Instruments and Methods

A tutorial on adjoint methods and their use for data assimilation in glaciology

Glen D. GRANZOW

Department of Computer Science, University of Montana, Missoula, MT, USA
E-mail: ggranzow@gmail.com

ABSTRACT. This paper provides an introduction to adjoint methods, which are used to find the gradient of an objective function, as required by optimization algorithms. Examples are included, culminating in a data-assimilation problem from glaciology.

KEYWORDS: glacier modelling, glaciological instruments and methods, glaciological model experiments, ice-sheet modelling

PREFACE

Numerical models of glaciers and ice sheets include parameters such as viscosity (often expressed using other parameters), density, thermal conductivity, heat capacity, geothermal heat flux and basal traction coefficients. (Even basal topography can be considered a model 'parameter'.) Many of these parameters are difficult to measure directly. We use the phrase 'data assimilation' to refer to a method where more easily measured data, such as surface velocities, are used to estimate the values of such parameters. The idea behind data assimilation is to find the values of the parameter of interest (e.g. a frictional coefficient along the bed) that result in the best match between values calculated by the model (surface velocities, perhaps) and measured data. Thus, finding appropriate values for model parameters can be accomplished by solving an optimization problem, namely that of minimizing a measure of the difference between calculated and measured values.

In this paper 'adjoint methods' are described in the context of optimization problems. Application to an idealized data-assimilation problem from glaciology is described in the final section. Although applications using real glaciological data can be found in various published research papers (e.g. Brinkerhoff and others, 2011) no attempt is made to review them here.

INTRODUCTION

Suppose we have a problem involving a collection of parameters $\mathbf{p}$ whose solution is $\mathbf{u} = \mathbf{F}(\mathbf{p})$. We want to find the values of the parameters $\mathbf{p}$ that minimize (or maximize) a given (scalar) function $g(\mathbf{u})$. Since $\mathbf{u}$ is a function of the parameters $\mathbf{p}$, we can think of g as a function of $\mathbf{p}$. (Formally we could introduce a new function, $\tilde{g}(\mathbf{p}) = g(\mathbf{F}(\mathbf{p}))$, but we believe the presentation is clearer, albeit less rigorous, without it.) Most efficient algorithms used to optimize g require knowledge of $\partial g/\partial p_k$ for each of the parameters p_k (components of $\mathbf{p}$). Adjoint methods can be used to find these

---

*The adjoint of a matrix Z whose elements are complex numbers is the transpose of the matrix whose elements are the complex conjugates of the elements of Z. For real-valued matrices, such as the Jacobian matrices in this paper, the transpose and the adjoint are the same.

derivatives. We write this collection of partial derivatives as a column matrix $\partial g/\partial \mathbf{p} = [\partial g/\partial p_1 \; \partial g/\partial p_2 \; \cdots \; \partial g/\partial p_m]^{\mathsf{T}}$.

The Jacobian matrix of the function $\mathbf{F}$ that maps the parameters $\mathbf{p}$ to the solution $\mathbf{u}$ is defined as

$$\frac{\partial \mathbf{u}}{\partial \mathbf{p}} = \begin{bmatrix} \dfrac{\partial u_1}{\partial p_1} & \dfrac{\partial u_1}{\partial p_2} & \cdots & \dfrac{\partial u_1}{\partial p_m} \\ \dfrac{\partial u_2}{\partial p_1} & \dfrac{\partial u_2}{\partial p_2} & \cdots & \dfrac{\partial u_2}{\partial p_m} \\ \vdots & & \ddots & \vdots \\ \dfrac{\partial u_n}{\partial p_1} & \dfrac{\partial u_n}{\partial p_2} & \cdots & \dfrac{\partial u_n}{\partial p_m} \end{bmatrix}.$$

This Jacobian matrix can be used to approximate changes in the solution $\mathbf{u}$ resulting from small changes in the parameters $\mathbf{p}$. For an individual component of the solution $\mathbf{u}$,

$$\Delta u_i \approx \frac{\partial u_i}{\partial p_1} \Delta p_1 + \frac{\partial u_i}{\partial p_2} \Delta p_2 + \cdots + \frac{\partial u_i}{\partial p_m} \Delta p_m.$$

The approximate changes for all components of $\mathbf{u}$ can be written compactly (using matrix multiplication) as

$$\Delta \mathbf{u} \approx \frac{\partial \mathbf{u}}{\partial \mathbf{p}} \Delta \mathbf{p}.$$

To find the desired derivatives, $\partial g/\partial \mathbf{p}$, consider

$$g(\mathbf{u}) = g(u_1, u_2, \ldots, u_n)$$
$$= g(u_1(p_1, \ldots, p_m), u_2(p_1, \ldots, p_m), \ldots, u_n(p_1, \ldots, p_m)).$$

Using the chain rule,

$$\frac{\partial g}{\partial p_k} = \frac{\partial g}{\partial u_1} \frac{\partial u_1}{\partial p_k} + \frac{\partial g}{\partial u_2} \frac{\partial u_2}{\partial p_k} + \cdots + \frac{\partial g}{\partial u_n} \frac{\partial u_n}{\partial p_k}.$$

Note that the derivatives of the components of $\mathbf{u}$ with respect to p_k are the elements of the kth *column* of the Jacobian matrix, $\partial \mathbf{u}/\partial \mathbf{p}$. We can write the desired derivatives compactly using the transpose (also called the *adjoint**) of the Jacobian matrix

$$\frac{\partial g}{\partial \mathbf{p}} = \frac{\partial \mathbf{u}}{\partial \mathbf{p}}^{\mathsf{T}} \frac{\partial g}{\partial \mathbf{u}}.$$

Use of the adjoint of a Jacobian matrix seems to be where 'adjoint methods' get their name.

The method can be illustrated by a simple example where the function $\mathbf{F}$ mapping the parameters $\mathbf{p}$ to the solution $\mathbf{u}$ is given explicitly.

Example 1:

Suppose $\mathbf{u} = \begin{bmatrix} p_1^2 + p_2 \\ p_1 p_2 \end{bmatrix}$ and $g(\mathbf{u}) = u_1 + u_2^2$. We want to find $\dfrac{\partial g}{\partial \mathbf{p}} = \begin{bmatrix} \frac{\partial g}{\partial p_1} \\ \frac{\partial g}{\partial p_2} \end{bmatrix}$.

To calculate $\partial g / \partial \mathbf{p}$ using the adjoint method we begin by finding the Jacobian matrix

$$\frac{\partial \mathbf{u}}{\partial \mathbf{p}} = \begin{bmatrix} 2p_1 & 1 \\ p_2 & p_1 \end{bmatrix} \quad \text{and} \quad \frac{\partial g}{\partial \mathbf{u}} = \begin{bmatrix} 1 \\ 2u_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2p_1 p_2 \end{bmatrix}.$$

Then

$$\frac{\partial g}{\partial \mathbf{p}} = \frac{\partial \mathbf{u}}{\partial \mathbf{p}}^{\mathsf{T}} \frac{\partial g}{\partial \mathbf{u}} = \begin{bmatrix} 2p_1 & p_2 \\ 1 & p_1 \end{bmatrix} \begin{bmatrix} 1 \\ 2p_1 p_2 \end{bmatrix} = \begin{bmatrix} 2p_1 + 2p_1 p_2^2 \\ 1 + 2p_1^2 p_2 \end{bmatrix}.$$

To check this solution we find $\partial g / \partial \mathbf{p}$ directly using $g(\mathbf{u}) = u_1 + u_2^2 = (p_1^2 + p_2) + (p_1 p_2)^2$, from which we find $\partial g / \partial p_1 = 2p_1 + 2p_1 p_2^2$ and $\partial g / \partial p_2 = 1 + 2p_1^2 p_2$.

A TYPICAL SITUATION

Often the problem to be solved (whose solution is $\mathbf{u} = \mathbf{F}(\mathbf{p})$) is expressed as a system of n equations, $\mathbf{f}(\mathbf{u}, \mathbf{p}) = \mathbf{0}$. For this problem we still have a function $g(\mathbf{u})$, for which we want to find optimal values of the parameters $\mathbf{p}$ and thus need $\partial g / \partial \mathbf{p} = [\partial g / \partial p_1 \ \partial g / \partial p_2 \ \cdots \ \partial g / \partial p_m]^{\mathsf{T}}$. In the preceding section the chain rule was used to show that $\partial g / \partial \mathbf{p} = \partial \mathbf{u} / \partial \mathbf{p}^{\mathsf{T}} \partial g / \partial \mathbf{u}$. To simplify the appearance of some of the equations that follow, we begin by taking the transpose of both sides of this equation and use the matrix identity $(AB)^{\mathsf{T}} = B^{\mathsf{T}} A^{\mathsf{T}}$ to obtain

$$\frac{\partial g}{\partial \mathbf{p}}^{\mathsf{T}} = \frac{\partial g}{\partial \mathbf{u}}^{\mathsf{T}} \frac{\partial \mathbf{u}}{\partial \mathbf{p}}.$$

As before, the elements of $\partial g / \partial \mathbf{u}$ are obtained by differentiating $g(\mathbf{u})$, but now the elements of $\partial \mathbf{u} / \partial \mathbf{p}$ must be determined from $\mathbf{f}(\mathbf{u}, \mathbf{p}) = \mathbf{0}$. If we take the total derivative of both sides of the equation $f_i(\mathbf{u}, \mathbf{p}) = 0$ with respect to p_k we obtain

$$\frac{\mathrm{d}f_i}{\mathrm{d}p_k} = \frac{\partial f_i}{\partial u_1} \frac{\partial u_1}{\partial p_k} + \frac{\partial f_i}{\partial u_2} \frac{\partial u_2}{\partial p_k} + \cdots + \frac{\partial f_i}{\partial u_n} \frac{\partial u_n}{\partial p_k} + \frac{\partial f_i}{\partial p_k} = 0.$$

Doing this for every combination of the n elements of $\mathbf{u}$ and the m elements of $\mathbf{p}$ results in nm equations that can be represented compactly as $\partial \mathbf{f} / \partial \mathbf{u} \, \partial \mathbf{u} / \partial \mathbf{p} + \partial \mathbf{f} / \partial \mathbf{p} = 0$:

$$\begin{bmatrix} \partial f_1 / \partial u_1 & \partial f_1 / \partial u_2 & \cdots & \partial f_1 / \partial u_n \\ \partial f_2 / \partial u_1 & \ddots & & \\ \vdots & & \ddots & \vdots \\ \partial f_n / \partial u_1 & \partial f_n / \partial u_2 & \cdots & \partial f_n / \partial u_n \end{bmatrix}$$

$$\times \begin{bmatrix} \partial u_1 / \partial p_1 & \partial u_1 / \partial p_2 & \cdots & \partial u_1 / \partial p_m \\ \partial u_2 / \partial p_1 & \ddots & & \\ \vdots & & \ddots & \vdots \\ \partial u_n / \partial p_1 & \partial u_n / \partial p_2 & \cdots & \partial u_n / \partial p_m \end{bmatrix}$$

$$+ \begin{bmatrix} \partial f_1 / \partial p_1 & \partial f_1 / \partial p_2 & \cdots & \partial f_1 / \partial p_m \\ \partial f_2 / \partial p_1 & \ddots & & \\ \vdots & & \ddots & \vdots \\ \partial f_n / \partial p_1 & \partial f_n / \partial p_2 & \cdots & \partial f_n / \partial p_m \end{bmatrix} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & \ddots & & \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$

This equation can be solved for the $\partial u_i / \partial p_k$ factors

$$\frac{\partial \mathbf{u}}{\partial \mathbf{p}} = -\frac{\partial \mathbf{f}}{\partial \mathbf{u}}^{-1} \frac{\partial \mathbf{f}}{\partial \mathbf{p}}$$

needed to obtain our goal

$$\frac{\partial g}{\partial \mathbf{p}}^{\mathsf{T}} = \frac{\partial g}{\partial \mathbf{u}}^{\mathsf{T}} \frac{\partial \mathbf{u}}{\partial \mathbf{p}} = \frac{\partial g}{\partial \mathbf{u}}^{\mathsf{T}} \left(-\frac{\partial \mathbf{f}}{\partial \mathbf{u}}^{-1} \frac{\partial \mathbf{f}}{\partial \mathbf{p}} \right).$$

Here $\partial g / \partial \mathbf{u}^{\mathsf{T}}$ is a $1 \times n$ (row) matrix, $\partial \mathbf{f} / \partial \mathbf{u}$ is $n \times n$ and $\partial \mathbf{f} / \partial \mathbf{p}$ is $n \times m$. To avoid multiplying the two large matrices we perform the multiplication in the order

$$\frac{\partial g}{\partial \mathbf{p}}^{\mathsf{T}} = -\left(\frac{\partial g}{\partial \mathbf{u}}^{\mathsf{T}} \frac{\partial \mathbf{f}}{\partial \mathbf{u}}^{-1} \right) \frac{\partial \mathbf{f}}{\partial \mathbf{p}}$$

with the product $\lambda^{\mathsf{T}} = \partial g / \partial \mathbf{u}^{\mathsf{T}} \partial \mathbf{f} / \partial \mathbf{u}^{-1}$ found by solving the *adjoint problem*

$$\frac{\partial \mathbf{f}}{\partial \mathbf{u}}^{\mathsf{T}} \lambda = \frac{\partial g}{\partial \mathbf{u}}.$$

Note that $\partial \mathbf{f} / \partial \mathbf{u}$ is a Jacobian matrix; once again obtaining the desired partial derivatives ($\partial g / \partial \mathbf{p}$) uses the transpose (i.e. *adjoint*) of a Jacobian matrix. Also note that this Jacobian matrix is the one used in Newton's method to (approximately) solve $\mathbf{f}(\mathbf{u}, \mathbf{p}) = \mathbf{0}$ (with fixed $\mathbf{p}$) iteratively.

Example 2:

Suppose
$$f_1(u_1, u_2, p_1, p_2) = u_1 + u_2 + p_1$$
$$f_2(u_1, u_2, p_1, p_2) = u_1^3 - u_2 + p_2$$
$$g(u_1, u_2) = u_1^2 + u_2^2$$

The system of equations $f_1(u_1, u_2, p_1, p_2) = 0$ and $f_2(u_1, u_2, p_1, p_2) = 0$ has a solution, (u_1, u_2), that depends on the parameters p_1 and p_2. Changing the values of the parameters p_1 and p_2 causes the values of the variables u_1 and u_2 to change and thus the value of $g(u_1, u_2)$ also changes. We want to find $\partial g / \partial \mathbf{p}^{\mathsf{T}} = [\partial g / \partial p_1 \ \partial g / \partial p_2]$.

To calculate $\partial g / \partial \mathbf{p}^{\mathsf{T}}$ using the adjoint method we begin by finding the Jacobian matrix

$$\frac{\partial \mathbf{f}}{\partial \mathbf{u}} = \begin{bmatrix} 1 & 1 \\ 3u_1^2 & -1 \end{bmatrix}$$

along with $\dfrac{\partial \mathbf{f}}{\partial \mathbf{p}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $\dfrac{\partial g}{\partial \mathbf{u}} = \begin{bmatrix} 2u_1 \\ 2u_2 \end{bmatrix}$.

Next we write the adjoint problem $(\partial \mathbf{f} / \partial \mathbf{u})^{\mathsf{T}} \lambda = \partial g / \partial \mathbf{u}$

$$\begin{bmatrix} 1 & 3u_1^2 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 2u_1 \\ 2u_2 \end{bmatrix}$$

and solve it to find

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 2u_2 + \frac{2u_1 - 2u_2}{3u_1^2 + 1} \\ \frac{2u_1 - 2u_2}{3u_1^2 + 1} \end{bmatrix}.$$

The desired derivatives are now found from

$$\frac{\partial g}{\partial \mathbf{p}}^{\mathsf{T}} = -\left(\frac{\partial g}{\partial \mathbf{u}}^{\mathsf{T}} \frac{\partial \mathbf{f}}{\partial \mathbf{u}}^{-1} \right) \frac{\partial \mathbf{f}}{\partial \mathbf{p}} = -\lambda^{\mathsf{T}} \frac{\partial \mathbf{f}}{\partial \mathbf{p}}$$

$$= -\begin{bmatrix} 2u_2 + \dfrac{2u_1 - 2u_2}{3u_1^2 + 1} & \dfrac{2u_1 - 2u_2}{3u_1^2 + 1} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} -2u_2 - \dfrac{2u_1 - 2u_2}{3u_1^2 + 1} & -\dfrac{2u_1 - 2u_2}{3u_1^2 + 1} \end{bmatrix}.$$

To appreciate the meaning of these derivatives we consider a couple of specific value pairs for the parameters p_1 and p_2.
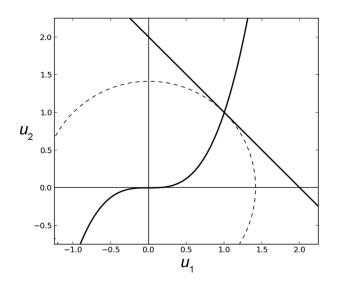
Fig. 1. Curves for the problem presented in example 2, with $p_1 = -2$, $p_2 = 0$ and $g(\mathbf{u}) = 2$.

First, if $p_1 = 0$ and $p_2 = 0$, the equation $\mathbf{f}(\mathbf{u}, \mathbf{p}) = \mathbf{0}$ has solution $\mathbf{u} = \mathbf{0}$, so $g(\mathbf{u}) = 0$. Since $g = u_1^2 + u_2^2 \geq 0$ (always), $g(\mathbf{u}) = 0$ must be a global minimum so we expect $\partial g / \partial \mathbf{p} = \mathbf{0}$. Indeed, substituting $u_1 = 0$ and $u_2 = 0$ into our expression, $\partial g / \partial \mathbf{p}^{\mathsf{T}} = \left[-2 u_2 - \dfrac{2 u_1 - 2 u_2}{3 u_1^2 + 1} \quad -\dfrac{2 u_1 - 2 u_2}{3 u_1^2 + 1} \right]$, confirms this.

To gain further appreciation of $\partial g / \partial \mathbf{p}$, we consider a graphical representation of our example problem. If we plot solutions of $f_1(u_1, u_2, p_1, p_2) = u_1 + u_2 + p_1 = 0$ in the (u_1, u_2) plane we obtain a line with slope -1 and vertical intercept $-p_1$. If we plot solutions of $f_2(u_1, u_2, p_1, p_2) = u_1^3 - u_2 + p_2 = 0$ in the (u_1, u_2) plane we obtain a cubic curve that crosses the vertical axis at p_2 with slope 0. The solution to the system of equations $\mathbf{f}(\mathbf{u}, \mathbf{p}) = \mathbf{0}$ is represented by the point where the line and the cubic curve cross. The function g gives the square of the distance from the origin, so circles centered on the origin represent contours of constant g. Curves for $p_1 = -2$, $p_2 = 0$ and $g = 2$ are shown in Figure 1.
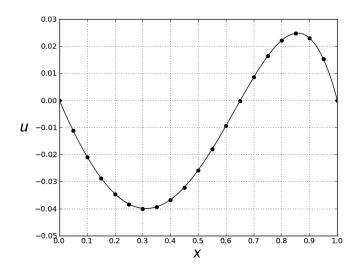

Fig. 2. Solution to the boundary-value problem $u'' - 2u' + u = 1 + x - 5x^2$ for $0 < x < 1$, and $u(0) = 0$, $u(1) = 0$. The solid curve is the exact solution; the black circles are an approximate solution found using finite differences with $n = 20$.

As can be seen from Figure 1, for this value of $\mathbf{p}$ the solution of $\mathbf{f}(\mathbf{u}, \mathbf{p}) = \mathbf{0}$ is $\mathbf{u} = (1, 1)$. If we increase p_1, the line with slope 1 will move down in the figure so the solution point will follow the cubic curve, decreasing its distance from the origin, so g decreases. If, however, we increase p_2, the cubic curve will move up so the solution point will follow the line with slope -1 which is tangent to the (dashed) circle representing a contour of constant g. Thus we expect $\partial g / \partial p_1 < 0$ and $\partial g / \partial p_2 = 0$. Substituting the solution $u_1 = 1$ and $u_2 = 1$ into our expression $\partial g / \partial \mathbf{p}^{\mathsf{T}} = \left[-2 u_2 - \dfrac{2 u_1 - 2 u_2}{3 u_1^2 + 1} \quad -\dfrac{2 u_1 - 2 u_2}{3 u_1^2 + 1} \right]$ gives $\partial g / \partial \mathbf{p}^{\mathsf{T}} = [-2 \quad 0]$, confirming this. The reader is encouraged to use the graphical representation to find other parameter pairs where one of the components of $\partial g / \partial \mathbf{p}$ is zero and use the expression for $\partial g / \partial \mathbf{p}^{\mathsf{T}}$ to confirm their findings.

A NUMERICAL EXAMPLE

In this section we use finite differences to approximate the boundary-value problem

$$c_2 u'' + c_1 u' + c_0 u = p(x) \quad \text{for} \quad 0 < x < 1,$$
$$u(0) = a_0,$$
$$u(1) = a_1,$$

where $p(x) = p_0 + p_1 x + p_2 x^2$ (a second-order polynomial). After computing an approximate solution for particular values of the parameters c_2, c_1, c_0, p_0, p_1, p_2, a_0 and a_1, we use the adjoint method to calculate the rate of change of a function of the solution, $g(u)$, as we vary these parameters. Two different functions are considered: (1) g is the value of u at the midpoint of the domain, $g(u) = u(\frac{1}{2})$, and (2) g is the average value of u, $g(u) = \int_0^1 u(x) \, dx$.

To apply the finite-difference method to our differential equation we divide the domain $[0, 1]$ into n subintervals, each with length $\Delta x = \frac{1}{n}$. The endpoints of these subintervals are x_0, x_1, $\ldots$, x_n, where $x_k = k \Delta x$. We denote approximate values of u at each of these points as u_k. The first derivative, u', at x_k is approximated by $u_k' = \dfrac{u_{k+1} - u_{k-1}}{2 \Delta x}$. The second derivative, u'', at x_k is approximated by $u_k'' = \dfrac{u_{k-1} - 2 u_k + u_{k+1}}{\Delta x^2}$. Substituting these approximations into the differential equation gives

$$c_2 \left(\frac{u_{k-1} - 2 u_k + u_{k+1}}{\Delta x^2} \right) + c_1 \left(\frac{u_{k+1} - u_{k-1}}{2 \Delta x} \right) + c_0 u_k = p(x_k),$$

which can be rearranged to give

$$\left(\frac{c_2}{\Delta x^2} - \frac{c_1}{2 \Delta x} \right) u_{k-1} + \left(-\frac{2 c_2}{\Delta x^2} + c_0 \right) u_k$$
$$+ \left(\frac{c_2}{\Delta x^2} + \frac{c_1}{2 \Delta x} \right) u_{k+1} = p(x_k),$$

a linear algebraic equation that can be written for each of the interior points $k = 1, 2, \ldots, n - 1$. At the endpoints of the domain we assert $u_0 = a_0$ and $u_n = a_1$. This collection of equations can be written compactly as $\mathbf{Au} = \mathbf{b}$, where $\mathbf{A}$ is a tridiagonal matrix, and $\mathbf{u}$ and $\mathbf{b}$ are column matrices. The matrix equation can be solved (e.g. using Gaussian elimination and back substitution) for $\mathbf{u}$, giving an approximation to the solution of the boundary-value problem.

Following the procedure in the preceding paragraph for the (somewhat arbitrary) parameter values $c_2 = 1$, $c_1 = -2$,

$$\frac{\partial \mathbf{f}}{\partial \mathbf{u}} = \mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ \left(\frac{c_2}{\Delta x^2} - \frac{c_1}{2\Delta x}\right) & \left(-\frac{2c_2}{\Delta x^2} + c_0\right) & \left(\frac{c_2}{\Delta x^2} + \frac{c_1}{2\Delta x}\right) & 0 & 0 & \cdots & 0 \\ 0 & \left(\frac{c_2}{\Delta x^2} - \frac{c_1}{2\Delta x}\right) & \left(-\frac{2c_2}{\Delta x^2} + c_0\right) & \left(\frac{c_2}{\Delta x^2} + \frac{c_1}{2\Delta x}\right) & 0 & \cdots & 0 \\ \vdots & & & \ddots & & & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}$$

$$\frac{\partial \mathbf{f}}{\partial \mathbf{p}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ \frac{u_0 - 2u_1 + u_2}{\Delta x^2} & \frac{u_2 - u_0}{2\Delta x} & u_1 & -1 & -x_1 & -x_1^2 & 0 & 0 \\ \frac{u_1 - 2u_2 + u_3}{\Delta x^2} & \frac{u_3 - u_1}{2\Delta x} & u_2 & -1 & -x_2 & -x_2^2 & 0 & 0 \\ & & & \vdots & & & & \\ \frac{u_{n-2} - 2u_{n-1} + u_n}{\Delta x^2} & \frac{u_n - u_{n-2}}{2\Delta x} & u_{n-1} & -1 & -x_{n-1} & -x_{n-1}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$$

Fig. 3. Matrices used to calculate $\partial g/\partial \mathbf{p}$ for the numerical example involving a boundary-value problem.

$c_0 = 1$, $p_0 = 1$, $p_1 = 1$, $p_2 = -5$, $a_0 = 0$ and $a_1 = 0$ generates the approximation shown in Figure 2.

We can use the adjoint method to find the rate of change of a function of $\mathbf{u}$ with respect to the parameters $c_2, c_1, c_0, p_0, p_1, p_2, a_0$ and a_1. We consider first the simple function $g(\mathbf{u}) = u_{n/2} \approx u\left(\frac{1}{2}\right)$, where an even value for n has been chosen.

To utilize the procedure described in the preceding section, the system of equations represented by the matrix equation $\mathbf{Au} = \mathbf{b}$ is written in the form $\mathbf{f}(\mathbf{u}, \mathbf{p}) = \mathbf{0}$, by subtracting the right-hand sides from both sides of each equation. Here $\mathbf{p} = [c_2 \; c_1 \; c_0 \; p_0 \; p_1 \; p_2 \; a_0 \; a_1]$. To find $\partial g/\partial \mathbf{p}$ we need the three matrices $\partial \mathbf{f}/\partial \mathbf{u}$, $\partial \mathbf{f}/\partial \mathbf{p}$ and $\partial g/\partial \mathbf{u}$. Since $\partial \mathbf{f}/\partial \mathbf{u}$ and $\partial \mathbf{f}/\partial \mathbf{p}$ require two columns to be displayed clearly, they are shown in Figure 3. For $g(\mathbf{u}) = u_{n/2}$,

$$\partial g/\partial \mathbf{u} = [0 \; 0 \; 0 \; \cdots \; 0 \; 0 \; 1 \; 0 \; 0 \; \cdots \; 0 \; 0 \; 0]^\mathsf{T}.$$

After solving the adjoint problem $\partial \mathbf{f}/\partial \mathbf{u}^\mathsf{T} \lambda = \partial g/\partial \mathbf{u}$ for λ we can calculate $\partial g/\partial \mathbf{p}^\mathsf{T} = -\lambda^\mathsf{T} \partial \mathbf{f}/\partial \mathbf{p}$. The results for the example problem whose solution is shown in Figure 2 are given in Table 1.

The values of the partial derivatives found using the adjoint method were verified by perturbing the parameters one at a time. For each parameter ($c_2, c_1, c_0, p_0, p_1, p_2, a_0$ and a_1) an additional finite-difference approximation was calculated with that parameter increased by 0.01, giving new values of $\mathbf{u}$. Calling these new values $\tilde{\mathbf{u}}$ we can approximate the partial derivative of g with respect to the perturbed parameter by $\frac{g(\tilde{\mathbf{u}}) - g(\mathbf{u})}{0.01}$. As seen in Table 1, these approximations are very close to the values found using the adjoint method.

To calculate the partial derivatives of $g(u) = \int_0^1 u(x)\,dx$ we use (the composite) Simpson's rule (which requires an even value for n) to approximate this integral,

$$g(\mathbf{u}) = \frac{\Delta x}{3}\left[u_0 + 4u_1 + 2u_2 + 4u_3 + 2u_4 + \cdots + 4u_{n-3}\right.$$
$$\left. + 2u_{n-2} + 4u_{n-1} + u_n\right],$$

from which we have

$$\frac{\partial g}{\partial \mathbf{u}} = \frac{\Delta x}{3}[1 \; 4 \; 2 \; 4 \; 2 \; \cdots \; 4 \; 2 \; 4 \; 1]^\mathsf{T}.$$

Again, $\partial g/\partial \mathbf{p}^\mathsf{T} = -\lambda^\mathsf{T} \partial \mathbf{f}/\partial \mathbf{p}$, where λ is the solution of the adjoint problem $\partial \mathbf{f}/\partial \mathbf{u}^\mathsf{T} \lambda = \partial g/\partial \mathbf{u}$. Results are given in Table 1.

Table 1. Partial derivatives for $g(u) = u\left(\frac{1}{2}\right)$ and $g(u) = \int_0^1 u(x)\,dx$ using a finite-difference approximation to the boundary-value problem $c_2 u'' + c_1 u' + c_0 u = p_0 + p_1 x + p_2 x^2$ for $0 < x < 1$, and $u(0) = a_0$, $u(1) = a_1$ with $c_2 = 1$, $c_1 = -2$, $c_0 = 1$, $p_0 = 1$, $p_1 = 1$, $p_2 = -5$, $a_0 = 0$ and $a_1 = 0$

| | $g(u) = u\left(\frac{1}{2}\right)$ | | $g(u) = \int_0^1 u(x)\,dx$ | |
Partial derivative	Value from adjoint method	Numerical verification	Value from adjoint method	Numerical verification
$\partial g/\partial c_2$	0.04372056	0.04315389	0.02133546	0.02104484
$\partial g/\partial c_1$	0.00762168	0.00763637	0.00424091	0.00424846
$\partial g/\partial c_0$	−0.00262876	−0.00263144	−0.00157897	−0.00158068
$\partial g/\partial p_0$	−0.12775518	−0.12775518	−0.08625040	−0.08625040
$\partial g/\partial p_1$	−0.05862544	−0.05862544	−0.04027710	−0.04027710
$\partial g/\partial p_2$	−0.03210644	−0.03210644	−0.02305057	−0.02305057
$\partial g/\partial a_0$	0.82464012	0.82464012	0.71847410	0.71847410
$\partial g/\partial a_1$	0.30311507	0.30311507	0.36777630	0.36777630

A DATA-ASSIMILATION EXAMPLE FROM GLACIOLOGY

Mathematical models of glaciers typically include parameters which are difficult to estimate directly from field measurements. For example, some type of frictional coefficient is required at the base of the glacier where the ice rests upon the earth. The basal traction (described mathematically using this frictional coefficient) affects the velocity of the ice throughout the glacier. Thus, it is plausible to use the more easily measured velocity at the upper surface of the glacier to determine the frictional coefficient. In this concluding section we present an example problem, using a simplified geometry to demonstrate the use of the adjoint method to accomplish this.

The example problem is related to a benchmark problem, 'Experiment D' of the Ice Sheet Model Intercomparison Project for Higher-Order Models (ISMIP-HOM) (Pattyn and others, 2008). In ISMIP-HOM Experiment D, a sheet of ice with uniform thickness of 1000 m rests on a plane inclined at an angle $\alpha = 0.1°$. A domain of length 20 km with periodic boundary conditions at each end is considered. A numerical model is used to calculate the velocity of the ice, which varies not only throughout the thickness of the ice but also along the length of the domain, due to variations in basal traction. There is no variation in the horizontal direction transverse to the inclination.

In ISMIP-HOM Experiment D, the basal traction is prescribed and ice velocities (and pressure) are calculated. We call this a 'forward problem'. Here we are interested in a related 'inverse problem': given the surface velocity, find the basal traction.

The model used is a finite-element approximation* of Stokes' equations[†]

$$\frac{\partial}{\partial x}\left(2\mu\frac{\partial u}{\partial x} - p\right) + \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)\right] = -\rho g\sin\alpha,$$

$$\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)\right] + \frac{\partial}{\partial y}\left(2\mu\frac{\partial v}{\partial y} - p\right) = \rho g\cos\alpha,$$

along with conservation of mass,

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

In these equations, x and y are the coordinate directions, x along the incline and y perpendicular to it (almost vertical). The components of velocity in the x- and y-directions are given by u and v, respectively. Pressure is represented by the variable p, while $\rho = 910\,\text{kg}\,\text{m}^{-3}$ is the density of ice and $g = 9.81\,\text{m}\,\text{s}^{-2}$ is the acceleration due to gravity. The viscosity, μ, varies according to Glen's flow law[‡]

$$\mu = \frac{1}{2}A^{-\frac{1}{n}}\dot{\epsilon}^{-\frac{n-1}{n}},$$

with the effective strain rate

$$\dot{\epsilon}^2 = \frac{1}{2}\left[\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2\right] + \frac{1}{4}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)^2,$$

and flow parameters $A = 10^{-16}\,\text{Pa}^{-n}\,\text{a}^{-1}$ and $n = 3$.

Stress-free boundary conditions apply at the upper surface

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

$$2\mu\frac{\partial v}{\partial y} - p = 0,$$

while at the base $v = 0$ and basal traction is modeled using the boundary condition

$$\mu\frac{\partial u}{\partial y} = \beta^2 u.$$

In ISMIP-HOM Experiment D

$$\beta^2 = 1000 + 1000\sin\left(\frac{2\pi x}{L}\right),$$

where L is the length of the domain (20 km for the problem presented here). This frictional coefficient is plotted as a heavy black curve in the lower left panel of Figure 4. The upper left panel of Figure 4 shows the velocity component, u, at the (top) surface of the ice for a solution of the 'forward' problem (using $\beta^2 = 1000 + 1000\sin(2\pi x/L)$) plotted as a heavy black line.

Now, for the inverse problem. Suppose we are given the surface velocity component, u, plotted as a heavy black curve in the upper left panel of Figure 4. We hope to use this information to discover the basal friction coefficient, β^2 (plotted as a heavy black curve in the lower left panel of Fig. 4). To do so we pose the problem as an optimization problem: find β^2 such that the surface velocity component, u, found by solving the 'forward' problem minimizes the error given by

$$g(u) = \int_0^L (u - u_\text{d})^2\,\mathrm{d}x,$$

where u_d is the desired surface velocity. To discretize the function β^2 a trigonometric expansion[§] is used,

$$\beta^2 = p_0 + \sum_{k=1}^{N/2} p_{2k-1}\sin\left(\frac{2\pi kx}{L}\right) + p_{2k}\cos\left(\frac{2\pi kx}{L}\right).$$

A variety of algorithms for solving optimization problems exist (Press, 2007). Here we choose the Broyden–Fletcher–Goldfarb–Shano (BFGS) algorithm (Press, 2007, p. 521–524), which uses partial derivatives of the objective function, $g(u)$, with respect to the parameters, p_k, that can be varied. These partial derivatives $(\partial g/\partial p_k)$ are *exactly* what the adjoint

---

*The finite-element method is a technique for discretizing a boundary-value problem, approximating its solution using the solution to a system of algebraic equations. One step in the method is to partition the domain of the problem into a set of 'elements'. Here a uniform grid with 32 elements along the x-axis and 16 elements along the y-axis was used.

[†]Stokes' equations represent Newton's second law, $\sum \mathbf{F} = m\mathbf{a}$, for a viscous fluid when the inertial term, $m\mathbf{a}$, is negligible. The equations presented here have been simplified using the assumption that derivatives with respect to z (the transverse direction) are zero.

[‡]Glen's flow law is a nonlinear constitutive equation used to model the relationship between stress and strain rates in ice. This relationship is temperature-dependent, but the model used here assumes that the ice is isothermal.

[§]Fourier series, which are trigonometric expansions of this form with an infinite number of terms, can be used to represent any differentiable periodic function (Tolstov, 1962). Truncated Fourier series, such as that used here, are often used to approximate functions of unknown form. An alternate discretization of β^2 would be an expansion in terms of the 'test functions' used in the finite-element method. This could result, for example, in approximating β^2 with a continuous piecewise linear function.
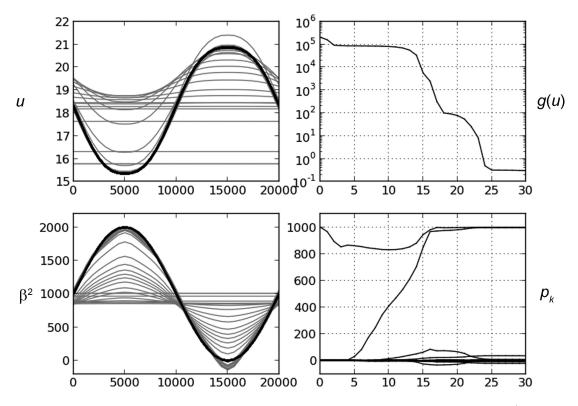
Fig. 4. Solution of an inverse problem. The component of the ice velocity in the direction parallel to the bed (u; m a^{-1}) at the (top) surface of the ice is shown in the upper left panel. This velocity evolves towards the desired solution (the heavy black curve) as the basal friction (β^2, plotted in the lower left panel) is changed. The upper right panel shows the error, $g(u)$, decreasing as the coefficients, p_k, in the trigonometric expansion of β^2 (shown in the lower right panel) change.

method described in the previous sections can provide. Starting with a 'guess' for β^2, an approximate solution to the inverse problem is obtained by repeating the following steps:

1. Solve the 'forward' problem with the current estimate of β^2.

2. Evaluate the error, $g(u)$.

3. Evaluate the partial derivatives, $\partial g / \partial p_k$, using the adjoint method.

4. Calculate an improved estimate of β^2 (i.e. the p_k) according to the BFGS algorithm.

The results of applying this process are shown in Figure 4. The initial guess was $\beta^2 = 1000$; i.e. $p_0 = 1000$ and $p_k = 0$ for $1 \leq k \leq N$ where, here, $N = 30$. The gray curves in the left panels of Figure 4 show approximations of u and β^2 as the process progressed. The original guess of $\beta^2 = 1000$ (represented by a horizontal line in the lower left panel of Fig. 4) resulted in the bottommost horizontal line in the upper left panel of Figure 4. As the estimate of β^2 improves, the curves approach the desired values plotted as heavy black curves. The upper right panel of Figure 4 shows the error, $g(u)$, decreasing throughout the process. Further improvement can be made by allowing more iterations of the algorithm. The lower right panel of Figure 4 shows the evolution of the coefficient (p_k) values. The upper curve represents p_0 which starts at 1000, dips down for a while but returns to a value very near 1000. The curve that increases from zero to near 1000 represents p_1. The remainder of the curves, which remain near zero, represent p_2 through p_{30}. In the optimal solution these values are all zero, but the largest, p_9, was 36.9 when the process was terminated. Again, further improvement can be made by allowing more iterations.

REFERENCES

Brinkerhoff DJ, Meierbachtol TW, Johnson JV and Harper JT (2011) Sensitivity of the frozen/melted basal boundary to perturbations of basal traction and geothermal heat flux: Isunguata Sermia, western Greenland. *Ann. Glaciol.*, **52**(59), 43–50 (doi: 10.3189/172756411799096330)

Errico RM (1997) What is an adjoint model? *Bull. Am. Meteorol. Soc.*, **78**(11), 2577–2591 (doi: 10.1175/1520-0477(1997)078<2577:WIAAM>2.0.CO;2)

Granzow GD (2013) An investigation of viscosity using measured velocities on Helheim Glacier. (Master's thesis, University of Montana)

Pattyn F and 20 others (2008) Benchmark experiments for higher-order and full-Stokes ice sheet models (ISMIP-HOM). *Cryosphere*, **2**(2), 95–108 (doi: 10.5194/tc-2-95-2008)

Press WH (2007) *Numerical recipes: the art of scientific computing*, 3rd edn. Cambridge University Press, Cambridge

Strang G (2007) *Computational science and engineering.* Wellesey-Cambridge, Wellesey, MA

Tolstov GP [transl. A. Silverman] (1962) *Fourier series.* Prentice-Hall, Englewood Cliffs, NJ