T_j\theta_j}{\partial z}\frac{\partial \theta_i}{\partial z}\right) d\Omega + \iiint_\Omega u\frac{\partial T_j\theta_j}{\partial x}\theta_i + v\frac{\partial T_j\theta_j}{\partial y}\theta_i + w\frac{\partial T_j\theta_j}{\partial z}\theta_i d\Omega$$

$$= \iiint_\Omega \frac{\Phi}{\rho c}\theta_i d\Omega + \iint_{\Gamma_w} \frac{\rho_w c_{pM}\gamma}{\rho c}\left(T_{pmp} - \sum_{j=1}^{nods} T_j\theta_j\right)\theta_i dS + \iint_{\Gamma_b} \frac{1}{\rho c}\left(G + \overrightarrow{\tau}_b.\overrightarrow{u}_b\right)\theta_i dS \tag{3.10}$$

One can extract T_j:

$$\sum_{j=1}^{nods}\left[\iiint_\Omega \frac{k}{\rho c}\left(\frac{\partial \theta_j}{\partial x}\frac{\partial \theta_i}{\partial x} + \frac{\partial \theta_j}{\partial y}\frac{\partial \theta_i}{\partial y} + \frac{\partial \theta_j}{\partial z}\frac{\partial \theta_i}{\partial z}\right) d\Omega\right.$$

$$\left. + \iiint_\Omega u\frac{\partial \theta_j}{\partial x}\theta_i + v\frac{\partial \theta_j}{\partial y}\theta_i + w\frac{\partial \theta_j}{\partial z}\theta_i d\Omega + \iint_{\Gamma_w} \frac{\rho_w c_{pM}\gamma}{\rho c}\theta_j\theta_i dS\right] T_j$$

$$= \iiint_\Omega \frac{\Phi}{\rho c}\theta_i d\Omega + \iint_{\Gamma_w} \frac{\rho_w c_{pM}\gamma}{\rho c}T_{pmp}\theta_i dS + \iint_{\Gamma_b} \frac{1}{\rho c}\left(G + \overrightarrow{\tau}_b.\overrightarrow{u}_b\right)\theta_i dS$$

We see the stiffness matrix K and load vector F appearing, so that the previous equations can be written as a system matrix :

$$[K]\,T = F \tag{3.11}$$

We use the following index:

$$\begin{bmatrix} & \vdots & \\ \cdots & K_{ij} & \cdots \\ & \vdots & \end{bmatrix}\begin{bmatrix} \vdots \\ T_i \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ F_i \\ \vdots \end{bmatrix} \tag{3.12}$$

And we have the following expressions for the stiffness matrix and the load vector :

$$\begin{aligned} K_{i,j} &= \iiint_\Omega \frac{k}{\rho c}\left(\frac{\partial \theta_j}{\partial x}\frac{\partial \theta_i}{\partial x} + \frac{\partial \theta_j}{\partial y}\frac{\partial \theta_i}{\partial y} + \frac{\partial \theta_j}{\partial z}\frac{\partial \theta_i}{\partial z}\right) d\Omega \\ &+ \iiint_\Omega u\frac{\partial \theta_j}{\partial x}\theta_i + v\frac{\partial \theta_j}{\partial y}\theta_i + w\frac{\partial \theta_j}{\partial z}\theta_i d\Omega + \iint_{\Gamma_w} \frac{\rho_w c_{pM}\gamma}{\rho c}\theta_j\theta_i dS \\ F_i &= \iiint_\Omega \frac{\Phi}{\rho c}\theta_i d\Omega + \iint_{\Gamma_w} \frac{\rho_w c_{pM}\gamma}{\rho c}T_{pmp}\theta_i dS \\ &+ \iint_{\Gamma_b} \frac{1}{\rho c}\left(G + \overrightarrow{\tau}_b.\overrightarrow{u}_b\right)\theta_i dS \end{aligned} \tag{3.13}$$

3.1.5 Decomposition over the elements and reference element

See same section in Pattyn's Model. One element E gives 6 equations: we can take 6 nodal functions for $\theta_i(x, y, z)$ (the nodal functions corresponding to the vertexes of E, which is an pentahedron), each θ_i gives 1 equation. Therefore, an elementary stiffness matrix K_E will be 6×6, and the elementary load vector will be of size 6.

3.1.6 Elementary stiffness matrix

The elementary stiffness matrix is divided into 3 matrices: one conduction stiffness matrix, one advection stiffness matrix, and one ocean/ice heat exchange stiffness matrix.

3.1.6.1 Conduction stiffness matrix

This matrix has the following shape:

$$1 \le i \le 6 \text{ and } 1 \le j \le 6$$

$$K1_{i,j} = \iiint_E \frac{k}{\rho c}\left(\frac{\partial\theta_j}{\partial x}\frac{\partial\theta_i}{\partial x} + \frac{\partial\theta_j}{\partial y}\frac{\partial\theta_i}{\partial y} + \frac{\partial\theta_j}{\partial z}\frac{\partial\theta_i}{\partial z}\right) dV \tag{3.14}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix becomes:

$$1 \le i \le 6 \text{ and } 1 \le j \le 6$$

$$\begin{aligned} K1_{i,j} &= \iiint_{\hat{E}} \left(\frac{k}{\rho c}\left(\frac{\partial L_j}{\partial\varphi_x}\frac{\partial L_i}{\partial\varphi_x} + \frac{\partial L_j}{\partial\varphi_y}\frac{\partial L_i}{\partial\varphi_y} + \frac{\partial L_j}{\partial\varphi_z}\frac{\partial L_i}{\partial\varphi_z}\right)\right)|J|d\hat{V} \\ &\simeq \sum_{g=1}^{6} W_g \left(\frac{k}{\rho c}\left(\left.\frac{\partial L_j}{\partial\varphi_x}\right|_g \left.\frac{\partial L_i}{\partial\varphi_x}\right|_g + \left.\frac{\partial L_j}{\partial\varphi_y}\right|_g \left.\frac{\partial L_i}{\partial\varphi_y}\right|_g + \left.\frac{\partial L_j}{\partial\varphi_z}\right|_g \left.\frac{\partial L_i}{\partial\varphi_z}\right|_g\right)\right)|J_g| \end{aligned}$$

This elementary stiffness matrix is calculated with three matrices: $K1 = \sum_{g=1}^{6}[B]^T[D][B]$ as follows:

$$[B] = \left[\begin{array}{cccccc} B_1 & B_2 & B_3 & B_4 & B_5 & B_6 \end{array}\right] \tag{3.15}$$

with:

$$B_i = \left[\begin{array}{c} \left.\frac{\partial L_i}{\partial\varphi_x}\right|_g \\ \\ \left.\frac{\partial L_i}{\partial\varphi_y}\right|_g \\ \\ \left.\frac{\partial L_i}{\partial\varphi_z}\right|_g \end{array}\right] \tag{3.16}$$

and

$$D = W_g|J_g|\frac{k}{\rho c} \tag{3.17}$$

3.1.6.2 Advection stiffness matrix

This matrix has the following shape:

$$1 \le i \le 6 \text{ and } 1 \le j \le 6$$

$$K2_{i,j} = \iiint_E u\frac{\partial\theta_j}{\partial x}\theta_i + v\frac{\partial\theta_j}{\partial y}\theta_i + w\frac{\partial\theta_j}{\partial z}\theta_i dV \tag{3.18}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix becomes:

$$1 \le i \le 6 \text{ and } 1 \le j \le 6$$

$$\begin{aligned} K2_{i,j} &= \iiint_{\hat{E}} \left(u \frac{\partial L_j}{\partial \varphi_x} L_i + v \frac{\partial L_j}{\partial \varphi_y} L_i + w \frac{\partial L_j}{\partial \varphi_z} L_i \right) |J| d\hat{V} \\ &\simeq \sum_{g=1}^{6} W_g \left(u\,(g) \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g L_i\,(g) + v\,(g) \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g L_i\,(g) + w\,(g) \left.\frac{\partial L_j}{\partial \varphi_z}\right|_g L_i\,(g) \right) |J_g| \end{aligned} \tag{3.19}$$

This elementary stiffness matrix is calculated with three matrices: $K2 = \sum_{g=1}^{6} [B]^T [D][B']$ as follows:

$$[B] = \left[\begin{array}{cccccc} B_1 & B_2 & B_3 & B_4 & B_5 & B_6 \end{array} \right] \tag{3.20}$$

with:

$$B_i = \left[\begin{array}{c} L_i(g) \\ \\ L_i(g) \\ \\ L_i(g) \end{array} \right] \quad B'_i = \left[\begin{array}{c} \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g \\ \\ \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g \\ \\ \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \end{array} \right] \tag{3.21}$$

and

$$D = W_g |J_g| \left[\begin{array}{ccc} \sum_{k=1}^{6} u_k L_k(g) & 0 & 0 \\ 0 & \sum_{k=1}^{6} v_k L_k(g) & 0 \\ 0 & 0 & \sum_{k=1}^{6} w_k L_k(g) \end{array} \right] \tag{3.22}$$

3.1.6.3 Ocean/ice heat exchange

This elementary stiffness has the following shape:

$$1 \leq i \leq 3 \text{ and } 1 \leq j \leq 3$$

$$K3_{i,j} = \iint_{\Gamma_w \cap E} \frac{\rho_w c_{pM} \gamma}{\rho c} \theta_j \theta_i dS \tag{3.23}$$

Using the reference element $\hat{E}$ and the gaussian points, this vector becomes:

$$1 \leq i \leq 6$$

$$\begin{aligned} K3_{i,j} &= \iint_{\Gamma_w \cap \hat{E}} \frac{\rho_w c_{pM} \gamma}{\rho c} L_j L_i |J| d\hat{S} \\ &\simeq \sum_{g=1}^{3} W_g |J_g| \frac{\rho_w c_{pM} \gamma}{\rho c} L_j(g) L_i(g) \end{aligned} \tag{3.24}$$

This elementary stiffness matrix is calculated with three matrices: $K3 = \sum_{g=1}^{3} [B]^T [D][B]$ as follows:

$$[B] = \left[\begin{array}{ccc} B_1 & B_2 & B_3 \end{array} \right] \quad \text{and} \quad D = W_g |J_g| \frac{\rho_w c_{pM} \gamma}{\rho c} \tag{3.25}$$

with:

$$B_i = L_i(g) \tag{3.26}$$

3.1.7 Elementary load vector

3.1.7.1 Deformational heating

This elementary load vector has the following shape:

$$1 \leq i \leq 6$$

$$F_i = \iiint_E \frac{\Phi}{\rho c} \theta_i d\Omega dV \tag{3.27}$$

Using the reference element $\hat{E}$ and the gaussian points, this vector becomes:

$$1 \leq i \leq 6$$

$$F_i = \iiint_{\hat{E}} \frac{\Phi}{\rho c} L_i |J| d\hat{V} = \sum_{g=1}^{6} W_g |J_g| \frac{\Phi(g)}{\rho c} L_i(g) \tag{3.28}$$

3.1.7.2 Basal heating

This elementary load vector has the following shape:

$$1 \leq i \leq 6$$

$$F_i = \iint_{\Gamma_b \cap E} \frac{1}{\rho c} \left(G + \overrightarrow{\tau}_b . \overrightarrow{u}_b\right) \theta_i dS \tag{3.29}$$

Using the reference element $\hat{E}$ and the gaussian points, this vector becomes:

$$1 \leq i \leq 6$$

$$\begin{aligned} F_i &= \iint_{\Gamma_b \cap \hat{E}} \frac{1}{\rho c} \left(G + \overrightarrow{\tau}_b . \overrightarrow{u}_b\right) L_i |J| d\hat{S} \\ &\simeq \sum_{g=1}^{3} W_g |J_g| \frac{1}{\rho c} \left(G(g) + \overrightarrow{\tau}_b(g) . \overrightarrow{u}_b(g)\right) L_i(g) \end{aligned} \tag{3.30}$$

Only the nodal functions assigned to the nodes on Γ_b are involved.

3.1.7.3 Ocean/ice heat exchange

This elementary load vector has the following shape:

$$1 \leq i \leq 6$$

$$F_i = \iint_{\Gamma_w \cap E} \frac{\rho_w c_{pM} \gamma}{\rho c} T_{pmp} \theta_i dS \tag{3.31}$$

Using the reference element $\hat{E}$ and the gaussian points, this vector becomes:

$$1 \leq i \leq 6$$

$$\begin{aligned} F_i &= \iint_{\Gamma_w \cap \hat{E}} \frac{\rho_w c_{pM} \gamma}{\rho c} T_{pmp} L_i |J| d\hat{S} \\ &\simeq \sum_{g=1}^{3} W_g |J_g| \frac{\rho_w c_{pM} \gamma}{\rho c} T_{pmp}(g) L_i(g) \end{aligned} \tag{3.32}$$

Only the nodal functions assigned to the nodes on Γ_w are involved.

3.2 Transient

3.2.1 Equations

This time, one cannot neglect $\frac{\partial T}{\partial t}$. The equation is the same as in the first chapter (1.65):

$$\frac{\partial T}{\partial t} = -\left(u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y} + w\frac{\partial T}{\partial z}\right) + \frac{k}{\rho c}\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2}\right) + \frac{\Phi}{\rho c} \tag{3.33}$$

The boundary conditions are the same as in the previous chapter.

3.2.2 Weak Formulation

The weak formulation is the same as in the previoous chapter with the transient term:

$$\iiint_\Omega \frac{k}{\rho c}\left(\frac{\partial T}{\partial x}\frac{\partial \theta}{\partial x} + \frac{\partial T}{\partial y}\frac{\partial \theta}{\partial y} + \frac{\partial T}{\partial z}\frac{\partial \theta}{\partial z}\right) d\Omega + \iiint_\Omega u\frac{\partial T}{\partial x}\theta + v\frac{\partial T}{\partial y}\theta + w\frac{\partial T}{\partial z}\theta d\Omega$$

$$= \iiint_\Omega \left(-\frac{\partial T}{\partial t} + \frac{\Phi}{\rho c}\right)\theta d\Omega + \iint_{\Gamma_b} \frac{1}{\rho c}\left(G + \overrightarrow{\tau}_b.\overrightarrow{u}_b\right)\theta dS \tag{3.34}$$

3.2.3 Finite difference scheme

To evaluate the transient term, one uses a finite differences time domain scheme. For a given time step i, the transient term is computed implicitly[1] as follow:

$$\left.\frac{\partial T}{\partial t}\right|_i = \frac{T_i - T_{i-1}}{\Delta t} \tag{3.35}$$

Where Δt is the time difference between t_i and t_{i-1}. The weak formuation becomes:

$$\iiint_\Omega \frac{k}{\rho c}\left(\frac{\partial T_i}{\partial x}\frac{\partial \theta}{\partial x} + \frac{\partial T_i}{\partial y}\frac{\partial \theta}{\partial y} + \frac{\partial T_i}{\partial z}\frac{\partial \theta}{\partial z}\right) d\Omega + \iiint_\Omega u\frac{\partial T_i}{\partial x}\theta + v\frac{\partial T_i}{\partial y}\theta + w\frac{\partial T_i}{\partial z}\theta d\Omega$$

$$= \iiint_\Omega \left(-\frac{T_i - T_{i-1}}{\Delta t} + \frac{\Phi}{\rho c}\right)\theta d\Omega + \iint_{\Gamma_b} \frac{1}{\rho c}\left(G + \overrightarrow{\tau}_b.\overrightarrow{u}_b\right)\theta dS \tag{3.36}$$

By rearranging terms:

$$\iiint_\Omega T_i\theta d\Omega + \Delta t \iiint_\Omega \frac{k}{\rho c}\left(\frac{\partial T_i}{\partial x}\frac{\partial \theta}{\partial x} + \frac{\partial T_i}{\partial y}\frac{\partial \theta}{\partial y} + \frac{\partial T_i}{\partial z}\frac{\partial \theta}{\partial z}\right) d\Omega + \Delta t \iiint_\Omega u\frac{\partial T_i}{\partial x}\theta + v\frac{\partial T_i}{\partial y}\theta + w\frac{\partial T_i}{\partial z}\theta d\Omega$$

$$= \iiint_\Omega T_{i-1}\theta d\Omega + \Delta t \iiint_\Omega \frac{\Phi}{\rho c}\theta d\Omega + \Delta t \iint_{\Gamma_b} \frac{1}{\rho c}\left(G + \overrightarrow{\tau}_b.\overrightarrow{u}_b\right)\theta dS \tag{3.37}$$

[1] an implicit method is much more stable for this kind of equation

3.2.4 Elementary stiffness matrix

The elementary stiffness matrix is very similar to the one of the steady state case: all the previous matrices are multiplied by Δt, and a transient term is added:

3.2.4.1 Conduction stiffness matrix

This matrix has the following shape:

$$1 \leq i \leq 6 \text{ and } 1 \leq j \leq 6$$

$$K1_{i,j} = \Delta t \iiint_E \frac{k}{\rho c} \left(\frac{\partial \theta_j}{\partial x} \frac{\partial \theta_i}{\partial x} + \frac{\partial \theta_j}{\partial y} \frac{\partial \theta_i}{\partial y} + \frac{\partial \theta_j}{\partial z} \frac{\partial \theta_i}{\partial z} \right) dV \tag{3.38}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix becomes:

$$1 \leq i \leq 6 \text{ and } 1 \leq j \leq 6$$

$$\begin{aligned} K1_{i,j} &= \Delta t \iiint_{\hat{E}} \left(\frac{k}{\rho c} \left(\frac{\partial L_j}{\partial \varphi_x} \frac{\partial L_i}{\partial \varphi_x} + \frac{\partial L_j}{\partial \varphi_y} \frac{\partial L_i}{\partial \varphi_y} + \frac{\partial L_j}{\partial \varphi_z} \frac{\partial L_i}{\partial \varphi_z} \right) \right) |J| d\hat{V} \\ &\simeq \sum_{g=1}^{6} W_g \Delta t \left(\frac{k}{\rho c} \left(\left.\frac{\partial L_j}{\partial \varphi_x}\right|_g \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g + \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g + \left.\frac{\partial L_j}{\partial \varphi_z}\right|_g \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \right) \right) |J_g| \end{aligned}$$

This elementary stiffness matrix is calculated with three matrices: $K1 = \sum_{g=1}^{6} [B]^T [D] [B]$ as follows:

$$[B] = \begin{bmatrix} B_1 & B_2 & B_3 & B_4 & B_5 & B_6 \end{bmatrix} \tag{3.39}$$

with:

$$B_i = \begin{bmatrix} \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g \\ \\ \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g \\ \\ \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \end{bmatrix} \tag{3.40}$$

and

$$D = W_g |J_g| \Delta t \frac{k}{\rho c} \tag{3.41}$$

3.2.4.2 Advection stiffness matrix

This matrix has the following shape:

$$1 \leq i \leq 6 \text{ and } 1 \leq j \leq 6$$

$$K2_{i,j} = \Delta t \iiint_E u \frac{\partial \theta_j}{\partial x} \theta_i + v \frac{\partial \theta_j}{\partial y} \theta_i + w \frac{\partial \theta_j}{\partial z} \theta_i dV \tag{3.42}$$

Using the reference element $\hat{E}$ and the gaussian points, this matrix becomes:

$$1 \leq i \leq 6 \text{ and } 1 \leq j \leq 6$$

$$
\begin{aligned}
K2_{i,j} &= \Delta t \iiint_{\hat{E}} \left(u \frac{\partial L_j}{\partial \varphi_x} L_i + v \frac{\partial L_j}{\partial \varphi_y} L_i + w \frac{\partial L_j}{\partial \varphi_z} L_i \right) |J| d\hat{V} \\
&\simeq \sum_{g=1}^{6} W_g \Delta t \left(u\left(g\right) \left.\frac{\partial L_j}{\partial \varphi_x}\right|_g L_i\left(g\right) + v\left(g\right) \left.\frac{\partial L_j}{\partial \varphi_y}\right|_g L_i\left(g\right) + w\left(g\right) \left.\frac{\partial L_j}{\partial \varphi_z}\right|_g L_i\left(g\right) \right) |J_g|
\end{aligned}
\tag{3.43}
$$

This elementary stiffness matrix is calculated with three matrices: $K2 = \sum_{g=1}^{6} [B]^T [D][B']$ as follows:

$$
[B] = \left[\begin{array}{cccccc} B_1 & B_2 & B_3 & B_4 & B_5 & B_6 \end{array} \right] \tag{3.44}
$$

with:

$$
B_i = \left[\begin{array}{c} L_i(g) \\ \\ L_i(g) \\ \\ L_i(g) \end{array} \right] \quad B'_i = \left[\begin{array}{c} \left.\frac{\partial L_i}{\partial \varphi_x}\right|_g \\ \\ \left.\frac{\partial L_i}{\partial \varphi_y}\right|_g \\ \\ \left.\frac{\partial L_i}{\partial \varphi_z}\right|_g \end{array} \right] \tag{3.45}
$$

and

$$
D = W_g |J_g| \Delta t \left[\begin{array}{ccc} \sum_{k=1}^{6} u_k L_k(g) & 0 & 0 \\ 0 & \sum_{k=1}^{6} v_k L_k(g) & 0 \\ 0 & 0 & \sum_{k=1}^{6} w_k L_k(g) \end{array} \right] \tag{3.46}
$$

3.2.4.3 Transient

The transient stiffness matrix has the following shape

$$
K_{i,j} = \iint_E \theta_j \phi_i dV \tag{3.47}
$$

We now use gaussian points for the integration :

$$
1 \leq i \leq 6 \text{ and } 1 \leq j \leq 6
$$

$$
\begin{aligned}
K_{i,j} &= \iint_{\hat{E}} L_j L_i |J| d\hat{V} \\
&\simeq \sum_{g=1}^{6} W_g L_j(g) L_i(g) |J_g|
\end{aligned}
\tag{3.48}
$$

This elementary stiffness matrix is calculated with three matrices: $K3 = \sum_{g=1}^{3} [B]^T [D][B]$ as follows:

$$
[B] = \left[\begin{array}{cccccc} B_1 & B_2 & B_3 & B_4 & B_5 & B_6 \end{array} \right] \quad \text{and} \quad D = W_g |J_g| \tag{3.49}
$$

with:

$$
B_i = L_i(g) \tag{3.50}
$$

3.2.5 Elementary load vector

3.2.5.1 Deformational heating

This elementary load vector has the following shape:

$$1 \leq j \leq 6$$

$$F_j = \iiint_E \left(-\frac{\partial T}{\partial t} + \frac{\Phi}{\rho c} \right) \theta_j d\Omega dV \tag{3.51}$$

Using the reference element $\hat{E}$ and the gaussian points, this vector becomes:

$$1 \leq i \leq 6$$

$$F_j = \iiint_{\hat{E}} \left(-\frac{\partial T}{\partial t} + \frac{\Phi}{\rho c} \right) L_j |J| d\hat{V} = \sum_{g=1}^{6} W_g |J_g| \left(-\left.\frac{\partial T}{\partial t}\right|_g + \frac{\Phi(g)}{\rho c} \right) L_j(g) \tag{3.52}$$

3.2.5.2 Basal heating

This elementary load vector has the following shape:

$$1 \leq i \leq 6$$

$$F_j = \iint_{\Gamma_b \cap E} \frac{1}{\rho c} \left(G + \vec{\tau}_b . \vec{u}_b \right) \theta_j dS \tag{3.53}$$

Using the reference element $\hat{E}$ and the gaussian points, this vector becomes:

$$1 \leq j \leq 6$$

$$\begin{aligned} F_j &= \iint_{\Gamma_b \cap \hat{E}} \frac{1}{\rho c} \left(G + \vec{\tau}_b . \vec{u}_b \right) L_j |J| d\hat{S} \\ &\simeq \sum_{g=1}^{3} W_g |J_g| \frac{1}{\rho c} \left(G(g) + \vec{\tau}_b(g) . \vec{u}_b(g) \right) L_j(g) \end{aligned} \tag{3.54}$$

Only the nodal functions assigned to the nodes on Γ_b are involved.

3.2.5.3 Transient

The transient load vector has the following shape[2]:

$$F_j = \iint_E T_i \theta_j |J| dV \tag{3.55}$$

We now use gaussian points for the integration :

$$1 \leq j \leq 6$$

$$\begin{aligned} F_j &= \iint_{\hat{E}} T_i L_j |J| d\hat{V} \\ &\simeq \sum_{g=1}^{6} W_g T_i(g) L_j(g) |J_g| \end{aligned} \tag{3.56}$$

[2] NB: T_i stands for the time step i temperature field

Chapter 4

Prognostic Finite Element Formulation

4.1 Introduction: presentation of the problem

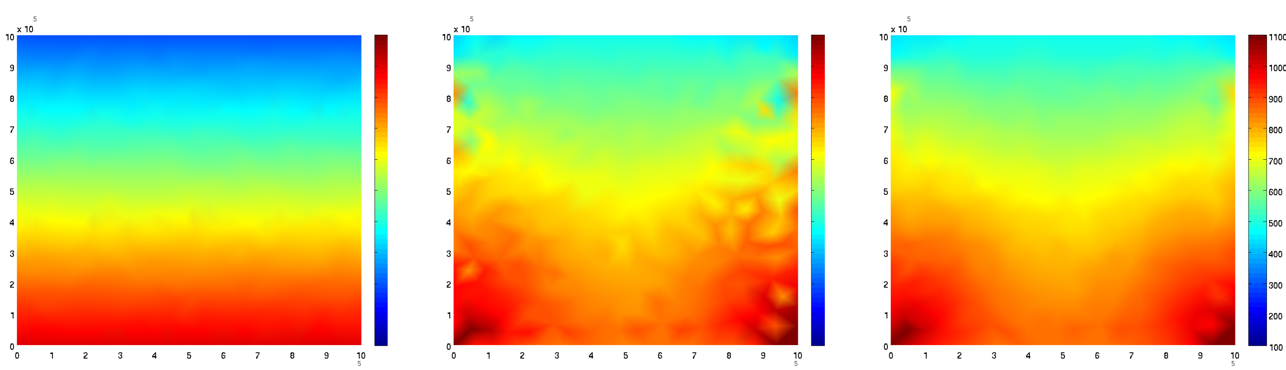
For transient computation, one uses a finite differences time domain scheme. For each time step, the velocity is computed as well as the temperature (taken as transient). But to compute the next time step, one needs to update the glacier's geometry: the thickness evolution is required. This is done thanks to equation (1.28):

$$\frac{\partial H}{\partial t} = -div\left(H\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right) + \dot{M}_s - \dot{M}_b \tag{4.1}$$

We implemented this model using an implicit finite difference approximation for time:

$$\frac{H\left(t_{i+1}\right) - H\left(t_i\right)}{\Delta t} \simeq -div\left(H\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right) + \dot{M}_s - \dot{M}_b \tag{4.2}$$

But doing so introduces a negative diffusion term that leads to unstable solutions. To avoid wiggly solutions, we use the classical technique of adding an *artificial diffusivity*. For a square ice shelf with an accumulation of 10 m/year it gives:



(a) Initial thickness (b) Final thickness (c) With artificial diffusivity

Figure 4.1: Results of the prognostic model for a square ice shelf after 100 years

4.2 Why is the solution unstable?

To make things simpler, we will consider a one-dimensional problem with a constant velocity u with no melting and no accumulation:

$$\frac{\partial H}{\partial t} = -u\frac{\partial H}{\partial x} \tag{4.3}$$

With an implicit finite difference scheme for time, we are actually solving:

$$\frac{H\left(t_{i+1}\right) - H\left(t_i\right)}{\Delta t} = -u\frac{\partial H\left(t_i\right)}{\partial x} \tag{4.4}$$

A Taylor development to the order 2 gives:

$$\left.\frac{\partial H}{\partial t}\right|_{t_i} = \frac{H\left(t_{i+1}\right) - H\left(t_i\right)}{\Delta t} - \frac{\Delta t}{2}\left.\frac{\partial^2 H}{\partial t^2}\right|_{t_i} + \mathcal{O}\left(\Delta t^2\right) \tag{4.5}$$

Therefore, the exact solution of (4.10) is:

$$\frac{H\left(t_{i+1}\right) - H\left(t_i\right)}{\Delta t} = -u\frac{\partial H\left(t_i\right)}{\partial x} + \frac{\Delta t}{2}\left.\frac{\partial^2 H}{\partial t^2}\right|_{t_i} + \mathcal{O}\left(\Delta t^2\right) \tag{4.6}$$

But by solving (4.4) only, we are actually neglecting the last term. By doing so, we are actually solving:

$$\frac{\partial H}{\partial t} = -u\frac{\partial H}{\partial x} - \frac{\Delta t}{2}\frac{\partial^2 H}{\partial t^2} \tag{4.7}$$

We have the following property (progressive waves equation):

$$\frac{\partial^2 H}{\partial t^2} = \frac{\partial}{\partial t}\left(-u\frac{\partial H}{\partial x}\right) = u^2\frac{\partial^2 H}{\partial x^2} \tag{4.8}$$

Therefore, the solution of (4.4) is the same as the solution of the following diffusion equation in which the diffusion coefficient is negative:

$$\frac{\partial H}{\partial t} = -u\frac{\partial H}{\partial x} - \frac{\Delta t}{2}u^2\frac{\partial^2 H}{\partial x^2} \tag{4.9}$$

This negative diffusion term that leads to unstable solutions: progressive waves of H (oscillations) are exponentially growing with time.

4.3 Streamline upwind / Petrov Galerkin formulation

It can be shown that for finite differences, a cure for the unwanted wiggles is provided by use of upwind differences to approximate the convective term, which essentially introduces a truncation error having the form of a physical diffusion. This term is a "balancing-diffusion" that is counteracting the negative diffusion introduced by the truncation error. By so doing, the numerical solution will yield exact values at the nodes. In addition to directly adding a diffusion term, similar effects of stabilization can be achieved in finite difference through According to Brooks and Hughes (1982), the artificial diffusion is only desirable and required in the direction of the flow vector. The schemes eliminated the crosswind diffusion, which unnecessarily overdiffuses. This technique is named the *streamline upwind/Petrov-Galerkin* (SUPG) formulation.

For finite elements, a conventional Galerkin finite element discretization is employed with the same additional artificial diffusion:

$$\frac{\partial H}{\partial t} = -div\left(H\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right) + \dot{M}_s - \dot{M}_b + div\left(\underline{\kappa}\ \overrightarrow{grad}H\right) \tag{4.10}$$

where $\underline{\kappa}$ is a 2×2 diffusivity tensor designed to diffuse along flowlines and minimize diffusion across flowlines. According to MacAyeal[8] (p172), for a given element E of the mesh it has the following expression:

$$\underline{\kappa} = \left(\frac{\sqrt{2Area(E)}}{2}\right)\left[\begin{array}{cc}\frac{1}{3}\left|\sum_{i=1}^{3}u_i\right| & 0\\ 0 & \frac{1}{3}\left|\sum_{i=1}^{3}v_i\right|\end{array}\right] = \left[\begin{array}{cc}\kappa_{xx} & 0\\ 0 & \kappa_{yy}\end{array}\right] \tag{4.11}$$

where $(u_i v_i)$ are the nodal values of the depth averaged horizontal velocity components. The artificial diffusivity tensor is designed to give about the same damping as that which would be associated with upwind differencing if the finite-element mesh consisted of a 2d regular array of nodes similar to a finite difference grid. (The absolute value of the local velocity is used in defining $\underline{\kappa}$ to ensure the diffusivity will always be positive). The addition of artificial diffusion bumps the prognostic equation up to a higher order (two spatial derivatives instead of one are now involved), and this necessitates specifying additional boundary conditions. For our purposes, we might as well take $\overrightarrow{grad}H.\overrightarrow{n} = 0$ on all portions of $\partial\Omega$.

4.4 Weak Formulation

Here, the domain Ω is a 2d domain (since H only depends on x and y). We define a kinematically admissible velocity field $\overrightarrow{\phi}$:

$$\phi(x, y) \in \{H^1(\Omega \cup \partial\Omega)\} \tag{4.12}$$

For any function of this virtual field, we can take the product of the equation (4.10) with ϕ, and integrate this product over the domain. This gives us:

$$\iint_\Omega \frac{\partial H}{\partial t}\phi + div\left(H\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right)\phi - \dot{M}_s\phi + \dot{M}_b\phi - div\left(\underline{\kappa}\ \overrightarrow{grad}H_i\right)\phi d\Omega = 0 \tag{4.13}$$

Exactly as in the transient thermal section, one uses a finite differences time domain scheme to evaluate the transient term. For a given time step n, the transient term is computed implicitly[1] as follow:

$$\left.\frac{\partial H}{\partial t}\right|_n = \frac{H_n - H_{n-1}}{\Delta t} \tag{4.14}$$

Where Δt is the time difference between t_n and t_{n-1}. The weak formuation becomes:

$$\iint_\Omega H_i\phi - H_{n-1}\phi + \Delta t\ div\left(H_n\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right)\phi - \Delta t\ \dot{M}_s\phi + \Delta t\ \dot{M}_b\phi - \Delta t\ div\left(\underline{\kappa}\ \overrightarrow{grad}H_i\right)\phi d\Omega = 0 \tag{4.15}$$

One can integrate by part the artifficial damping term:

$$\iint_\Omega H_n\phi + \Delta t\ div\left(H_n\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right)\phi + \Delta t\ \left(\underline{\kappa}\ \overrightarrow{grad}H_n\right).\overrightarrow{grad}\phi d\Omega = \iint_\Omega H_{n-1}\phi + \Delta t\ \dot{M}_s\phi - \Delta t\ \dot{M}_b\phi d\Omega + \int_{\partial\Omega}\Delta t\ \left(\underline{\kappa}\ \overrightarrow{grad}H_n\right).\overrightarrow{n}\phi dSd\Omega \tag{4.16}$$

[1] an implicit method is much more stable for this kind of equation

Using the boudary condition, the last term vanishes:

$$\iint_{\Omega} H_n\phi + \Delta t\ div\left(H_n\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\right)\phi + \Delta t\left(\underline{\kappa}\ \overrightarrow{grad}H_n\right).\overrightarrow{grad}\phi d\Omega = \iint_{\Omega} H_{n-1}\phi + \Delta t\dot{M}_s\phi - \Delta t\dot{M}_b\phi d\Omega \tag{4.17}$$

4.5 Elementary stiffness matrix and load vector

Following the same method as in the previous chapters, we obtain the stiffness matrix as follows:

$$\begin{aligned} K_{i,j} &= \iint_{\Omega} \phi_j\phi_i + \Delta t\ div\left(\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]\phi_j\right)\phi_i + \Delta t\left(\kappa_{xx}\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x} + \kappa_{yy}\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}\right)d\Omega \\ F_i &= \iint_{\Omega} H_{n-1}\phi_i + \Delta t\dot{M}_s\phi_i - \Delta t\dot{M}_b\phi_i d\Omega \end{aligned} \tag{4.18}$$

We now decompose these two matrices over the reference elements and use gaussian points for the integration :

$$1 \leq i \leq 3 \text{ and } 1 \leq j \leq 3$$

$$\begin{aligned} K_{i,j} &= \iint_{\hat{E}} L_jL_i + \Delta t\ div\left(\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]L_j\right)L_i + \Delta t\left(\kappa_{xx}\frac{\partial L_j}{\partial x}\frac{\partial L_i}{\partial x} + \kappa_{yy}\frac{\partial L_j}{\partial y}\frac{\partial L_i}{\partial y}\right)|J|d\hat{S} \\ &\simeq \sum_{g=1}^{3} W_g\left(\underbrace{L_jL_i}_{K1} + \underbrace{\Delta t\ \ div\left(\left[\begin{array}{c}\bar{u}\\ \bar{v}\end{array}\right]L_j\right)\Bigg|_g L_i(g)}_{K2} + \underbrace{\Delta t\left(\kappa_{xx}\left.\frac{\partial L_j}{\partial x}\right|_g\left.\frac{\partial L_i}{\partial x}\right|_g + \kappa_{yy}\left.\frac{\partial L_j}{\partial y}\right|_g\left.\frac{\partial L_i}{\partial y}\right|_g\right)}_{K3}\right)|J_g| \\ F_i &= \iint_{\hat{E}} H_{n-1}L_i + \Delta t\dot{M}_sL_i - \Delta t\dot{M}_bL_i|J|d\Gamma \\ &\simeq \sum_{g=1}^{3}\left(H_{n-1}L_i(g) + \Delta t\dot{M}_sL_i(g) - \Delta t\dot{M}_bL_i(g)\right)|J_g|\delta(icesheet) \end{aligned}$$

4.5.1 First term $K1$

This elementary stiffness matrix is calculated with three matrices: $K1 = \sum_{g=1}^{3}[B]^T[D][B]$ as follows:

$$[B] = \left[\begin{array}{ccc} B_1 & B_2 & B_3 \end{array}\right] \quad \text{and} \quad D = W_g|J_g| \tag{4.19}$$

with:

$$B_i = L_i(g) \tag{4.20}$$

4.5.2 Second term $K2$

The second elementary stiffness matrix is computed as follows: $K2 = \sum_{g=1}^{3}[B]^T[D][B] + [B]^T[D'][B']$

$$[B] = \left[\begin{array}{ccc} B_1 & B_2 & B_3 \end{array}\right] \quad [B'] = \left[\begin{array}{ccc} B_1' & B_2' & B_3' \end{array}\right] \tag{4.21}$$

$$D = \Delta t\, W_g |J_g| \left[\begin{array}{cc} \left.\dfrac{\partial \overline{u}}{\partial \varphi_x}\right|_g & 0 \\ 0 & \left.\dfrac{\partial \overline{v}}{\partial \varphi_y}\right|_g \end{array} \right] \quad \text{and} \quad D' = \Delta t\, W_g |J_g| \left[\begin{array}{cc} \overline{u}(g) & 0 \\ 0 & \overline{v}(g) \end{array} \right] \tag{4.22}$$

with:

$$B_i = \left[\begin{array}{c} L_i(g) \\ L_i(g) \end{array} \right] \quad \text{and} \quad B'_i = \left[\begin{array}{c} \left.\dfrac{\partial L_i}{\partial \varphi_x}\right|_g \\ \left.\dfrac{\partial L_i}{\partial \varphi_y}\right|_g \end{array} \right] \tag{4.23}$$

4.5.3 Third term $K3$

The third elementary stiffness matrix is computed with three matrices: $K3 = \sum_{g=1}^{3} [B]^T [D][B]$ as follows:

$$[B] = \left[\begin{array}{ccc} B_1 & B_2 & B_3 \end{array} \right] \quad \text{and} \quad D = \Delta t\, W_g |J_g| \underline{\kappa} \tag{4.24}$$

with:

$$B_i = \left[\begin{array}{c} \left.\dfrac{\partial L_i}{\partial \varphi_x}\right|_g \\ \left.\dfrac{\partial L_i}{\partial \varphi_y}\right|_g \end{array} \right] \tag{4.25}$$

4.6 A comment on Δt

According to MacAyeal[8] (p175), the implicit time-stepping scheme suggests that the solution of the prognostic (mass balance) equation should be stable for any time-step size. In practice, the time-step size should be chosen so that all imaginary passive-tracer particles initially released at the nodal points (except those starting at ice-front nodes) remain confined by one single element throughout the time span $[t, t + \Delta t]$. In other words, it is best to choose a Δt small enough so that the distance traveled by one of these imaginary passive-tracer particles will not exceed the mesh spacing. In circumstances where the velocity field of the ice shelf changes substantially through the time, it may be wise to consider adaptively modifying Δt during the simulation. The time-step size should follow the rule:

$$\Delta t = \frac{\frac{\Delta L}{\sqrt{2}}}{2 \max{(v)}} \tag{4.26}$$

where ΔL is the nondimensional size of the finite-element mesh spacing (shortest distance between node points in the regular mesh). The idea of the above equation is that Δt is just large enough so that the fastest moving passive-tracer particle will move half the distance along a diagonal trajectory across a typical triangular element.

Chapter 5

Penalty Method

5.1 Introduction

Penalty method is a well known method which is often applied in finite element analysis of contact problems (See Wriggers [12]). A penalty term, which can be interpreted as a spring with a very large stiffness, is added to the energy so that when a constraint is not fulfilled, the energy is much higher than it should be. Thanks to this penalized energy, a state that minimizes the energy is also a state that respects the constraints. Since finite element is a method that tries to minimize the energy of a system, the penalty forces the model to be properly constrained. In $ISSM$, penalties are used beween 2d and 3d meshes (horizontal velocities), bewteen 2d and 2d meshes (basal vertical velocity), bewteen 3d and 3d meshes (vertical velocity and temperature), and for melting.

5.2 Constraining velocities of two meshes

The first exemple of penalty used in $ISSM$ is between the 2d and the 3d model (to compute the horizontal velocity (u, v)). In the same mesh, one wants to compute the velocity field with a 2d model on one part of the mesh, and with a 3d model on the other. We start from a 2d mesh, the grids on the boundary between the 2d model and the 3d model are splitted into 2 grids. Then, the part of the mesh assigned to compute the 3d model is extruded into several layers in order to compute have a 3d mesh. For example, one can have the following geometry:

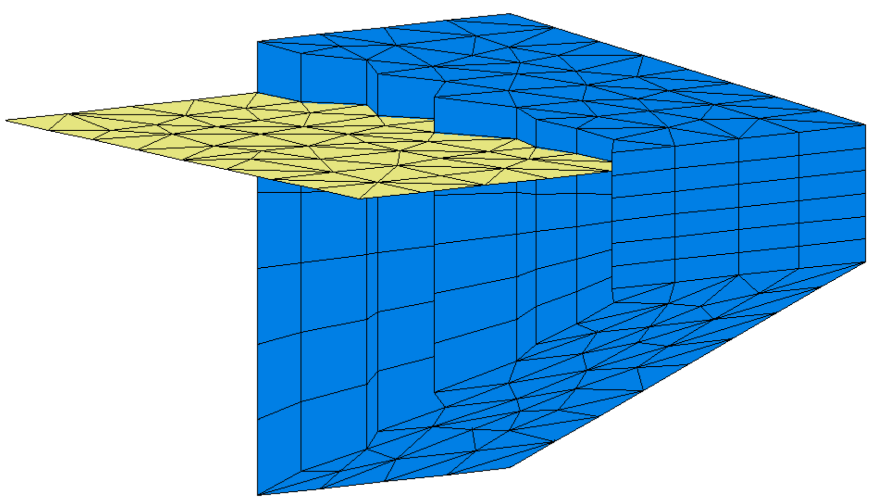


Figure 5.1: Mesh of a system coupling a 2d and a 3d model

Although two different models are used, one wants to have a continuous velocity between the two part of the mesh. The penalties applied here are used to constrain the velocity of the 2d grids (in red) at the models interface to a value equal to the one of all the grids vertically extruded of the 3d mesh (blue).

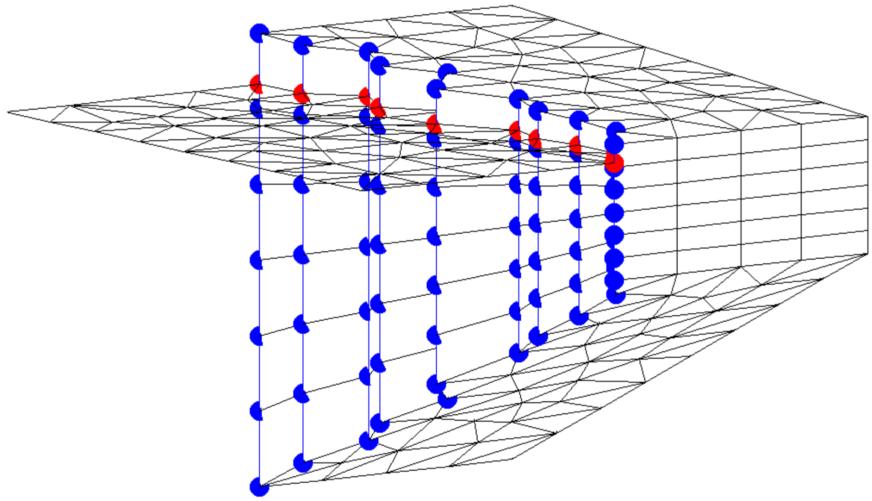


Figure 5.2: Applied penalties at the models interface

To ensure that, one adds penalties to the energy of the system: when the velocity of one of the 2d grid on the boundary is not the same as the corresponding 3d grids the energy must increase. Here, we have 10 red grids and for each red grid there are 10 blue grids (there are 10 layers). Therefore, there are 100 pair of grids and one wants the velocity of each grid to be the same as its pair. The penalized energy has the following expression (for the sake of simplicity, only one component is represented):

$$E = [U]^T [K] [U] + [U]^T [F] + \sum_{n=1}^{red\ grids} \left(\sum_{l=1}^{layers} \kappa \left(u_n - u_{nl} \right)^2 \right) \tag{5.1}$$

Where K and F are the stiffness matrix and the load vetor of the system, U is the velocity of the grids, and κ is the penalty term: a very large number compared to K_{ij} so that the energy of the system explode when one u_n is two far from its corresponding u_{nl} (ie. when the velocities of the 2 grids belonging to the same pair of grids are not equal). From now on, we will explain the penalty method only for one pair of grids. The general case can be easily found by reproducing the following pattern for every pair of grids. The penalized energy is:

$$E = [U]^T [K] [U] + [U]^T [F] + \kappa \left(u_{red} - u_{blue} \right)^2 \tag{5.2}$$

We can transform this expression as follows:

$$E = \begin{bmatrix} \vdots \\ u_j \\ \vdots \end{bmatrix}^T \begin{bmatrix} & \vdots & \\ \cdots & K_{ij} & \cdots \\ & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ u_j \\ \vdots \end{bmatrix} + \begin{bmatrix} \vdots \\ u_j \\ \vdots \end{bmatrix}^T \begin{bmatrix} \vdots \\ F_j \\ \vdots \end{bmatrix} + \begin{bmatrix} u_{red} \\ u_{blue} \end{bmatrix}^T \begin{bmatrix} \kappa & -\kappa \\ -\kappa & \kappa \end{bmatrix} \begin{bmatrix} u_{red} \\ u_{blue} \end{bmatrix} \tag{5.3}$$

Finally one can concatenate the stiffness matrix with the penalty matrix:

$$E = \begin{bmatrix} \vdots \\ u_{red} \\ \vdots \\ u_{blue} \\ \vdots \end{bmatrix}^T \begin{bmatrix} & \vdots & & \vdots & \vdots & \\ \cdots & K_{rr}+\kappa & \cdots & K_{rb}-\kappa & \cdots \\ & \vdots & \vdots & \vdots & \\ \cdots & K_{br}-\kappa & \cdots & K_{bb}+\kappa & \cdots \\ & \vdots & \vdots & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ u_{red} \\ \vdots \\ u_{blue} \\ \vdots \end{bmatrix} + \begin{bmatrix} \vdots \\ u_j \\ \vdots \end{bmatrix}^T \begin{bmatrix} \vdots \\ F_j \\ \vdots \end{bmatrix} \tag{5.4}$$

Given that energy, the linear system to solve is:

$$\boxed{\begin{bmatrix} & \vdots & \vdots & \vdots & \\ \cdots & K_{rr}+\kappa & \cdots & K_{rb}-\kappa & \cdots \\ & \vdots & \vdots & \vdots & \\ \cdots & K_{br}-\kappa & \cdots & K_{bb}+\kappa & \cdots \\ & \vdots & \vdots & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ u_{red} \\ \vdots \\ u_{blue} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ F_j \\ \vdots \end{bmatrix}} \tag{5.5}$$

Since κ is much larger that K_{ij}, the first equation that links u_{red} to u_{blue}:

$$\sum_{i=1}^{n} K_{ir} u_i + \kappa \left(u_{red} - u_{blue} \right) = F_r \tag{5.6}$$

To the zero order:

$$\kappa \left(u_{red} - u_{blue} \right) = 0 \qquad \Longleftrightarrow \qquad u_{red} = u_{blue} \tag{5.7}$$

We see that the penalized stiffness matrix ensure that the velocities of the grids pair is the same. The exact same process is used between two 2d meshes (basal vertical velocity) and two 3d meshes (temperature or vertical velocity). The crucial step is to build the pairs of grids and then add penalties on the corresponding rows and clumns of the stiffness matrix.

Here are the results for a square iceshelf. The upperview does not show any discontinuity in the velocity field:

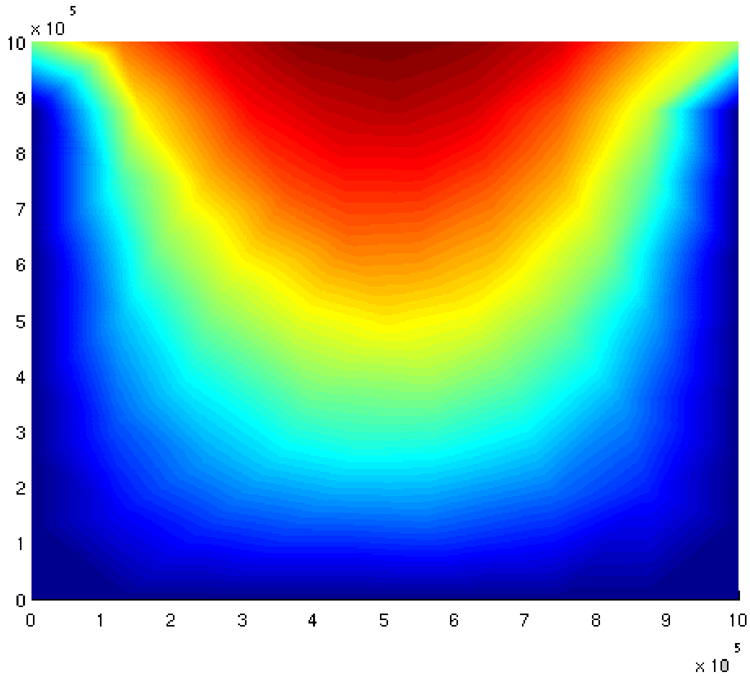


Figure 5.3: coupling between a 2d and a 3d model, upper view

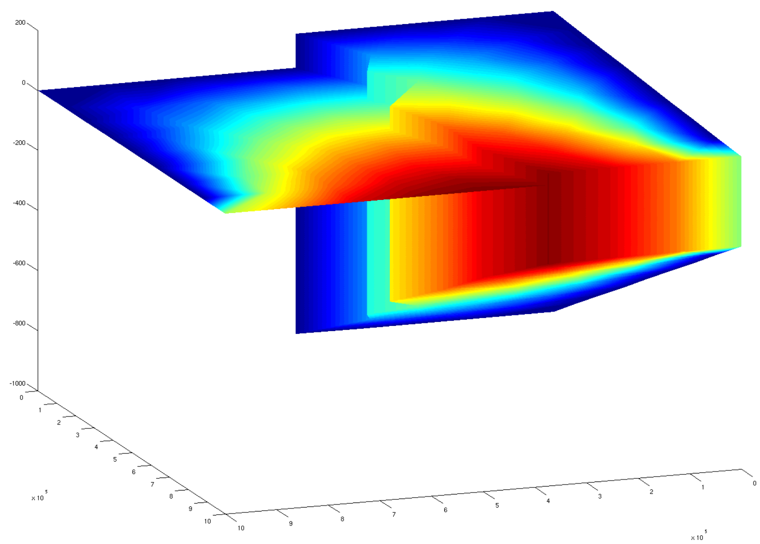


Figure 5.4: coupling between a 2d and a 3d model, 3d view

5.3 Stokes model

We have seen in section 2.4 that the boundary condition on ice/till interface for the vertical velocity was:

$$w = -\frac{\partial b}{\partial x}u - \frac{\partial b}{\partial y}v \tag{5.8}$$

In order to apply this boundary condition, we use a penalty to connect the three components of the

velocity.
The equation of the penalty we have to add in the elementary matrix is:

$$\kappa \left(w + \frac{\partial b}{\partial x} u + \frac{\partial b}{\partial y} v \right) = 0 \tag{5.9}$$

If we note K the elementary stiffness matrix, and F the elementary load vector , the system to solve with these penalties is:

$$\begin{bmatrix} & \vdots & \vdots & \vdots & \\ \cdots & K_{3i,3j-2} + \kappa\, n_x & K_{3i,3j-1} + \kappa\, n_y & K_{3i,3j} + \kappa & \cdots \\ & \vdots & \vdots & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ u_j \\ v_j \\ w_j \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ F_j \\ \vdots \end{bmatrix} \tag{5.10}$$

5.4 Thermal model

We have seen in section 1.7.5 that the equation on the Ice/Till interface was:

$$\begin{cases} -k \left.\dfrac{\partial T}{\partial z}\right|_b = G + \overrightarrow{\tau}_b . \overrightarrow{u}_b - \rho L \dot{M}_b \\ \\ \dot{M}_b \left(T - T_{pmp}\right) = 0 \\ \\ T - T_{pmp} \leq 0 \\ \\ \dot{M}_b \geq 0 \end{cases} \tag{5.11}$$

The third equation imposes the temperature to be below the pressure melting point T_{pmp}. One can use penalty to constrain the temperature. We will see that one can then use these penalties to deduce the melting rate.

5.4.1 Constraining temperature

The temperature of each grid must stay below the pressure melting point T_{pmp}. In the first step, it is assumed that the computed temperature is everywhere below T_{pmp}. The computed temperature is then compared to the pressure melting point. When a grid goes above T_{pmp}, one adds penalties in the stiffness matrix and in the load vector to keep it at the pressure melting point as follows:

$$\boxed{\begin{bmatrix} & \vdots & \\ \cdots & K_{ij} + \kappa & \cdots \\ & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ T_j \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ F_j + \kappa T_{pmp} \\ \vdots \end{bmatrix}} \tag{5.12}$$

So that T_j is constrained to T_{pmp}. One does that for all the grids above the pressure melting point. Then the temperature is computed again and compared to T_{pmp}. The grids above this temperature are the penalized, whereas those which were already penalized and present a temperature now below the pressure melting point are released. After several iterations, one finaly gets the temperature field.

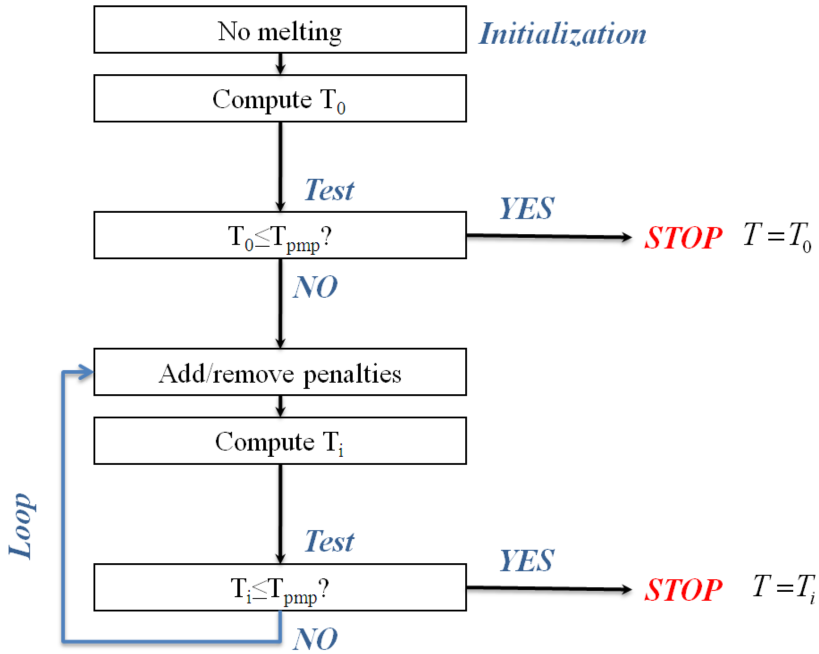


Figure 5.5: Thermal Algotithm of *ISSM*

5.4.2 Basal melting

using the previous algorithm, one can solve the third equation of the system (5.11). But the use of penalties allows also to compute melting. One a penalty is applied (T reaches the pressure melting point), the system to solve is:

$$\begin{bmatrix} & \vdots & \\ \cdots & K_{ij} & \cdots \\ & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ T_j \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ F_j + \kappa\left(T_{pmp} - T_j\right) \\ \vdots \end{bmatrix} \tag{5.13}$$

We saw that the melting was not computed in the load vector. But one can use the previous load vector to deduce melting: the total load veclor (that includes basal melting) should have had the following expression:

$$\begin{aligned} F_j^{tot} &= \iiint_\Omega \left(-\frac{\partial T}{\partial t} + \frac{\Phi}{\rho c}\right)\theta_i d\Omega + \iint_{\Gamma_b} \frac{1}{\rho c}\left(G + \overrightarrow{\tau}_b.\overrightarrow{u}_b + \rho L \dot{M}_b\right)\theta_j dS \\ &= F_j + \frac{1}{\rho c}\iint_{\Gamma_b} \rho L \dot{M}_b \theta_j dS \end{aligned} \tag{5.14}$$

That gives a simple relation between the penalized load and basal melting:

$$\kappa\left(T_{pmp} - T_j\right) = \frac{L}{c}\iint_{\Gamma_b} \dot{M}_b \theta_j dS \tag{5.15}$$

We can use a 2d mesh (we are dealing with the glacier base only), to deduce $\dot{M}_{b_j}$:

$$\boxed{\begin{bmatrix} & \vdots & \\ \cdots & K'_{ij} & \cdots \\ & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ \dot{M}_{b_j} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ F'_j \\ \vdots \end{bmatrix}} \tag{5.16}$$

with:

$$\boxed{\begin{aligned} K'_{ij} &= \frac{L}{c} \iint_{\Gamma_b} \theta_i \theta_j dS \\ F'_j &= \kappa \left(T_{pmp} - T_j \right) \end{aligned}} \tag{5.17}$$

The grids where no penalty is applied (ie. the temperature is below the pressure melting point) are constrained to 0. Thanks to that we solved all the equations of the system (5.11).

Appendix A

Reference Elements

In finite element models, rather than calculating the integrals for each element, one uses a *Reference Element* $\hat{E}$, and a transformation φ that transforms the reference element to the element E.

A.1 One dimension, Segment

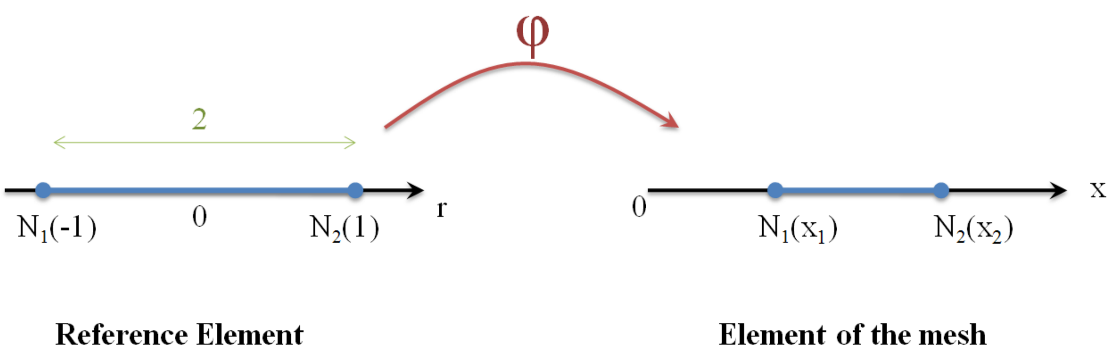


Figure A.1: Transformation from the reference segment, to the current segment of the mesh

For any function $f(x)$, we have the following equation:

$$\int_E f(x)dx = \int_{\hat{E}} f(\varphi(r))|\varphi'|dr \tag{A.1}$$

For the reference element $\hat{E}$, the nodal functions are:

$$\begin{aligned} L_1(r) &= -\frac{1}{2}r + \frac{1}{2} \\ L_2(r) &= \frac{1}{2}r + \frac{1}{2} \end{aligned} \tag{A.2}$$

One can use these nodal functions to calculate the function φ:

$$\varphi(r) = x_1 L_1(r) + x_2 L_2(r) \tag{A.3}$$

So,

$$\varphi(r) = \frac{1}{2}(-x_1 + x_2)\, r + \frac{1}{2}(x_1 + x_2) \tag{A.4}$$

The derivative of φ is :

$$\varphi'(r) = \frac{1}{2}\left(-x_1 + x_2\right) \tag{A.5}$$

A.2 Two dimensions, Triangle

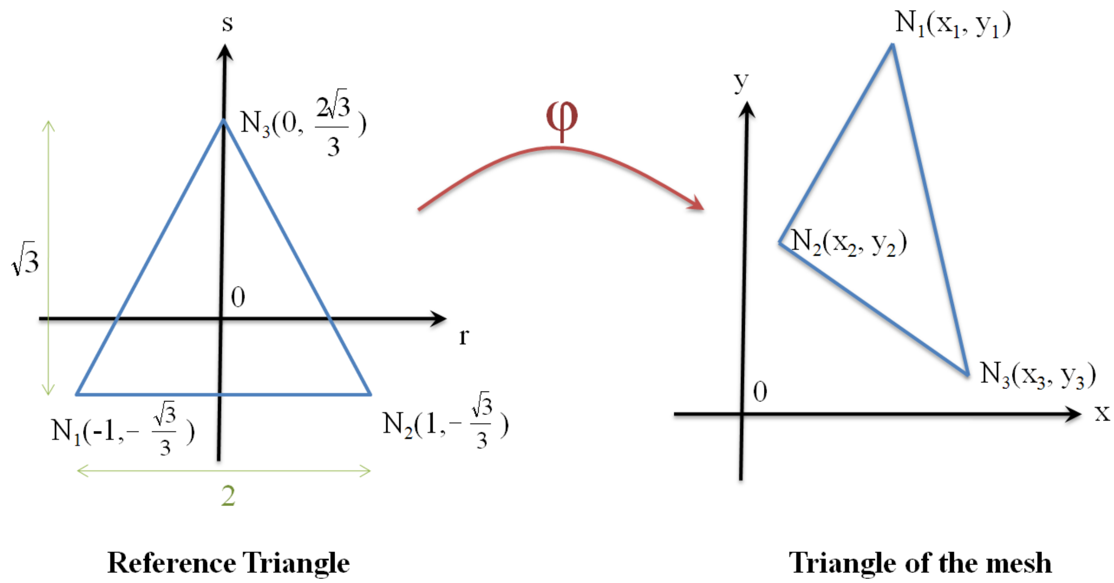


Figure A.2: Transformation from the reference triangle, to the current element of the mesh

For any function $f(x,y)$, we have the following equation:

$$\iint_E f(x,y)dS = \iint_{\hat{E}} \hat{f}(r,s)|J|dS \tag{A.6}$$

Where J is the Jacobian determinant of the φ function. For the reference element $\hat{E}$, the nodal functions are:

$$\begin{aligned} L_1(r,s) &= -\frac{1}{2}r - \frac{\sqrt{3}}{6}s + \frac{1}{3} \\ L_2(r,s) &= \frac{1}{2}r - \frac{\sqrt{3}}{6}s + \frac{1}{3} \\ L_3(r,s) &= \frac{\sqrt{3}}{3}s + \frac{1}{3} \end{aligned} \tag{A.7}$$

One can use these nodal functions to calculate the function φ:

$$\varphi(r,s) = \begin{cases} x = & x_1L_1(r,s) + x_2L_2(r,s) + x_3L_3(r,s) \\ y = & y_1L_1(r,s) + y_2L_2(r,s) + y_3L_3(r,s) \end{cases} \tag{A.8}$$

So,

$$\varphi(r,s) = \begin{cases} x = \left(-\frac{1}{2}x_1 + \frac{1}{2}x_2\right)r + \left(-\frac{\sqrt{3}}{6}x_1 - \frac{\sqrt{3}}{6}x_2 + \frac{\sqrt{3}}{3}x_3\right)s + \frac{1}{6}\left(2x_1 + 2x_2 + x_3\right) \\ y = \left(-\frac{1}{2}y_1 + \frac{1}{2}y_2\right)r + \left(-\frac{\sqrt{3}}{6}y_1 - \frac{\sqrt{3}}{6}y_2 + \frac{\sqrt{3}}{3}y_3\right)s + \frac{1}{6}\left(2y_1 + 2y_2 + y_3\right) \end{cases} \tag{A.9}$$

The Jacobian of the φ function is now easily computable:

$$J = \begin{bmatrix} \dfrac{\partial \varphi_x}{\partial r} & \dfrac{\partial \varphi_x}{\partial s} \\[2ex] \dfrac{\partial \varphi_y}{\partial r} & \dfrac{\partial \varphi_y}{\partial s} \end{bmatrix} = \begin{bmatrix} \dfrac{1}{2}\left(x_2 - x_1\right) & \dfrac{\sqrt{3}}{6}\left(2x_3 - x_1 - x_2\right) \\[2ex] \dfrac{1}{2}\left(y_2 - y_1\right) & \dfrac{\sqrt{3}}{6}\left(2y_3 - y_1 - y_2\right) \end{bmatrix} \tag{A.10}$$

A.3 Two dimensions, Squares

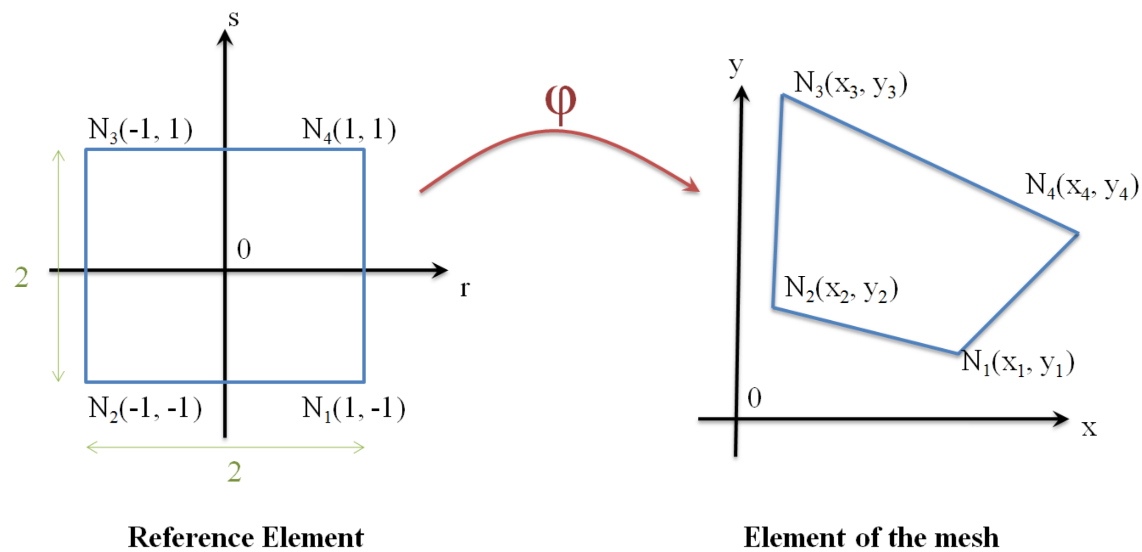


Figure A.3: Transformation from the reference quadrangle, to the current element of the mesh

For any function $f(x, y)$, we have the following equation:

$$\iint_E f(x,z)dS = \iint_{\hat{E}} \hat{f}(r,s)|J|dS \tag{A.11}$$

Where J is the Jacobian determinant of the φ function. For the reference element $\hat{E}$, the nodal functions are:

$$\begin{aligned}
L_1(r,s) &= -\frac{1}{4}(r+1)(s-1) \\
L_2(r,s) &= \frac{1}{4}(r-1)(s-1) \\
L_3(r,s) &= -\frac{1}{4}(r-1)(s+1) \\
L_4(r,s) &= \frac{1}{4}(r+1)(s+1)
\end{aligned} \tag{A.12}$$

One can use these nodal functions to calculate the function φ:

$$\varphi(r,s) = \begin{cases} x = & x_1L_1(r,s) + x_2L_2(r,s) + x_3L_3(r,s) + x_4L_4(r,s) \\ z = & z_1L_1(r,s) + z_2L_2(r,s) + z_3L_3(r,s) + z_4L_4(r,s) \end{cases} \tag{A.13}$$

So,

$$
\varphi(r,s) = \begin{cases} x &= \dfrac{1}{4}\left(-x_1+x_2-x_3+x_4\right)rs + \dfrac{1}{4}\left(x_1-x_2-x_3+x_4\right)r \\ & +\dfrac{1}{4}\left(-x_1-x_2+x_3+x_4\right)s + \dfrac{1}{4}\left(x_1+x_2+x_3+x_4\right) \\ z &= \dfrac{1}{4}\left(-z_1+z_2-z_3+z_4\right)rs + \dfrac{1}{4}\left(z_1-z_2-z_3+z_4\right)r \\ & +\dfrac{1}{4}\left(-z_1-z_2+z_3+z_4\right)s + \dfrac{1}{4}\left(z_1+z_2+z_3+z_4\right) \end{cases} \tag{A.14}
$$

The Jacobian of the φ function is now easily computable:

$$
J = \begin{bmatrix} \dfrac{\partial \varphi_x}{\partial r} & \dfrac{\partial \varphi_x}{\partial s} \\ \dfrac{\partial \varphi_y}{\partial r} & \dfrac{\partial \varphi_y}{\partial s} \end{bmatrix} \tag{A.15}
$$

$$
\begin{aligned}
J_{11} &= \frac{1}{4}\left(-x_1+x_2-x_3+x_4\right)s + \frac{1}{4}\left(x_1-x_2-x_3+x_4\right) \\
J_{12} &= \frac{1}{4}\left(-x_1+x_2-x_3+x_4\right)r + \frac{1}{4}\left(-x_1-x_2+x_3+x_4\right) \\
J_{12} &= \frac{1}{4}\left(-z_1+z_2-z_3+z_4\right)s + \frac{1}{4}\left(z_1-z_2-z_3+z_4\right) \\
J_{22} &= \frac{1}{4}\left(-z_1+z_2-z_3+z_4\right)r + \frac{1}{4}\left(-z_1-z_2+z_3+z_4\right)
\end{aligned} \tag{A.16}
$$

A.4 Three dimensions, tetrahedron

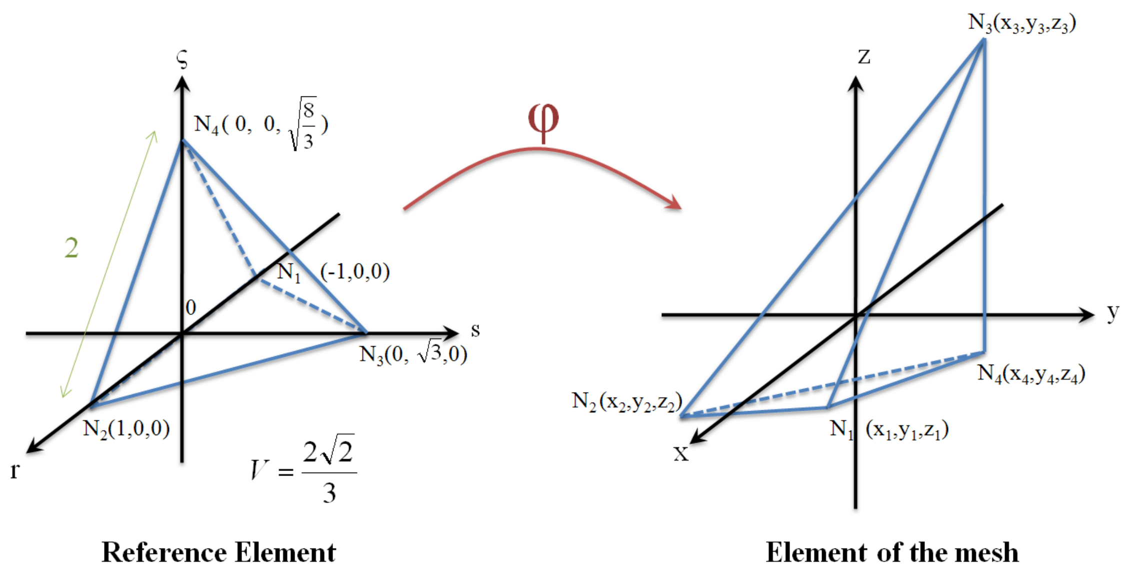


Figure A.4: Transformation from the reference element, to the current pentahedron of the mesh

For any function $f(x, y, z)$, the transformation verifies:

$$\iiint_E f(x,y,z)dV = \iiint_{\hat{E}} \hat{f}(r,s,\zeta)|J|d\hat{V} \tag{A.17}$$

Where J is the Jacobian determinant of the φ function. For the reference element $\hat{E}$, the nodal functions are:

$$\begin{aligned}
L_1(r,s,\zeta) &= \frac{1}{2}\left(1 - r - \frac{1}{\sqrt{3}}s - \frac{1}{\sqrt{6}}\zeta\right) \\
L_2(r,s,\zeta) &= \frac{1}{2}\left(1 + r - \frac{1}{\sqrt{3}}s - \frac{1}{\sqrt{6}}\zeta\right) \\
L_3(r,s,\zeta) &= \frac{1}{\sqrt{3}}\left(s - \frac{1}{\sqrt{8}}\zeta\right) \\
L_4(r,s,\zeta) &= \sqrt{\frac{3}{8}}\zeta
\end{aligned} \tag{A.18}$$

One can use these nodal functions to calculate the function φ:

$$\varphi(r,s) = \begin{cases} x = \sum_{i=1}^{4} x_i L_i(r,s,\zeta) \\ y = \sum_{i=1}^{4} y_i L_i(r,s,\zeta) \\ z = \sum_{i=1}^{4} z_i L_i(r,s,\zeta) \end{cases} \tag{A.19}$$

For example, the first component of $\varphi(r,s,\zeta)$ is:

$$\varphi_x(r,s,t) = \frac{1}{2}\left(x_1 + x_2\right) + \frac{1}{2}\left(-x_1 + x_2\right) r + \frac{1}{2\sqrt{3}}\left(-x_1 - x_2 + 2x_3\right) s + \frac{1}{2\sqrt{6}}\left(-x_1 - x_2 - x_3 + x_4\right) \zeta$$

The 2 other components have exactly the same equation, except the the x_i must be replaced by in turn y_i and z_i. The Jacobian of the φ function is now easily computable:

$$J = \begin{bmatrix} \frac{\partial \varphi_x}{\partial r} & \frac{\partial \varphi_x}{\partial s} & \frac{\partial \varphi_x}{\partial \zeta} \\ \frac{\partial \varphi_y}{\partial r} & \frac{\partial \varphi_y}{\partial s} & \frac{\partial \varphi_y}{\partial \zeta} \\ \frac{\partial \varphi_z}{\partial r} & \frac{\partial \varphi_z}{\partial s} & \frac{\partial \varphi_z}{\partial \zeta} \end{bmatrix} \tag{A.20}$$

with:

$$J_{11} = \frac{1}{2}\left(-x_1 + x_2\right) \qquad J_{21} = \frac{1}{2}\left(-y_1 + y_2\right) \qquad J_{31} = \frac{1}{2}\left(-z_1 + z_2\right)$$

$$J_{12} = \frac{1}{2\sqrt{3}}\left(-x_1 - x_2 + 2x_3\right) \qquad J_{22} = \frac{1}{2\sqrt{3}}\left(-y_1 - y_2 + 2y_3\right) \qquad J_{32} = \frac{1}{2\sqrt{3}}\left(-z_1 - z_2 + 2z_3\right)$$

$$J_{31} = \frac{1}{2\sqrt{6}}\left(-x_1 - x_2 - x_3 + x_4)\right) \qquad J_{32} = \frac{1}{2\sqrt{6}}\left(-x_1 - y_2 - x_3 + y_4\right) \qquad J_{33} = \frac{1}{2\sqrt{6}}\left(-z_1 - z_2 - z_3 + z_4\right)$$

A.5 Three dimensions, pentahedron

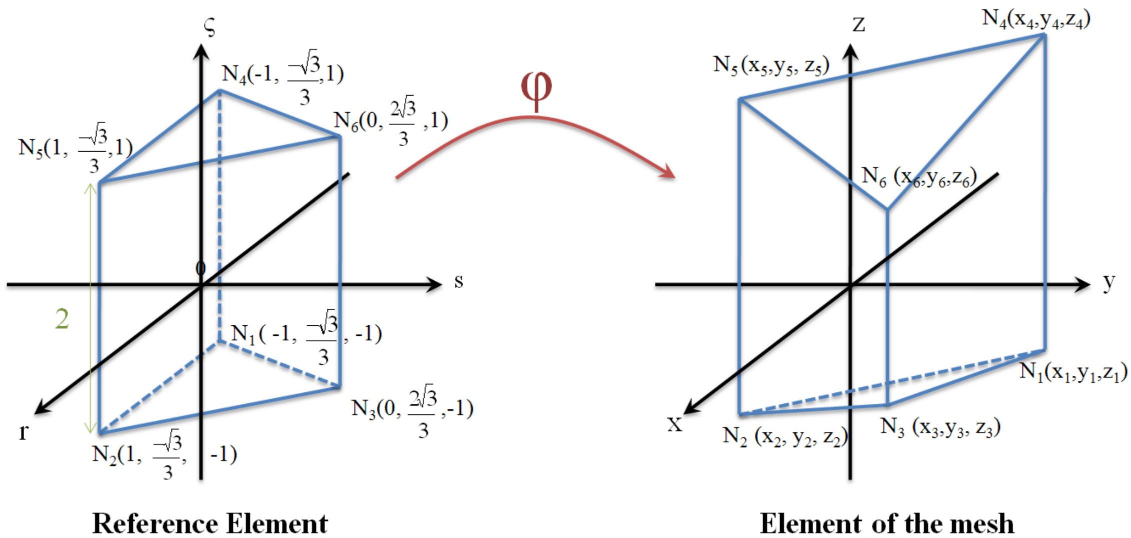


Figure A.5: Transformation from the reference element, to the current pentahedron of the mesh

For any function $f(x, y, z)$, the transformation verifies:

$$\iiint_E f(x,y,z)dV = \iiint_{\hat{E}} \hat{f}(r,s,\zeta)|J|d\hat{V} \tag{A.21}$$

Where J is the Jacobian determinant of the φ function. For the reference element $\hat{E}$, the nodal functions are:

$$\begin{aligned}
L_1(r,s,\zeta) &= \left(-\frac{1}{2}r - \frac{\sqrt{3}}{6}s + \frac{1}{3}\right)\frac{1-\zeta}{2} \\
L_2(r,s,\zeta) &= \left(\frac{1}{2}r - \frac{\sqrt{3}}{6}s + \frac{1}{3}\right)\frac{1-\zeta}{2} \\
L_3(r,s,\zeta) &= \left(\frac{\sqrt{3}}{3}s + \frac{1}{3}\right)\frac{1-\zeta}{2} \\
L_4(r,s,\zeta) &= \left(-\frac{1}{2}r - \frac{\sqrt{3}}{6}s + \frac{1}{3}\right)\frac{\zeta+1}{2} \\
L_5(r,s,\zeta) &= \left(\frac{1}{2}r - \frac{\sqrt{3}}{6}s + \frac{1}{3}\right)\frac{\zeta+1}{2} \\
L_6(r,s,\zeta) &= \left(\frac{\sqrt{3}}{3}s + \frac{1}{3}\right)\frac{\zeta+1}{2}
\end{aligned} \tag{A.22}$$

One can use these nodal functions to calculate the function φ:

$$\varphi(r,s) = \begin{cases} x = \sum_{i=1}^{6} x_i L_i(r,s,\zeta) \\ \\ y = \sum_{i=1}^{6} y_i L_i(r,s,\zeta) \\ \\ z = \sum_{i=1}^{6} z_i L_i(r,s,\zeta) \end{cases} \tag{A.23}$$

For example, the first component of $\varphi(r,s,\zeta)$ is:

$$\begin{aligned}\varphi_x(r,s,t) = &\frac{1}{6}\left(x_1+x_2+x_3+x_4+x_5+x_6\right)+\frac{1}{4}\left(-x_1+x_2-x_4+x_5\right)r \\ &+\frac{\sqrt{3}}{12}\left(-x_1-x_2+2x_3-x_4-x_5+2x_6\right)s+\frac{1}{6}\left(-x_1-x_2-x_3+x_4+x_5+x_6\right)\zeta \\ &+\frac{1}{4}\left(x_1-x_2-x_4+x_5\right)r\zeta+\frac{\sqrt{3}}{12}\left(x_1+x_2-2x_3-x_4-x_5+2x_6\right)s\zeta\end{aligned}$$

The 2 other components have exactly the same equation, except the the x_i must be replaced by in turn y_i and z_i. The Jacobian of the φ function is now easily computable:

$$J = \begin{bmatrix} \frac{\partial \varphi_x}{\partial r} & \frac{\partial \varphi_x}{\partial s} & \frac{\partial \varphi_x}{\partial \zeta} \\ \\ \frac{\partial \varphi_y}{\partial r} & \frac{\partial \varphi_y}{\partial s} & \frac{\partial \varphi_y}{\partial \zeta} \\ \\ \frac{\partial \varphi_z}{\partial r} & \frac{\partial \varphi_z}{\partial s} & \frac{\partial \varphi_z}{\partial \zeta} \end{bmatrix} \tag{A.24}$$

with:

$$\begin{aligned}
J_{11} &= \frac{1}{4}\left(-x_1+x_2-x_4+x_5\right)+\frac{1}{4}\left(x_1-x_2-x_4+x_5\right)\zeta \\
J_{21} &= \frac{1}{4}\left(-y_1+y_2-y_4+y_5\right)+\frac{1}{4}\left(y_1-y_2-y_4+y_5\right)\zeta \\
J_{31} &= \frac{1}{4}\left(-z_1+z_2-z_4+z_5\right)+\frac{1}{4}\left(z_1-z_2-z_4+z_5\right)\zeta \\
J_{12} &= \frac{\sqrt{3}}{12}\left(-x_1-x_2+2x_3-x_4-x_5+2x_6\right)+\frac{\sqrt{3}}{12}\left(x_1+x_2-2x_3-x_4-x_5+2x_6\right)\zeta \\
J_{22} &= \frac{\sqrt{3}}{12}\left(-y_1-y_2+2y_3-y_4-y_5+2y_6\right)+\frac{\sqrt{3}}{12}\left(y_1+y_2-2y_3-y_4-y_5+2y_6\right)\zeta \\
J_{32} &= \frac{\sqrt{3}}{12}\left(-z_1-z_2+2z_3-z_4-z_5+2z_6\right)+\frac{\sqrt{3}}{12}\left(z_1+z_2-2z_3-z_4-z_5+2z_6\right)\zeta
\end{aligned} \tag{A.25}$$

$$J_{13} = \frac{1}{4}\left(x_1 - x_2 - x_4 + x_5\right) r + \frac{\sqrt{3}}{12}\left(x_1 + x_2 - 2x_3 - x_4 - x_5 + 2x_6\right) s + \frac{1}{6}\left(-x_1 - x_2 - x_3 + x_4 + x_5 + x_6\right)$$

$$J_{32} = \frac{1}{4}\left(y_1 - y_2 - y_4 + y_5\right) r + \frac{\sqrt{3}}{12}\left(y_1 + y_2 - 2y_3 - y_4 - y_5 + 2y_6\right) s + \frac{1}{6}\left(-y_1 - y_2 - y_3 + y_4 + y_5 + y_6\right)$$

$$J_{33} = \frac{1}{4}\left(z_1 - z_2 - z_4 + z_5\right) r + \frac{\sqrt{3}}{12}\left(z_1 + z_2 - 2z_3 - z_4 - z_5 + 2z_6\right) s + \frac{1}{6}\left(-z_1 - z_2 - z_3 + z_4 + z_5 + z_6\right)$$

A.6 MINI element

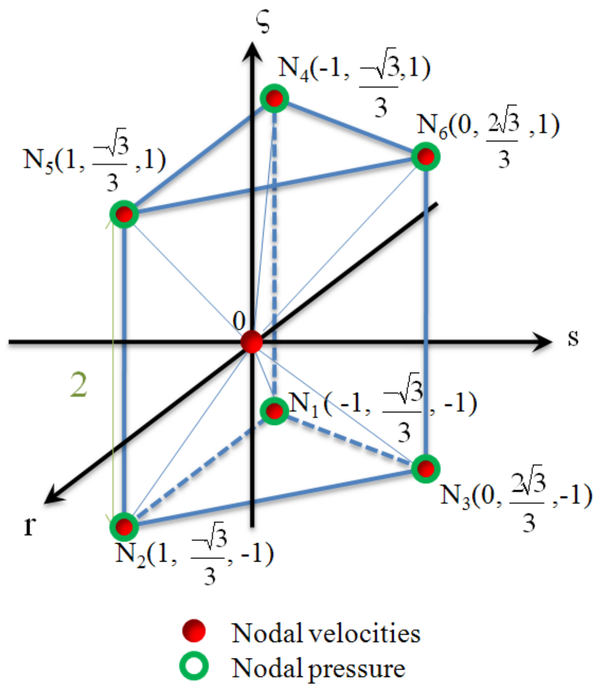


Figure A.6: MINI element

The MINI element is an extruded triangular element with continuous velocity and pressure approximations. Introduced by Arnold et al. (1984), the MINI element satisfies the LLB conditions (Ladyzhenskaya-Babuska-Brezzi, See Nowicki [9] for more details), and therefore results in stable velocities and pressure.

The MINI element is based on a extruded straight-sided triangular element (called P1-P1), which has a node at each vertex. In the MINI element, the pressure and velocity are approximated via

$$\begin{aligned} P(x,y,z) &= \sum_{i=1}^{6} P_j \phi_j(x,y,z) \\ u(x,y,z) &= \sum_{i=1}^{6} u_j \phi_j(x,y,z) + u_b \phi_b(x,y,z) \\ v(x,y,z) &= \sum_{i=1}^{6} v_j \phi_j(x,y,z) + v_b \phi_b(x,y,z) \\ w(x,y,z) &= \sum_{i=1}^{6} w_j \phi_j(x,y,z) + w_b \phi_b(x,y,z) \end{aligned} \tag{A.26}$$

The ϕ_j functions has the same shape as in the pentahedron (A.5). Therefore, the transformation φ that transforms the reference element to the element E has exactly the same shape as inthe previous section (A.5).

The nodal function $\phi_b(x,y,z)$ is called bubble function. According to Nowicki [9] and Guillem [2], the bubble shape function in a reference element $L_b(x,y,z)$ is:

$$L_b\left(x,y,z\right) = 27\left(-\frac{1}{2}r - \frac{\sqrt{3}}{6}s + \frac{1}{3}\right)\left(\frac{1}{2}r - \frac{\sqrt{3}}{6}s + \frac{1}{3}\right)\left(\frac{\sqrt{3}}{3}s + \frac{1}{3}\right)\left(\zeta + 1\right)\left(1 - \zeta\right) \tag{A.27}$$

$\phi_b\left(x,y,z\right)$ is equal to 1 at the center of the element (due to the factor 27), and vanishes at the element boundary.

Appendix B

Parameters and constants

Parameter	Symbol	Unit	Value
Glacier's lower surface z-coordinate	b	m	
Glen's flow law viscosity parameter	B	MPa $a^{1/n}$	
Ice specific heat capacity	c	J kg^{-1} K^{-1}	2093
Mixed layer specific heat	c_{pM}	J kg^{-1} K^{-1}	3974
Acceleration due to gravity	g	m s^{-1}	9.81
Geothermal heat flux	G	W m^{-2}	$\sim$0.05
Glacier's Height	H	m	
Ice thermal conductivity	k	W m^{-1} K^{-1}	2.4
Basal drag parameter	K	$Pa^{\frac{1-r}{2}}$ $s^{\frac{s}{2}}$ $m^{-\frac{s}{2}}$	
Ice specific latent heat of fusion	L	J kg^{-1}	3.34×10^5
Mass production term (m/a ice equivalent)	$\dot{M}$	m a^{-1}	
Melting (m/a ice equivalent)	$\dot{M}_b$	m a^{-1}	
Accumulation (m/a ice equivalent)	$\dot{M}_s$	m a^{-1}	
Glen's flow law exponent	n	dimensionless	
Normal vector x-component	n_x	dimensionless	
Normal vector y-component	n_y	dimensionless	
Normal vector z-component	n_z	dimensionless	
Normal vector pointing outward from the galcier	$\overrightarrow{n}$	dimensionless	
Effective pressure	N_{eff}	Pa	
Basal drag first exponent	p	dimensionless	
Ice pressure	P	Pa	
Atmosphere pressure	P_{air}	Pa	
Water pressure	P_w	Pa	
Basal drag second exponent	q	dimensionless	
Glacier's upper surface z-coordinate	s	m	
Ice temperature	T	K	
Glacier base temperature	T_b	K	
Pressure melting point	T_{pmp}	K	
Ice velocity x-component	u	m s^{-1}	
Depth averaged ice velocity x-component	$\overline{u}$	m s^{-1}	
Basal velocity x-component	u_b	m s^{-1}	
Basal velocity parallel to the bedrock surface	$\overrightarrow{u_b}$	m s^{-1}	
Ice velocity y-component	v	m s^{-1}	
Depth averaged ice velocity y-component	$\overline{v}$	m s^{-1}	
Basal velocity y-component	v_b	m s^{-1}	
Ice velocity	$\overrightarrow{v}$	m s^{-1}	
Ice velocity z-component	w	m s^{-1}	
Basal velocity z-component	w_b	m s^{-1}	
First horizontal coordinate	x	m	

Second horizontal coordinate	y	m	
Vertical coordinate	z	m	
Rate of change of melting point with pressure	β	K Pa^{-1}	9.8×10^{-8}
Thermal exchange velocity	γ	m s^{-1}	$\sim 1.00 \times 10^{-4}$
Strain tensor	ε	dimensionless	
Strain rate tensor	$\dot{\varepsilon}$	s^{-1}	
Effective strain rate tensor	$\dot{\varepsilon}_e$	s^{-1}	
Ice viscosity	μ	Pa s	
Viscous heating	Φ	W m^{-3}	
Ice density	ρ	kg m^{-3}	916
Ice density	ρ_i	kg m^{-3}	916
Water density	ρ_w	kg m^{-3}	1000
Stress tensor	σ	Pa	
Deviatoric stress tensor	σ'	Pa	
Effective shear stress	σ_e	Pa	
Friction stress	$\overrightarrow{\tau_b}$	Pa	

Bibliography

[1] Glen J. W., *The creep of polycrystalline ice.* Proceedings of the Royal Society, London, 1955.

[2] Guillén-Gonzàlez F., Rodríguez-Gómez D., *Bubble finite elements for the primitive equations of the ocean*, Numerische Mathematik 101: 689-728, 2005.

[3] Holland D.M., Jenkins A., *Modeling Thermodynamic Ice-Ocean Interactions at the Base of an Ice Shelf.* J. Phys. Oceanogr., 29, 1787-1800, 1999.

[4] Hooke Roger LeB., *Principles of Glacier Mechanics.* 2d Edition,Cambridge University Press, 2005.

[5] Hulbe, Christina L., MacAyeal Douglas *A new numerical model of coupled inland ice sheet, ice stream, and ice shelf flow and its application to the West Antarctique Ice Sheet.* Journal of Geophysical Research, vol 104, B11, pages 25,349-25,366, November 10th, 1999.

[6] Larour Éric, *Modélisation numerique du comportement des banquises flottantes, validée par imagerie satellitaire.* Thèse de l'école doctorale de l'École Centrale Paris, 2005.

[7] MacAyeal Douglas R., *Large-scale Ice Flow Over a Viscous Basal Sediment: Theory and Application* . Journal of Geophysical Research-Solid Earth and Planets, 1989.

[8] MacAyeal Douglas R., *EISMINT: Lessons in Ice-Sheet Modelling* . 21st May 1997. http://homepages.vub.ac.be/~phuybrec/pdf/MacAyeal.lessons.pdf

[9] Nowicki Sophie M. J., *Modelling the transition zone of marine ice sheets*, PhD Thesis of University College London, Centre for Polar Observation and Modelling Department of Space and Climate Physics, September 2007.

[10] Paterson W. S. B., *The Physics of glaciers.* 3ieme Edition, Butterworth Heinemann, 1994.

[11] Pattyn F., *A new three-dimensional higher-order thermomechanical ice sheet model: basic sensitivity, ice stream development and ice flow across subglacial lakes* Journal of Geophysical Research (Solid Earth), 2003

[12] Wriggers P., *Computational Contact Mechanics* Springers 2d edition, 2002

[13] Zienkiewicz, O.C. and Taylor R.L., *The Finite Element Method* Fourth Edition. McGraw-Hill Book Company Europe. Berkshire England., 1994.