



**A linear algorithm for  
solving ice shelf  
equations**

A. Sargent and  
J. L. Fastook

# A linear algorithm for solving non-linear isothermal ice-shelf equations

A. Sargent<sup>1</sup> and J. L. Fastook<sup>2</sup>

<sup>1</sup>University of Maine, Department of Mathematics and Statistics, Orono, USA

<sup>2</sup>University of Maine, Department of Computer Science, Orono, USA

Received: 14 February 2014 – Accepted: 3 March 2014 – Published: 18 March 2014

Correspondence to: A. Sargent (aitbala.sargent@umit.maine.edu)

Published by Copernicus Publications on behalf of the European Geosciences Union.

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

# Abstract

A linear non-iterative algorithm is suggested for solving nonlinear isothermal steady-state Morland–MacAyeal ice shelf equations. The idea of the algorithm is in replacing the problem of solving the non-linear second order differential equations for velocities with a system of linear first order differential equations for stresses. The resulting system of linear equations can be solved numerically with direct methods which are faster than iterative methods for solving corresponding non-linear equations. The suggested algorithm is applicable if the boundary conditions for stresses can be specified. The efficiency of the linear algorithm is demonstrated for one-dimensional and two-dimensional ice shelf equations by comparing the linear algorithm and the traditional iterative algorithm on derived manufactured solutions. The linear algorithm is shown to be as accurate as the traditional iterative algorithm but significantly faster. The method may be valuable as the way to increase the efficiency of complex ice sheet models a part of which requires solving the ice shelf model as well as to solve efficiently two-dimensional ice-shelf equations.

# 1 Introduction

Most large-scale ice sheet models based either on shallow-ice approximation or shallow shelf approximation of the ice momentum equation (Gagliardini and Zwinger, 2008; Rutt et al., 2009; Larour et al., 2012; Winkelmann et al., 2011; Lipscomb et al., 2009). Derivation of the ice shelf model, called Morland–MacAyeal (MM) ice shelf model, can be found, for example, in Morland (1987) or MacAyeal (1989). The isothermal Morland–MacAyeal ice shelf model is a system of non-linear partial differential equations. Because of their non-linearity, the modelers use iterative methods to solve the equations numerically. In this work, we present a linear non-iterative algorithm for the MM ice shelf equations. The idea of the algorithm is to replace the problem of solving the non-linear second order differential equations for velocities with a system of linear first

GMDD

7, 1829–1864, 2014

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



order differential equations for stresses. The system of linear differential equations can be solved numerically with direct methods which are significantly faster than iterative methods for non-linear equations. The assumption of availability of boundary conditions for stresses – the assumption that they are known or can be supplied from other sources – is a limitation of this model.

To illustrate the algorithm, we present its application for solving one-dimensional and two-dimensional ice-shelf models. The accuracy and the speed of the algorithm is examined on manufactured analytical solutions of the models. Following Bueler et al. (2005, 2007) and Sargent and Fastook (2010), the analytical solutions are built by constructing a solution and then modifying the right hand side functions of the equations to satisfy the constructed solution.

The paper is organized as follows. In Sect. 2, we present the linear algorithm for solving the one-dimensional (1-D) MM model and test its accuracy and efficiency on a constructed manufactured solution. In Sect. 3, we present the linear algorithm for solving the two-dimensional (2-D) MM model and test its accuracy and efficiency on a corresponding constructed solution. In both 1-D and 2-D cases, we compare the direct method of solving the MM model with a traditional iterative method. In 2-D case, we abstract the effect of the boundary conditions on the method efficiency by considering Dirichlet boundary conditions. In Sect. 3, we provide some concluding remarks. In the appendices we give the formulas that can be used to calculate the right hand side functions of the 2-D equations that satisfy the manufactured solutions and the description of Fast Fourier Transformation method used to solve equations generated by the 2-D direct method. Finally, supplements to the manuscript contain Fortran 77 code to solve 1-D and 2-D ice shelf equations with the linear and traditional iterative methods.

## GMDD

7, 1829–1864, 2014

### A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

## 2 Linear algorithm for solving one-dimensional steady-state ice shelf equation

### 2.1 One-dimensional steady-state ice shelf momentum equation

We consider the following one-dimensional steady-state flowline ice-shelf model with Dirichlet boundary conditions at the upstream of the shelf and the Newman boundary condition at the calving front of the shelf:

$$\begin{aligned} \frac{\partial}{\partial \tilde{x}} \left( 2\tilde{\mu}\tilde{h} \frac{\partial \tilde{u}}{\partial \tilde{x}} \right) &= \rho g \tilde{h} \frac{\partial \tilde{s}}{\partial \tilde{x}}, & 0 \leq \tilde{x} \leq L, \\ \tilde{u} &= U, & \text{at } \tilde{x} = 0, \\ 4\tilde{\mu}\tilde{h} \frac{\partial \tilde{u}}{\partial \tilde{x}} &= \frac{\rho g \tilde{h}^2}{2} \left( 1 - \frac{\rho}{\rho_w} \right), & \text{at } \tilde{x} = L, \end{aligned} \quad (1)$$

where  $\tilde{x}$  is  $x$  coordinate,  $\tilde{u}$  is velocity,  $\tilde{h}$  is ice thickness,  $\tilde{s}$  is ice surface,  $\rho$ ,  $\rho_w$ ,  $g$ ,  $B$  are ice and water density, the gravitational constant, and ice stiffness parameter, and  $\tilde{\mu}$  is the effective viscosity defined as

$$\tilde{\mu} = B \left| \frac{\partial \tilde{u}}{\partial \tilde{x}} \right|^{\frac{1-n}{n}}. \quad (2)$$

Following MacAyeal (1997), we non dimensionalize these equations using the following typical values:  $Z$  – the mean thickness of the ice-shelf,  $L$  – the length of the ice-shelf,  $U$  – the typical velocity of the ice-shelf, and the following non-dimensional variables (variables without tilde):

$$\tilde{s} = Zs, \quad \tilde{h} = Zh, \quad \tilde{x} = Lx, \quad \tilde{u} = Uu, \quad \tilde{\mu} = B \left( \frac{U}{L} \right)^{\frac{1-n}{n}} \mu$$

The scale factors  $L$ ,  $U$  and  $Z$  are chosen to satisfy the relationship:

$$2B \left( \frac{U}{L} \right)^{\frac{1}{n}} = \rho g Z.$$

GMDD

7, 1829–1864, 2014

### A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Then, the nondimensional ice-shelf equation and boundary conditions are written as:

$$\frac{\partial}{\partial x} \left( \mu h \frac{\partial u}{\partial x} \right) = h \frac{\partial s}{\partial x}, \quad 0 \leq x \leq 1, \quad (3)$$

$$u = 1, \quad \text{at } x = 0, \quad (4)$$

$$\mu h \frac{\partial u}{\partial x} = \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) h^2, \quad \text{at } x = 1. \quad (5)$$

where the non-dimensional effective viscosity is  $\mu = \left| \frac{\partial u}{\partial x} \right|^{\frac{1-n}{n}}$ .

## 2.2 Linear algorithm

Equations (3)–(5) are non-linear equations and solved using iterative methods. To construct a linear non-iterative algorithm, we introduce a new variable  $\tau$ , which we call a stress variable:

$$\tau = \mu h \frac{\partial u}{\partial x} = h \left( \frac{\partial u}{\partial x} \right)^{\frac{1}{n}}. \quad (6)$$

Then Eqs. (3) and (5) in terms of the new variable  $\tau$  are written as:

$$\frac{\partial \tau}{\partial x} = h \frac{\partial s}{\partial x}, \quad 0 \leq x \leq 1, \quad (7)$$

$$\tau = \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) h^2, \quad \text{at } x = 1.$$

Variable  $\tau$  can be found by solving linear Eq. (7). After  $\tau$  is found, velocity  $u$  can be found solving the linear Eq. (6) with the boundary condition at the upstream (4):

$$\frac{\partial u}{\partial x} = \left( \frac{\tau}{h} \right)^n, \quad 0 \leq x \leq 1, \quad (8)$$

$$u = 1, \quad \text{at } x = 0.$$

Thus, instead of solving a non-linear Eq. (3), this algorithm consists of solving two linear first order differential Eqs. (7) and (8).

## 2.3 Manufactured solution

- 5 To check the applicability and the efficiency of the method, we generate a manufactured analytical solution of Eqs. (3)–(5) with modified right-hand side:

$$\frac{\partial}{\partial x} \left( \mu h \frac{\partial u}{\partial x} \right) = h \frac{\partial s}{\partial x} + f_1(x), \quad 0 \leq x \leq 1, \quad (9)$$

$$u = 1, \quad \text{at } x = 0, \quad (10)$$

$$\mu h \frac{\partial u}{\partial x} = \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) h^2 + f_2(x), \quad \text{at } x = 1. \quad (11)$$

where functions  $f_1(x)$  and  $f_2(x)$  are defined by substituting the desired solution into Eqs. (9) and (11).

The modified equations for new variable  $\tau$ , corresponding to Eqs. (9)–(11), are

$$\frac{\partial \tau}{\partial x} = h \frac{\partial s}{\partial x} + f_1, \quad 0 \leq x \leq 1, \quad (12)$$

$$\tau = \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) h^2 + f_2, \quad \text{at } x = 1. \quad (13)$$

To generate a particular solution, assume a flow with a linear sloping surface with a mean slope  $\alpha$  and a sinusoidal bed with frequency  $\omega$ :

$$s(x) = -\delta x \cdot \tan(\alpha), \quad b(x) = s(x) - 1 + a_b \sin^2(\omega x). \quad (13)$$

- 20 These surface and bed generate thinning ice thickness:

$$h(x) = s(x) - b(x) = 1 - a_b \sin^2(\omega x). \quad (14)$$

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

To satisfy the mass conservation equation, we construct ice velocity for a given ice thickness and a bed profile as follows:

$$u(x) = \frac{c_x}{h(x)}. \quad (15)$$

When the parameter  $c_x = 1$ , constructed velocity function (15) satisfies boundary condition (10).

For calculation of the right-hand side functions in Eqs. (9) and (11), we need velocity and ice thickness derivatives:

$$\begin{aligned} \frac{\partial h}{\partial x} &= -a_b \omega \sin(2\omega x), & \frac{\partial^2 h}{\partial x^2} &= -2a_b \omega^2 \cos(2\omega x), \\ \frac{\partial u}{\partial x} &= -c_x \frac{\frac{\partial h}{\partial x}}{h^2}, & \frac{\partial^2 u}{\partial x^2} &= \frac{c_x}{h^2} \left[ \frac{2}{h} \left( \frac{\partial h}{\partial x} \right)^2 - \frac{\partial^2 h}{\partial x^2} \right]. \end{aligned}$$

Then the right-hand sides of the modified Eqs. (9)–(11) are calculated as follows:

$$f_1 = \mu \left[ \frac{h}{n} \frac{\partial^2 u}{\partial x^2} + \frac{\partial h}{\partial x} \frac{\partial u}{\partial x} \right] - h \frac{\partial s}{\partial x}, \quad f_2 = \mu h \frac{\partial u}{\partial x} - \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) h^2. \quad (16)$$

## 2.4 Solving one-dimensional ice-shelf equation with linear algorithm

We used the finite difference method to solve the ice-shelf equations with the new algorithm. All variables and parameters are one-dimensional arrays. We use an even staggered grid with step  $\Delta x$ . Velocity  $u$  and ice-thickness  $h$  are defined at the nodes of the grid and new variable  $\tau$  is defined at the centroids of the grid as shown in Fig. 1 (left grid). To define the boundary condition at the right-hand side of the grid for  $\tau$ , assume that the last value of  $\tau$  is defined at the node  $x = 1$ .

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

To calculate  $\tau$ , we approximate Eq. (7) with the second order scheme as follows:

$$\begin{aligned}\tau_N &= \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) (h_N)^2 + (f_2)_N, \\ \frac{\tau_N - \tau_{N-1}}{(\Delta x)/2} &= \left( \frac{3}{4} h_N + \frac{1}{4} h_{N-1} \right) \frac{5s_N - 6s_{N-1} + s_{N-2}}{4\Delta x} + \left( \frac{3}{4} (f_1)_N + \frac{1}{4} (f_1)_{N-1} \right), \\ \frac{\tau_i - \tau_{i-1}}{\Delta x} &= h_i \frac{s_{i+1} - s_{i-1}}{2\Delta x} + f_{1i} \quad i = N-1, N-2, \dots, 2.\end{aligned}\quad (17)$$

After finding  $\tau$ , velocity  $u$  is found by approximating Eq. (8):

$$\begin{aligned}u_1 &= 1, \\ \frac{u_{i+1} - u_i}{\Delta x} &= \left( \frac{\tau_i}{\frac{h_{i+1} + h_i}{2}} \right)^n \quad i = 1, 2, \dots, N-1.\end{aligned}\quad (18)$$

## 2.5 Solving one-dimensional ice-shelf equation with traditional iterative method

To compare the new algorithm with the traditional iterative algorithm, we approximated Eqs. (9)–(11) with traditional three-point finite difference scheme:

$$\begin{aligned}\frac{1}{\Delta x} \left[ \mu(u_{i+1/2}) h_{i+1/2} \frac{u_{i+1} - u_i}{\Delta x} - \mu(u_{i-1/2}) h_{i-1/2} \frac{u_i - u_{i-1}}{\Delta x} \right] \\ = h_i \frac{s_{i+1} - s_{i-1}}{2\Delta x} + f_{1i}, \quad i = 2, 3, \dots, N-1, \\ u_1 = 1, \\ \mu(u_{N-1/2}) h_{N-1/2} \frac{u_N - u_{N-1}}{\Delta x} = \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) h_{N-1/2}^2 + (f_2)_{N-1/2}.\end{aligned}\quad (19)$$

## GMDD

7, 1829–1864, 2014

### A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Equation (19) is solved iteratively using values of  $u^k$  at the previous iterations  $k$  to estimate values of  $\mu(u)$  at iteration  $k + 1$ :

$$\begin{aligned} & \frac{1}{\Delta x} \left[ \mu(u_{i+1/2}^k) h_{i+1/2} \frac{u_{i+1}^{k+1} - u_i^{k+1}}{\Delta x} - \mu(u_{i-1/2}^k) h_{i-1/2} \frac{u_i^{k+1} - u_{i-1}^{k+1}}{\Delta x} \right] \\ & = h_i \frac{s_{i+1} - s_{i-1}}{2\Delta x} + f_{1i}, \quad i = 2, 3, \dots, N-1, \\ & u_1^{k+1} = 1, \\ & \mu(u_{N-1/2}^k) h_{N-1/2} \frac{u_N^{k+1} - u_{N-1}^{k+1}}{\Delta x} = \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) h_{N-1/2}^2 + (f_2)_{N-1/2}. \end{aligned} \quad (20)$$

At each iteration, Eq. (20) is solved using a direct method for solving a system with a three-diagonal matrix.

We stopped iterations when the error, calculated as  $\|e^{k+1}\|_{L_2}^2 = \frac{1}{N} \sum_{i=1, N} (u_i^{k+1} - u_i^k)^2$ , started increasing.

## 2.6 Comparing the direct and iterative algorithms for one-dimensional ice-shelf problem

We test the performance of the direct and iterative algorithms using the constructed manufactured solution with the following ice constants: ice density  $\rho = 910 \text{ kg m}^{-3}$ , gravitational constant  $g = 9.81 \text{ m s}^{-2}$ , exponent in Glen's flow law  $n = 3$  and the following parameters: the scale parameter  $\delta = \frac{L}{Z} = \frac{80000}{1000}$ , basal bumps frequency  $\omega = \pi/2$ , the surface incline angle  $\alpha = 0.1^\circ$ , the velocity parameter  $c_x = 1$ , and the amplitude of the bed undulation  $a_b = 0.5$ .

Six sets of uniform grids with 100, 1000, 10000, 100000, 1000000, and 10000000 nodes were used. The ice bed and surface profiles, analytical solutions (velocity and

# GMDD

7, 1829–1864, 2014

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

stress) and solutions calculated by iterative and direct methods for the grid with 100 nodes are shown in Fig. 2.

Table 1 and Fig. 3 show time it took for the methods to solve the equation, the number of iterations for the iterative method, the error in calculation of velocity ( $u$ ) and stress ( $\tau$ ), and the rate of convergence. The error measure is the  $L_2$  norm of the solution absolute error. The convergence rates are calculated between consecutive pairs of grids as  $\log_{10} \left( \frac{e_{i-1}}{e_i} \right)$ , where  $e_{i-1}, e_i$  are the errors on a series of uniformly refined grids.

The rates of convergence are about 2 for the velocity and about 0.83 for the stress. The error of the iterative method stops declining after the number of grid nodes reaches 1 000 000. We believe that it is due accumulation of rounding errors. For the grids with number of nodes up to 10 000, the accuracy of calculations of both velocity and stress is about the same for the traditional iterative method and newly proposed linear method. For more refined grids, the accuracy of the iterative method declines while the direct method continues to converge with the same rate as before. The linear method is about hundred (100) times faster than the iterative method.

### 3 Non-iterative linear algorithm for solving two-dimensional Morland–MacAyeal ice shelf equations

#### 3.1 Two-dimensional Morland–MacAyeal ice shelf equations

The steady-state MM diagnostic equations (the  $\tilde{x}$ - and  $\tilde{y}$ -components) in terms of depth-averaged horizontal velocities are written as follows:

$$\begin{aligned} \frac{\partial}{\partial \tilde{x}} \left( 2\tilde{\mu}\tilde{h} \left( 2\frac{\partial \tilde{u}}{\partial \tilde{x}} + \frac{\partial \tilde{v}}{\partial \tilde{y}} \right) \right) + \frac{\partial}{\partial \tilde{y}} \left( \tilde{\mu}\tilde{h} \left( \frac{\partial \tilde{u}}{\partial \tilde{y}} + \frac{\partial \tilde{v}}{\partial \tilde{x}} \right) \right) &= \rho g \tilde{h} \frac{\partial \tilde{s}}{\partial \tilde{x}} + \tilde{f}_{bx}, \\ \frac{\partial}{\partial \tilde{y}} \left( 2\tilde{\mu}\tilde{h} \left( 2\frac{\partial \tilde{v}}{\partial \tilde{y}} + \frac{\partial \tilde{u}}{\partial \tilde{x}} \right) \right) + \frac{\partial}{\partial \tilde{x}} \left( \tilde{\mu}\tilde{h} \left( \frac{\partial \tilde{u}}{\partial \tilde{y}} + \frac{\partial \tilde{v}}{\partial \tilde{x}} \right) \right) &= \rho g \tilde{h} \frac{\partial \tilde{s}}{\partial \tilde{y}} + \tilde{f}_{by}, \end{aligned} \quad (21)$$

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

where  $f_{bx}^{\sim}$  and  $f_{by}^{\sim}$  are terms approximating the basal drag in the case of the ice stream flow, and  $\tilde{\mu}$  is the depth-averaged effective viscosity:

$$\tilde{\mu} = \frac{B}{2} \left[ \left( \frac{\partial \tilde{u}}{\partial \tilde{x}} \right)^2 + \left( \frac{\partial \tilde{v}}{\partial \tilde{y}} \right)^2 + \frac{1}{4} \left( \frac{\partial \tilde{u}}{\partial \tilde{y}} + \frac{\partial \tilde{v}}{\partial \tilde{x}} \right)^2 + \frac{\partial \tilde{u}}{\partial \tilde{x}} \frac{\partial \tilde{v}}{\partial \tilde{y}} \right]^{\frac{1-n}{2n}}. \quad (22)$$

Two types of boundary conditions can be specified along the edge of the domain, Dirichlet and Neumann. Dirichlet boundary conditions, specification of the depth-averaged velocity, are applied at zero slip coast-lines or where ice streams flow into the ice shelf or at stagnant ice-shelf boundaries. Neumann boundary conditions are specified at the seaward, iceberg-caving front. The depth-integrated balance of forces at the ice front is formulated as a balance of the depth-integrated force transmitted across the ice front due to internal stresses and the integral of the hydrostatic pressure in the seawater beyond the ice front over the face of the ice front.

In the cases when the ice front extends along the  $\tilde{y}$ - and  $\tilde{x}$ -axes, that is, when  $n = n_{\tilde{x}}$  and  $n = n_{\tilde{y}}$  consequently, these boundary conditions can be written as (MacAyeal, 1997):

$$\begin{aligned} n = n_{\tilde{x}} : 2\tilde{\mu}\tilde{h} \left( 2\frac{\partial \tilde{u}}{\partial \tilde{x}} + \frac{\partial \tilde{v}}{\partial \tilde{y}} \right) &= \frac{\rho g \tilde{h}^2}{2} \left( 1 - \frac{\rho}{\rho_w} \right), \\ n = n_{\tilde{y}} : 2\tilde{\mu}\tilde{h} \left( \frac{\partial \tilde{u}}{\partial \tilde{x}} + 2\frac{\partial \tilde{v}}{\partial \tilde{y}} \right) &= \frac{\rho g \tilde{h}^2}{2} \left( 1 - \frac{\rho}{\rho_w} \right). \end{aligned} \quad (23)$$

To non-dimensionalize variables, we choose the following typical values:  $Z$  – the mean thickness of the ice-sheet,  $L$  – the length of ice-sheet,  $U$  – a typical velocity in the horizontal direction, and introduce the following non-dimensional variables (variables without tilde):

$$\tilde{s} = Zs, \quad \tilde{b} = Zb, \quad \tilde{x} = Lx, \quad \tilde{y} = Ly, \quad \tilde{u} = Uu, \quad \tilde{v} = Uv, \quad \tilde{\mu} = \frac{B}{2} \left( \frac{U}{L} \right)^{\frac{1-n}{n}} \mu$$

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

and require that scale factors  $L$ ,  $U$  and  $Z$  satisfy the relationship:

$$\frac{B}{2} \left( \frac{U}{L} \right)^{\frac{1}{n}} = \rho g Z.$$

Then the nondimensional momentum equation is written as:

$$\frac{\partial}{\partial x} \left( 2\mu h \left( 2\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \right) + \frac{\partial}{\partial y} \left( \mu h \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) = h \frac{\partial s}{\partial x} + f_{bx}, \quad (24)$$

$$\frac{\partial}{\partial y} \left( 2\mu h \left( 2\frac{\partial v}{\partial y} + \frac{\partial u}{\partial x} \right) \right) + \frac{\partial}{\partial x} \left( \mu h \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) = h \frac{\partial s}{\partial y} + f_{by},$$

where  $f_{bx}$  and  $f_{by}$  are non-dimensional terms approximating the basal drag in the case of the ice stream flow, and the non-dimensional depth-averaged effective viscosity  $\mu$  is:

$$\mu = \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 + \frac{1}{4} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 + \frac{\partial u}{\partial x} \frac{\partial v}{\partial y} \right]^{\frac{1-n}{2n}}. \quad (25)$$

Newman boundary conditions in non-dimensional variables become:

$$n = n_x : \mu h \left( 2\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) = \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) h^2, \quad (26)$$

$$n = n_y : \mu h \left( \frac{\partial u}{\partial x} + 2\frac{\partial v}{\partial y} \right) = \frac{1}{4} \left( 1 - \frac{\rho}{\rho_w} \right) h^2.$$

### 3.2 Linear algorithm

Non-linear Eq. (24) is solved using iterative methods. To construct a non-iterative algorithm, we introduce new variables (we will call them stresses):

$$\tau_x = 2\mu h \frac{\partial u}{\partial x}, \quad (27)$$

## GMDD

7, 1829–1864, 2014

### A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



$$\tau_y = \mu h \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right). \quad (28)$$

In terms of the new variables Morland–MacAyeal diagnostic Eqs. (24), after using incompressibility condition  $\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$ , are:

$$5 \quad \frac{\partial \tau_x}{\partial x} + \frac{\partial \tau_y}{\partial y} = h \frac{\partial z_s}{\partial x} + f_{bx}, \quad (29)$$

$$\frac{\partial \tau_y}{\partial x} - \frac{\partial \tau_x}{\partial y} = h \frac{\partial z_s}{\partial y} + f_{by}, \quad (30)$$

And the Newmann boundary conditions along the iceberg-calving front are:

$$n = n_x: \quad \tau_x = \frac{1}{8} \left( 1 - \frac{\rho}{\rho_w} \right) h^2, \quad (31)$$

$$10 \quad n = n_y: \quad \tau_x = -\frac{1}{8} \left( 1 - \frac{\rho}{\rho_w} \right) h^2.$$

Since Eqs. (29)–(30) is a system of equations with two variables  $\tau_x$  and  $\tau_y$ , in addition to boundary conditions (31) for  $\tau_x$ , we have to specify boundary conditions for variable  $\tau_y$  as well. The linear algorithm requires additional boundary conditions for stresses.

15 Variables  $\tau_x$  and  $\tau_y$  can be found by solving linear Eqs. (29)–(30) with boundary conditions (31) and some additional boundary conditions for  $\tau_y$ . After  $\tau_x$  and  $\tau_y$  are found, velocity  $u$  and  $v$  can be found from linear Eqs. (27)–(28) which can be re-written as

$$2h \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \frac{1}{4} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 \right]^{\frac{1-n}{2n}} \frac{\partial u}{\partial x} = \tau_x, \quad (32)$$

$$h \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \frac{1}{4} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 \right]^{\frac{1-n}{2n}} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) = \tau_y. \quad (33)$$

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

In the equations above we used the incompressibility condition  $\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$  in the effective viscosity expression.

After simplifying Eqs. (32) and (33) by dividing them by  $2h$  and  $h$  respectively and taking power of  $\frac{2n}{1-n}$ , we receive the following equations for velocities:

$$\left[ \left( \frac{\partial u}{\partial x} \right)^2 + \frac{1}{4} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 \right] \left( \frac{\partial u}{\partial x} \right)^{\frac{2n}{1-n}} = \left( \frac{\tau_x}{2h} \right)^{\frac{2n}{1-n}}, \quad (34)$$

$$\left[ \left( \frac{\partial u}{\partial x} \right)^2 + \frac{1}{4} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 \right] \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^{\frac{2n}{1-n}} = \left( \frac{\tau_y}{h} \right)^{\frac{2n}{1-n}}. \quad (35)$$

with boundary condition at the frozen bed:

$$u = U, \quad v = V. \quad (36)$$

To simplify Eqs. (34) and (35), we introduce new variables:

$$\alpha = \left( \frac{\partial u}{\partial x} \right)^2, \quad (37)$$

$$\beta = \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2. \quad (38)$$

Then, Eqs. (34) and (35) in terms of the variables  $\alpha$  and  $\beta$  are written as:

$$\left[ \alpha + \frac{1}{4}\beta \right] \alpha^{\frac{n}{1-n}} = \left( \frac{\tau_x}{2h} \right)^{\frac{2n}{1-n}}, \quad (39)$$

$$\left[ \alpha + \frac{1}{4}\beta \right] \beta^{\frac{n}{1-n}} = \left( \frac{\tau_y}{h} \right)^{\frac{2n}{1-n}}. \quad (40)$$

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

By dividing Eq. (39) by Eq. (40), we get that

$$\left(\frac{\alpha}{\beta}\right)^{\frac{n}{1-n}} = \left(\frac{\tau_x}{2\tau_y}\right)^{\frac{2n}{1-n}}, \quad (41)$$

or

$$\alpha = \beta \left(\frac{\tau_x}{2\tau_y}\right)^2. \quad (42)$$

Substituting expression for  $\alpha$  from Eq. (42) to (40), we get an equation

$$\left[\left(\frac{\tau_x}{2\tau_y}\right)^2 + \frac{1}{4}\right] \beta^{\frac{1}{1-n}} = \left(\frac{\tau_y}{h}\right)^{\frac{2n}{1-n}}. \quad (43)$$

From Eqs. (43) and (42), we get expressions for  $\beta$  and  $\alpha$ :

$$\beta = \frac{\left(\frac{\tau_y}{h}\right)^{2n}}{\left[\left(\frac{\tau_x}{2\tau_y}\right)^2 + \frac{1}{4}\right]^{1-n}}, \quad (44)$$

$$\alpha = \frac{\left(\frac{\tau_y}{h}\right)^{2n} \left(\frac{\tau_x}{2\tau_y}\right)^2}{\left[\left(\frac{\tau_x}{2\tau_y}\right)^2 + \frac{1}{4}\right]^{1-n}}. \quad (45)$$

Finally, using Eqs. (37) and (38), we get equations for velocities  $u$  and  $v$ :

$$\frac{\partial u}{\partial x} = \sqrt{\alpha} = \frac{\left(\frac{\tau_y}{h}\right)^n \frac{\tau_x}{2\tau_y}}{\left[\left(\frac{\tau_x}{2\tau_y}\right)^2 + \frac{1}{4}\right]^{\frac{1-n}{2}}} = \frac{\tau_x}{2h} \left[\left(\frac{\tau_x}{2h}\right)^2 + \left(\frac{\tau_y}{2h}\right)^2\right]^{\frac{n-1}{2}}, \quad (46)$$

$$\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} = \sqrt{\beta} = \frac{\left(\frac{\tau_y}{h}\right)^n}{\left[\left(\frac{\tau_x}{2\tau_y}\right)^2 + \frac{1}{4}\right]^{\frac{1-n}{2}}} = \frac{\tau_y}{h} \left[ \left(\frac{\tau_x}{2h}\right)^2 + \left(\frac{\tau_y}{2h}\right)^2 \right]^{\frac{n-1}{2}}. \quad (47)$$

Thus, instead of solving non-linear Eq. (24), this algorithm consists of solving two systems of first order linear differential equations: Eqs. (29) and (30) for stresses and Eqs. (46) and (47) for velocities.

### 3.3 Manufactured solution

To generate a manufactured solution, assume a flow with linearly sloping down in  $x$  direction ice surface and symmetric in  $y$  direction bed and velocity functions:

$$\begin{aligned} u(x, y) &= U_0 \cos(w_x x) \sin(w_y y), \\ v(x, y) &= -U_0 \frac{w_x}{w_y} \sin(w_x x) \cos(w_y y), \\ h(x, y) &= k_1 \cos^2(w_x x) \cos^2(w_y y) + k_2, \\ s(x, y) &= -\delta \tan(\alpha) x, \quad b(x, y) = s(x, y) - h(x, y), \end{aligned} \quad (48)$$

where

$$\begin{aligned} w_x &= \frac{\pi}{3}, \quad w_y = \pi, \quad k_1 = 1, \quad k_2 = 1, \quad \delta = \frac{L}{Z} = 200, \quad \alpha = 0.1, \\ U_0 &= 1, \quad L = 200000 \text{ [m]}, \quad Z = 1000 \text{ [m]}, \quad U = 500 \left[ \frac{\text{m}}{\text{a}} \right]. \end{aligned} \quad (49)$$

Functions  $u(x, y)$ ,  $v(x, y)$  and  $h(x, y)$  chosen as (48) to satisfy the incompressibility conditions and the conservation of mass for the steady-state flow:

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

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



$$u \frac{\partial h}{\partial x} + v \frac{\partial h}{\partial y} = 0.$$

Figure 4 presents the surface, bed, and ice thickness of the generated solution, Fig. 5 presents the  $x$ - and  $y$ -component of velocity, and Fig. 6 presents  $\tau_x = 2\mu h \frac{\partial u}{\partial x}$

5 and  $\tau_y = \mu h \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)$  components of the manufactured solution defined by Eq. (48).

Calculation of the right-hand side functions in Morland–MacAyeal equations for the manufactured analytical solutions are given in Appendix A.

### 3.4 Solving two-dimensional ice-shelf equations with linear algorithm

To implement the linear algorithm, we use the finite difference method. On the rectangular domain an uniform grid with steps  $\Delta x$  and  $\Delta y$  is constructed. Velocities  $u$  and  $v$ , ice-thickness  $h$ , ice surface  $s$  and ice bed  $b$  are defined at the nodes of the grid, while ice-stress variables  $\tau^x$  and  $\tau^y$  are defined at the centers of the cells of the grid as shown in Fig. 1 (right).

First we used the 1st order scheme for the system of linear first order differential Eqs. (29) and (30). This scheme was unstable – the roundoff error grew fast. The challenge in developing numerical schemes for solving systems of hyperbolic partial differential equations resides in approximating smooth solution regions with high spatial accuracy while capturing discontinuities as sharply as possible without any oscillations. In the last couple of decades important advances have been made in this area with the introduction of finite-volume high-resolution upwind methods. However, in this paper, we decided to replace solving a system of hyperbolic differential equations with solving the Poisson equation with linear coefficients. Thus, we replace the problem of solving Eqs. (29) and (30) with a problem of solving two Poisson equations for  $\tau_x$  and  $\tau_y$  generated from (29) and (30). To generate an equation for  $\tau_x$ , we differentiate Eq. (29) by  $x$  and Eq. (30) by  $y$ , and subtract the second equation from the first one. To generate an equation for  $\tau_y$ , we differentiate Eq. (29) by  $y$  and Eq. (30) by  $x$ , and add these two equations. The result will be the following two Poisson equations with constant

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



coefficients:

$$\frac{\partial^2 \tau_x}{\partial x^2} + \frac{\partial^2 \tau_x}{\partial y^2} = \frac{\partial}{\partial x} \left( h \frac{\partial s}{\partial x} + f_{bx} \right) - \frac{\partial}{\partial y} \left( h \frac{\partial s}{\partial y} + f_{by} \right), \quad (50)$$

$$\frac{\partial^2 \tau_y}{\partial x^2} + \frac{\partial^2 \tau_y}{\partial y^2} = \frac{\partial}{\partial y} \left( h \frac{\partial s}{\partial x} + f_{bx} \right) + \frac{\partial}{\partial x} \left( h \frac{\partial s}{\partial y} + f_{by} \right). \quad (51)$$

5 Finding an unique solution of Poisson equation requires more than two sets of boundary conditions. Therefore, we assume that the values of  $\tau_x$  and  $\tau_y$  are defined along the whole border of the domain:

$$\tau_x(0, y) = \tau^{lx}(y), \tau_x(1, y) = \tau^{rx}(y), \text{ for } 0 \leq y \leq 1,$$

$$\tau_y(0, y) = \tau^{ly}(y), \tau_y(1, y) = \tau^{ry}(y), \text{ for } 0 \leq y \leq 1,$$

$$10 \tau_x(x, 0) = \tau^{bx}(x), \tau_x(x, 1) = \tau^{tx}(x), \text{ for } 0 \leq x \leq 1,$$

$$\tau_y(x, 0) = \tau^{by}(x), \tau_y(x, 1) = \tau^{ty}(x), \text{ for } 0 \leq x \leq 1.$$

Equations (50) and (51) can be approximated with the second order scheme as follows:

$$15 \nabla^2 \tau_{i,j}^x = \nu_y \delta_x \left[ h \bar{\delta}_x s + f^{bx} \right]_{i,j} - \nu_x \delta_y \left[ h \bar{\delta}_y s + f^{by} \right]_{i,j}, \quad (52)$$

$$\nabla^2 \tau_{i,j}^y = \nu_y \delta_y \left[ h \bar{\delta}_x s + f^{bx} \right]_{i,j} + \nu_x \delta_x \left[ h \bar{\delta}_y s + f^{by} \right]_{i,j}, \quad (53)$$

where, following (Tannehill et al., 1997), we used the central-difference operators  $\nabla^2$ ,  $\delta$ ,  $\bar{\delta}$  and an averaging operator  $\nu$  defined as:

$$20 \nabla^2 f_{i,j} = \frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{(\Delta x)^2} + \frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{(\Delta y)^2},$$

$$\delta_x f_{i,j} = \frac{f_{i+1/2,j} - f_{i-1/2,j}}{\Delta x},$$

$$v_x f_{i,j} = \frac{f_{i+1/2,j} + f_{i-1/2,j}}{2},$$

$$\bar{\delta}_x f_{i,j} = \frac{f_{i+1,j} - f_{i-1,j}}{2\Delta x},$$

$$v_y f_{i,j} = \frac{f_{i,j+1/2} + f_{i,j-1/2}}{2}.$$

Equations (50) and (51) can be approximated with the second order scheme as follows:

$$\frac{\tau_{i+1,j}^x - 2\tau_{i,j}^x + \tau_{i-1,j}^x}{(\Delta x)^2} + \frac{\tau_{i,j+1}^x - 2\tau_{i,j}^x + \tau_{i,j-1}^x}{(\Delta y)^2} = \quad (54)$$

$$\frac{1}{\Delta x} \left[ h_{i+1,j+1/2} \frac{S_{i+2,j+1/2} - S_{i+1,j+1/2}}{2\Delta x} - h_{i,j+1/2} \frac{S_{i+1,j+1/2} - S_{i-1,j+1/2}}{2\Delta x} + f_{i+1,j+1/2}^{bx} - f_{i,j+1/2}^{bx} \right]$$

$$- \frac{1}{\Delta y} \left[ h_{i+1/2,j+1} \frac{S_{i+1/2,j+2} - S_{i+1/2,j}}{2\Delta y} - h_{i,j+1/2} \frac{S_{i+1/2,j+1} - S_{i+1/2,j-1}}{2\Delta y} + f_{i+1/2,j+1}^{by} - f_{i+1/2,j}^{by} \right],$$

$$\frac{\tau_{i+1,j}^y - 2\tau_{i,j}^y + \tau_{i-1,j}^y}{(\Delta x)^2} + \frac{\tau_{i,j+1}^y - 2\tau_{i,j}^y + \tau_{i,j-1}^y}{(\Delta y)^2} = \quad (55)$$

$$\frac{1}{\Delta y} \left[ h_{i+1/2,j+1} \frac{S_{i+1,j+1} - S_{i,j+1}}{\Delta x} - h_{i+1/2,j} \frac{S_{i+1,j} - S_{i,j}}{\Delta x} + f_{i+1/2,j+1}^{bx} - f_{i+1/2,j}^{bx} \right],$$

$$+ \frac{1}{\Delta x} \left[ h_{i+1,j+1/2} \frac{S_{i+1,j+1} - S_{i+1,j}}{\Delta y} - h_{i,j+1/2} \frac{S_{i,j+1} - S_{i,j}}{\Delta y} + f_{i+1,j+1/2}^{by} - f_{i,j+1/2}^{by} \right].$$

Equations (54) and (55) are Poisson equations with sparse block tridiagonal matrix with constant coefficients. There are many methods, such as cyclic reduction, Direct block factorization, Buneman, or fast Fourier transformation method, to optimally solve this linear system. In this work, we use Fast Fourier Transformation (FFT) method (Swarztrauber and Sweet, 1996). FFT method is accurate to roundoff errors, could be used with Dirichlet or Newmann boundary conditions, however, it can be used only on rectangular domains. Description of the FFT method is given in Appendix B.

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



After finding  $\tau_{i,j}^x$  and  $\tau_{i,j}^y$ , velocity  $u$  and  $v$  can be found by solving the system of linear first order differential Eqs. (46) and (47) as:

$$\frac{u_{i+1,j} - u_{i,j}}{\Delta x} = 0.5 \left( F_{i,j}^x + F_{i,j-1}^x \right), \quad (56)$$

$$\frac{v_{i+1,j} - v_{i,j}}{\Delta x} + 0.5 [\bar{\delta}_y u_{i,j} + \bar{\delta}_y u_{i+1,j}] = 0.5 \left( F_{i,j}^y + F_{i,j-1}^y \right), \quad (57)$$

where  $F^x$  and  $F^y$  in Eqs. (56) and (57) are the approximations of the right-hand side functions in Eqs. (46) and (47). In the following calculations, we assumed that Dirichlet boundary conditions are specified at the left and bottom boundaries:

$$u_{1,j} = u_{\text{ex}}, \quad u_{i,1} = u_{\text{ex}}, \quad (58)$$

$$v_{1,j} = v_{\text{ex}}, \quad v_{i,1} = v_{\text{ex}}. \quad (59)$$

### 3.5 Solving two-dimensional ice-shelf equations with traditional iterative method

To compare the new algorithm with the traditional iterative algorithm, we approximated Eq. (24) with traditional five-point finite difference equations of the form  $A(x)x = b$ . To solve these non-linear equations, we used Gauss–Zeidel iterative method.

The Gauss–Zeidel method for solving the system  $A(x)x = b$  can be expressed as

$$Lx^{(k+1)} + Dx^{(k+1)} + Ux^{(k)} = b, \text{ or}$$

$$x^{(k+1)} = (L + D)^{-1} \left( -Ux^{(k)} + b \right),$$

where  $D$ ,  $L$ , and  $U$  represent the diagonal, lower-triangular, and upper-triangular parts of  $A$ , respectively.

As a stopping criteria, we use

$$\|r^{(k)}\|_{\infty} \leq \epsilon \|b\|_{\infty},$$

where  $r^{(k)}$  is the vector of residuals  $r^{(k)} = Ax^{(k)} - b$  and the norm  $\|b\|_{\infty} = \max_{1 \leq i \leq N} |b_i|$ .

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

3.6 Comparing the direct and iterative algorithms for two- dimensional ice-shelf problem

We tested the performance of the direct and iterative algorithms on the manufactured analytical solution on the sets of grids with 100 × 100, 200 × 200, and 400 × 400 nodes.

5 The differences between the approximate solutions and analytical solutions were measured using  $L_2$ -norm:  $\|u\|_{L_2}^2 = \frac{1}{N_x \times N_y} \sum \left(u_{i,j} - u_{i,j}^{\text{exact}}\right)^2$ .

Table 2 shows time it took for the method to solve the problem or to converge to the required accuracy, number of iterations for the iterative method, and the errors in calculation of the components of velocity ( $u$  and  $v$ ) and stress ( $\tau_x$  and  $\tau_y$ ).

10 As the table shows, for all three grids both the direct method and the traditional non-linear iterative method produce about the same accuracy results. The convergence rates, calculated between consecutive pairs of grids as  $\log_4 \left(\frac{e_{i-1}}{e_i}\right)$ , where  $e_{i-1}, e_i$  are the errors on a series of uniformly refined grids, are about 0.60 – 0.68 for both direct and iterative methods. However, the direct method is more than thousand times faster  
15 than the iterative method.

4 Conclusions

In this work, we offer a linear non-iterative algorithm for solving ice-shelf equations. The applicability of the algorithm is demonstrated for one-dimensional and two-dimensional ice shelf equations through comparison of outputs to the manufactured analytical solutions of the equations. The efficiency of the linear algorithm is demonstrated by comparing the linear algorithm with the traditional non-linear iterative algorithm for derived manufactured solutions. The linear algorithm is shown to be as accurate as the traditional iterative algorithm but significantly faster. The assumption of availability of boundary conditions for stresses, the assumption that they are known or can be supplied from  
20 other sources, is a limitation of the method. The method may be valuable as the way  
25

A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

AbstractIntroduction

ConclusionsReferences

TablesFigures

⏮⏭

⏪⏩

BackClose

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



to increase the efficiency of complex ice sheet models a part of which requires solving the ice shelf models as well as to solve efficiently two-dimensional ice-shelf equations.

## Appendix A

### Calculation of the right-hand side functions of 2-D Morland–Mac–Ayeal equations corresponding to the manufactured analytical solution

For calculation of the right-hand side functions in MM equations, we need the derivatives of velocity and ice thickness:

$$\frac{\partial u}{\partial x} = -U_0 w_x \sin(w_x x) \sin(w_y y), \quad \frac{\partial^2 u}{\partial x^2} = -w_x^2 u, \quad (A1)$$

$$\frac{\partial u}{\partial y} = U_0 w_y \cos(w_x x) \cos(w_y y), \quad \frac{\partial^2 u}{\partial y^2} = -w_y^2 u,$$

$$\frac{\partial^2 u}{\partial x \partial y} = -U_0 w_x w_y \sin(w_x x) \cos(w_y y),$$

$$\frac{\partial v}{\partial x} = -U_0 \frac{w_x^2}{w_y} \cos(w_x x) \cos(w_y y), \quad \frac{\partial^2 v}{\partial x^2} = -w_x^2 v,$$

$$\frac{\partial v}{\partial y} = U_0 w_x \sin(w_x x) \sin(w_y y), \quad \frac{\partial^2 v}{\partial y^2} = -w_y^2 v,$$

$$\frac{\partial^2 v}{\partial x \partial y} = U_0 w_x^2 \cos(w_x x) \sin(w_y y),$$

$$\frac{\partial z_s}{\partial x} = -\delta \tan(\alpha), \quad \frac{\partial z_s}{\partial y} = 0,$$

$$\frac{\partial h}{\partial x} = -k_1 w_x \sin(2w_x x) \cos^2(w_y y), \quad \frac{\partial h}{\partial y} = -k_1 w_y \cos^2(w_x x) \sin(2w_y y),$$

### A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



$$\frac{\partial^2 h}{\partial x^2} = -2k_1 w_x^2 \cos(2w_x x) \cos^2(w_y y), \quad \frac{\partial^2 h}{\partial y^2} = -2k_1 w_y^2 \cos^2(w_x x) \cos(2w_y y),$$

$$\frac{\partial^2 h}{\partial x \partial y} = k_1 w_x w_y \sin(2w_x x) \sin(2w_y y).$$

Calculation of the right-hand-side functions requires calculation of the derivatives of the effective viscosity:

$$\frac{\partial \mu}{\partial x} = \mu \frac{1-n}{2n} \frac{\frac{\partial^2 u}{\partial x^2} \left( 2 \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \frac{\partial^2 v}{\partial x \partial y} \left( 2 \frac{\partial v}{\partial y} + \frac{\partial u}{\partial x} \right) + \frac{1}{2} \left( \frac{\partial^2 u}{\partial x \partial y} + \frac{\partial^2 v}{\partial x^2} \right) \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)}{\left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 + \frac{1}{4} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 + \frac{\partial u}{\partial x} \frac{\partial v}{\partial y}},$$

$$\frac{\partial \mu}{\partial y} = \mu \frac{1-n}{2n} \frac{\frac{\partial^2 v}{\partial y^2} \left( 2 \frac{\partial v}{\partial y} + \frac{\partial u}{\partial x} \right) + \frac{\partial^2 u}{\partial x \partial y} \left( 2 \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \frac{1}{2} \left( \frac{\partial^2 v}{\partial x \partial y} + \frac{\partial^2 u}{\partial y^2} \right) \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)}{\left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 + \frac{1}{4} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 + \frac{\partial u}{\partial x} \frac{\partial v}{\partial y}}.$$

Finally, substituting the constructed solutions into Morland–MacAyeal equations generates the following right-hand-side functions:

$$f_{bx} = 2 \left( h \frac{\partial \mu}{\partial x} + \mu \frac{\partial h}{\partial x} \right) \left( 2 \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + 2\mu h \left( 2 \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial x \partial y} \right) + \left( h \frac{\partial \mu}{\partial y} + \mu \frac{\partial h}{\partial y} \right) \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + \mu h \left( \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 v}{\partial x \partial y} \right) - h \frac{\partial s}{\partial x},$$

$$f_{by} = 2 \left( h \frac{\partial \mu}{\partial y} + \mu \frac{\partial h}{\partial y} \right) \left( 2 \frac{\partial v}{\partial y} + \frac{\partial u}{\partial x} \right) + 2\mu h \left( 2 \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 u}{\partial x \partial y} \right) + \left( h \frac{\partial \mu}{\partial x} + \mu \frac{\partial h}{\partial x} \right) \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + \mu h \left( \frac{\partial^2 u}{\partial x \partial y} + \frac{\partial^2 v}{\partial x^2} \right) - h \frac{\partial s}{\partial y}.$$

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

## Appendix B

### Fast Fourier Transformation Method for solving Poisson equation with constant coefficients

#### B1 Two dimensional Poisson equation

5 Let's consider the Dirichlet problem for Poisson equation:

$$\begin{aligned}\Delta U = U_{xx} + U_{yy} &= -f(x, y) \quad \text{in } G = \{(x, y) : x \in [0, l_1], y \in [0, l_2]\}, \\ U(0, y) &= \mu_1(y), & U(l_1, y) &= \mu_2(y), \\ U(x, 0) &= \mu_3(x), & U(x, l_2) &= \mu_4(x).\end{aligned}\tag{B1}$$

10 In a regular grid with steps  $\Delta x$  and  $\Delta y$ , Eq. (B1) can be approximated with a scheme (B2) with zero boundary conditions. The right-hand side functions of the equations have been rearranged to nullify the boundary conditions.

$$\begin{aligned}\frac{V_{i-1,j} - 2V_{i,j} + V_{i+1,j}}{(\Delta x)^2} + \frac{V_{i,j-1} - 2V_{i,j} + V_{i,j+1}}{(\Delta y)^2} &= -F_{i,j}, \quad i = 1, \dots, N_x - 1, \quad j = 1, \dots, N_y - 1, \\ V_{0,j} &= 0, \quad V_{N_x,j} = 0, \quad j = 1, \dots, N_y - 1, \\ V_{i,0} &= 0, \quad V_{i,N_y} = 0, \quad i = 1, \dots, N_x - 1.\end{aligned}\tag{B2}$$

## GMDD

7, 1829–1864, 2014

### A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



## B2 Solving One-Dimensional Poisson Equation with Fast Fourier Transformation

First let's consider the following one-dimensional eigenvalues problem of Poisson's equation:

$$\frac{V(j-1) - 2V(j) + V(j+1)}{(\Delta y)^2} + \lambda V(j) = 0, \quad j = 1, \dots, N_y - 1, \quad (B3)$$

$$V(0) = 0, \quad V(N_y) = 0,$$

where  $V(j) = V(y_j)$ .

This problem has a full set of orthogonal eigenvectors and eigenvalues:

$$V(j) = V_k(j) = \sqrt{\frac{2}{l}} \sin \frac{\pi k y_j}{l}, \quad \lambda_k = \frac{4}{(\Delta y)^2} \sin^2 \frac{\pi k \Delta y}{2l}, \quad k = 1, \dots, N_y - 1, \quad l = N_y \Delta y. \quad (B4)$$

The eigenvectors form a complete orthogonal set with respect to the following scalar product and norm:  $(p, q) = \sum_{j=1}^{N_y-1} p_j q_j \Delta y$ ,  $\text{norm } p = \sqrt{(p, p)} = \left( \sum_{j=1}^{N_y-1} p_j^2 \Delta y \right)^{1/2}$ .

## B3 Fast Furier Transformation

For every value of  $i$ , functions  $V_{i,j}$  and  $F_{i,j}$  can be considered as one-dimensional functions depending only on  $j$ . Then, they can be written as

$$V_{i,j} = \sum_{k=1}^{N_y-1} c_k(i) \mu_k(j), \quad F_{i,j} = \sum_{k=1}^{N_y-1} f_k(i) \mu_k(j), \quad (B5)$$

where Fourier coefficients  $f_k(i)$  are calculated as

$$f_k(i) = (f, \mu_k) = \sum_{j=1}^{N_y-1} F_{i,j} \mu_k(j), \quad i = 1, \dots, N_x - 1, \quad (B6)$$

**GMDD**

7, 1829–1864, 2014

**A linear algorithm for solving ice shelf equations**

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



and  $c_k(i)$  are unknown coefficients we are trying to find.

Substituting Eq. (B5) into Eq. (B2) and using independence of functions  $\mu_k(j)$  and relationship

$$\mu_k(j)_{\bar{y}y,j} = -\lambda_k \mu_k(j),$$

we get the following  $N_y - 1$  set of three-diagonal equations for  $c_k(i)$  for  $k = 1, \dots, N_y - 1$ :

$$\frac{c_k(i-1) - 2c_k(i) + c_k(i+1)}{(\Delta y)^2} - \lambda c_k(i) + f_k(i) = 0, \quad i = 1, \dots, N_x - 1, \quad (B7)$$

$$c_k(0) = 0, \quad c_k(N_x) = 0.$$

This set of equations can be solved with method of solving equations with three-diagonal matrix.

Thus, the fast Fourier transformation algorithm consists of the following steps:

1. Calculate Fourier coefficients  $f_k(i)$  of the right-hand side function  $F_{i,j}$  using formulas (B6). This takes  $O(N_y \log_2 N_y)$  operations. Total calculation of all  $N_x$  sums requires  $O(N_x N_y \log_2 N_y)$  operations.
2. To find coefficients  $c_k(i)$ , solve  $N_y$  systems of equations (B7) with three-diagonal matrices. This requires  $O(N_x N_y)$  operations.
3. After  $c_k(i)$  are found, find solution  $U_{i,j}$  using formulas (B5). This also requires  $O(N_x N_y \log_2 N_y)$  operations.

Thus, the total cost of the method is  $O(N_x N_y \log_2 N_y)$ . This method is a direct method and faster than many other methods. However, the method cannot be used for problems with Robin boundary conditions. The method can be easily parallelized.

**Supplementary material related to this article is available online at**  
<http://www.geosci-model-dev-discuss.net/7/1829/2014/gmdd-7-1829-2014-supplement.zip>.

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



## References

- Bueler, E., Lingle, C. S., Kallen-Brown, J. A., Covey, D. N., and Bowman, L. N.: Exact solutions and verification of numerical models for isothermal ice sheets, *J. Glaciol.*, 51, 291–306, 2005. 1831
- 5 Bueler, E., Brown, J., and Lingle, C.: Exact solutions to the thermomechanically coupled shallow ice approximation: effective tools for verification, *J. Glaciol.*, 53, 182, 499–516, 2007. 1831
- Gagliardini, O. and Zwinger, T.: The ISMIP-HOM benchmark experiments performed using the Finite-Element code Elmer, *The Cryosphere*, 2, 67–76, doi:10.5194/tc-2-67-2008, 2008. 1830
- 10 Larour, E., Seroussi, H., Morlighem, M., and Rignot, E.: Continental scale, high order, high spatial resolution, ice sheet modeling using the Ice Sheet System Model, *J. Geophys. Res.*, 117, F01022, doi:10.1029/2011JF002140, 2012. 1830
- Lipscomb, W., Bindschadler, R., Bueler, E., Holland, D., Johnson, J., and Price, S.: A community ice sheet model for sea level prediction, *Eos Trans.*, 90, 3, doi:10.1029/2009EO030004, 2009. 1830
- 15 MacAyeal, D. R.: Large-scale ice flow over a viscous basal sediment: theory and application to ice stream B, Antarctica, *J. Geophys. Res.*, 94, 4071–4087, 1989. 1830
- MacAyeal, D. R.: *EISMINT: Lessons in Ice-Sheet Modeling*, Department of Geophysical Sciences, University of Chicago, Chicago, IL, 1997. 1832, 1839
- 20 Morland, L. W.: Unconfined ice-shelf flow, in: *Proceedings of Workshop on the Dynamics of the West Antarctic Ice Sheet*, University of Utrecht, May 1985, edited by: van der Veen, C. J. and Oerlemans, J., Reidel, 99–116, 1987. 1830
- Rutt, I. C., M. Hagdorn, N. R. J. Hulton, and A. J. Payne: The Glimmer community ice sheet model, *J. Geophys. Res.*, 114, F02004, doi:10.1029/2008JF001015, 2009. 1830
- 25 Sargent, A. and Fastook, J. L.: Manufactured analytical solutions for isothermal full-Stokes ice sheet models, *The Cryosphere*, 4, 285–311, doi:10.5194/tc-4-285-2010, 2010. 1831
- Swarztrauber, P. N. and Sweet, R. A.: The Fourier and cyclic reduction methods for solving Poisson's Equation, in: *Handbook of Fluid Dynamics and Fluid Machinery*, edited by: Schetz, J. A. and Fuhs, A. E., John Wiley and Sons, New York, NY, 1354–1356, 1996. 1847
- 30 Tannehill, J. C., Anderson, D. A., and Pletcher, R. H.: *Computational Fluid Mechanics and Heat Transfer*, 2nd edn., Series in Computational and Physical Processes in Mechanics and Thermal Sciences, Taylor & Francis, ISBN 1-56032-046-X, Washington, DC, 1997. 1846

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Winkelmann, R., Martin, M. A., Haseloff, M., Albrecht, T., Bueler, E., Khroulev, C., and Levermann, A.: The Potsdam Parallel Ice Sheet Model (PISM-PIK) – Part 1: Model description, The Cryosphere, 5, 715–726, doi:10.5194/tc-5-715-2011, 2011. 1830

## GMDD

7, 1829–1864, 2014

### A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

**Table 1.** 1-dimensional benchmark experiments. Time and  $L^2$  errors in the velocity and stress.

Grid	Method	Time	Iteration	Velo. error ( $u$ )	Conv. rate ( $u$ )	Pres. error ( $\tau$ )	Conv. rate ( $\tau$ )
$N = 100$	Direct	$3.10 \times 10^{-5}$	76	$2.12 \times 10^{-4}$	–	$6.45 \times 10^{-4}$	–
	Iterative	$1.73 \times 10^{-3}$		$6.50 \times 10^{-4}$	–	$6.81 \times 10^{-4}$	–
$N = 1000$	Direct	$2.32 \times 10^{-4}$	68	$7.95 \times 10^{-6}$	1.43	$9.56 \times 10^{-5}$	0.83
	Iterative	$1.09 \times 10^{-2}$		$8.56 \times 10^{-6}$	1.88	$9.56 \times 10^{-5}$	0.85
$N = 10000$	Direct	$1.31 \times 10^{-3}$	58	$8.65 \times 10^{-8}$	1.96	$1.40 \times 10^{-5}$	0.83
	Iterative	$5.57 \times 10^{-2}$		$8.68 \times 10^{-8}$	1.99	$1.40 \times 10^{-5}$	0.83
$N = 100000$	Direct	$9.07 \times 10^{-3}$	50	$8.63 \times 10^{-10}$	2.00	$2.06 \times 10^{-6}$	0.83
	Iterative	$4.70 \times 10^{-1}$		$6.49 \times 10^{-9}$	1.13	$2.06 \times 10^{-6}$	0.83
$N = 1000000$	Direct	$8.40 \times 10^{-2}$	41	$8.58 \times 10^{-12}$	2.00	$3.02 \times 10^{-7}$	0.83
	Iterative	$3.89 \times 10^{+0}$		$4.77 \times 10^{-7}$	–1.87	$3.22 \times 10^{-7}$	0.81
$N = 10000000$	Direct	$8.62 \times 10^{-1}$	37	$2.67 \times 10^{-13}$	1.51	$4.44 \times 10^{-8}$	0.83
	Iterative	$3.53 \times 10^{+1}$		$9.25 \times 10^{-07}$	–0.29	$1.66 \times 10^{-6}$	–0.71

[Title Page](#)
[Abstract](#)
[Introduction](#)
[Conclusions](#)
[References](#)
[Tables](#)
[Figures](#)
[◀](#)
[▶](#)
[◀](#)
[▶](#)
[Back](#)
[Close](#)
[Full Screen / Esc](#)
[Printer-friendly Version](#)
[Interactive Discussion](#)

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

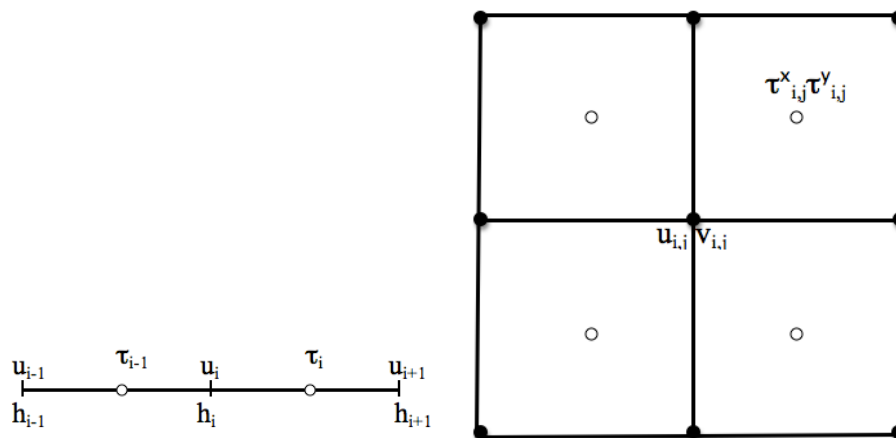
**Table 2.** 2-dimensional benchmark experiments. Time and  $L^2$  errors in the velocity and stress.

Grid	Method	Time [c]	Iteration	Error ( $u$ )	Error ( $v$ )	Error ( $\tau_x$ )	Error ( $\tau_y$ )
100 × 100	Linear	0.02		$9.78 \times 10^{-4}$	$3.48 \times 10^{-3}$	$4.71 \times 10^{-3}$	$1.34 \times 10^{-3}$
	Iterative	53.91	33 937	$3.50 \times 10^{-3}$	$2.37 \times 10^{-3}$	$3.50 \times 10^{-3}$	$2.37 \times 10^{-3}$
200 × 200	Linear	0.12		$3.78 \times 10^{-4}$	$1.49 \times 10^{-3}$	$2.05 \times 10^{-3}$	$5.25 \times 10^{-4}$
	Iterative	860.82	130 125	$1.38 \times 10^{-3}$	$9.31 \times 10^{-4}$	$1.38 \times 10^{-3}$	$9.31 \times 10^{-4}$
400 × 400	Linear	1.56		$1.47 \times 10^{-4}$	$6.29 \times 10^{-4}$	$8.78 \times 10^{-4}$	$2.08 \times 10^{-4}$
	Iterative	18 190.834	498 737	$5.46 \times 10^{-4}$	$3.69 \times 10^{-4}$	$5.46 \times 10^{-4}$	$3.69 \times 10^{-4}$

[Title Page](#)
[Abstract](#)
[Introduction](#)
[Conclusions](#)
[References](#)
[Tables](#)
[Figures](#)
[◀](#)
[▶](#)
[◀](#)
[▶](#)
[Back](#)
[Close](#)
[Full Screen / Esc](#)
[Printer-friendly Version](#)
[Interactive Discussion](#)

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

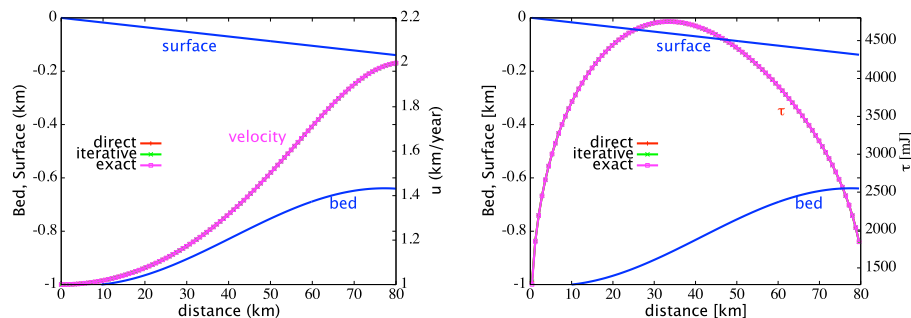


**Fig. 1.** Left: 1-D Staggered grid. Velocity  $u$  and ice-thickness  $h$  are defined at the nodes of the grid and stress variable  $\tau$  is defined at the centroids of the grid. Right: 2-D grid. velocities  $u$  and  $v$ , ice-thickness  $h$ , ice surface  $s$  and bed  $b$  are defined at the nodes of the grid, while ice-stress variables  $\tau^x$  and  $\tau^y$  are defined at the centers of the cells of the grid.

[Title Page](#)
[Abstract](#)
[Introduction](#)
[Conclusions](#)
[References](#)
[Tables](#)
[Figures](#)
[◀](#)
[▶](#)
[◀](#)
[▶](#)
[Back](#)
[Close](#)
[Full Screen / Esc](#)
[Printer-friendly Version](#)
[Interactive Discussion](#)

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook



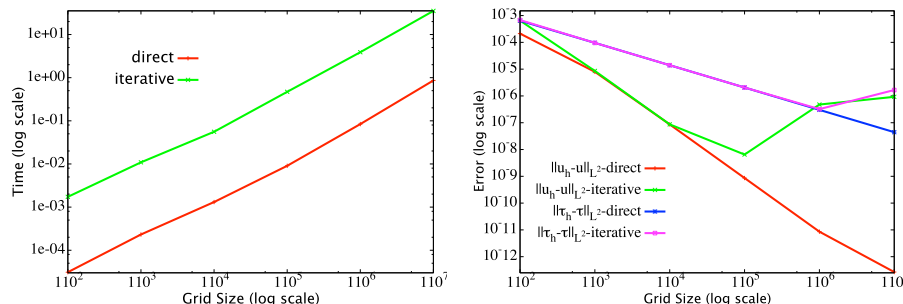
**Fig. 2.** The ice surface and bed and analytical solutions and solutions calculated by iterative and direct methods. Iterative and direct methods' solutions coincide, therefore only one is visible. Left: the velocity in  $\text{km a}^{-1}$ . Right: the stress in  $\text{mJ}$ . Grid with 100 nodes.

[Title Page](#)
[Abstract](#)
[Introduction](#)
[Conclusions](#)
[References](#)
[Tables](#)
[Figures](#)
[◀](#)
[▶](#)
[◀](#)
[▶](#)
[Back](#)
[Close](#)
[Full Screen / Esc](#)
[Printer-friendly Version](#)
[Interactive Discussion](#)



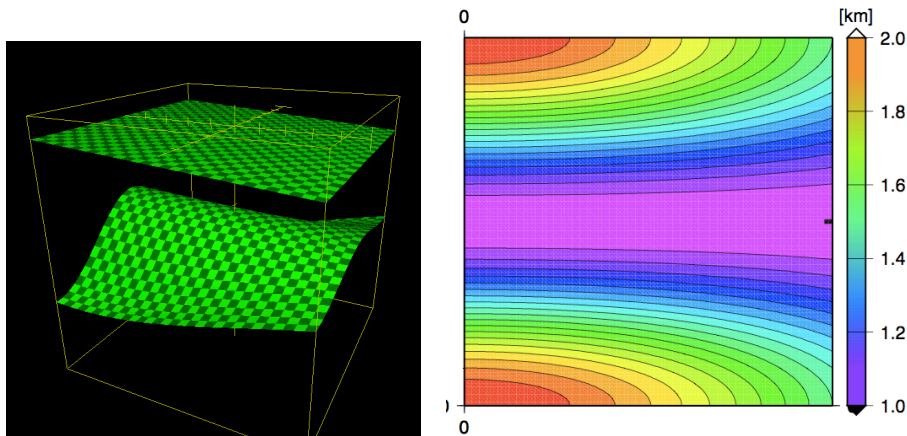
## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook



**Fig. 3.** The errors of the iterative and direct methods ( $L^2$  norm) and the time it took for the methods to solve the problem of sizes  $10^2$ ,  $10^3$ ,  $10^4$ ,  $10^5$ ,  $10^6$ , and  $10^7$ . Time, errors, and problem size are shown in log scale. The stresses and velocities are calculated with the same accuracy by both methods up to problem size of  $10^6$  and  $10^4$  respectively. The further increase in the problem size decreases the accuracy of the iterative method. The linear method is about hundred (100) times faster than the iterative method for every grid size.

[Title Page](#)
[Abstract](#)
[Introduction](#)
[Conclusions](#)
[References](#)
[Tables](#)
[Figures](#)
[◀](#)
[▶](#)
[◀](#)
[▶](#)
[Back](#)
[Close](#)
[Full Screen / Esc](#)
[Printer-friendly Version](#)
[Interactive Discussion](#)



**Fig. 4.** Two-dimensional manufactured solution. Left: ice surface and bed, Right: ice thickness.

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

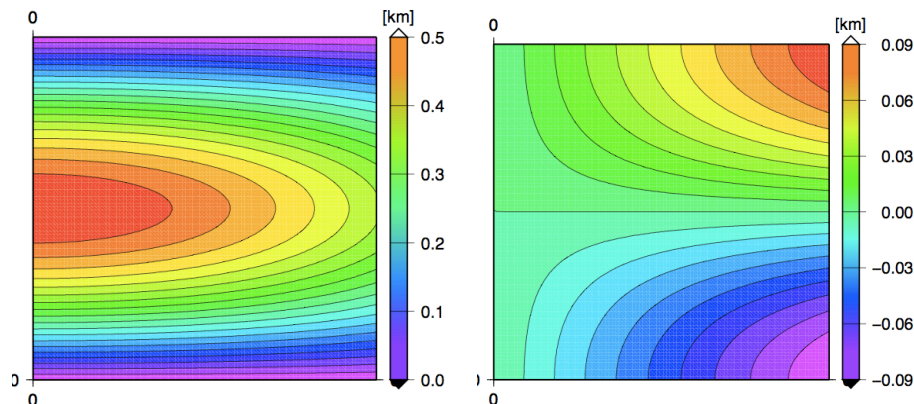
Full Screen / Esc

Printer-friendly Version

Interactive Discussion

## A linear algorithm for solving ice shelf equations

A. Sargent and  
J. L. Fastook



**Fig. 5.** Two-dimensional manufactured solution: velocity, Left:  $x$ -component, Right:  $y$ -component.

[Title Page](#)[Abstract](#)[Introduction](#)[Conclusions](#)[References](#)[Tables](#)[Figures](#)[◀](#)[▶](#)[◀](#)[▶](#)[Back](#)[Close](#)[Full Screen / Esc](#)[Printer-friendly Version](#)[Interactive Discussion](#)

