

# Time-step limits for stable solutions of the ice-sheet equation

RICHARD C. A. HINDMARSH

*British Antarctic Survey, Natural Environment Research Council, Cambridge CB3 0ET, England*

ANTONY J. PAYNE

*Geowissenschaften FB5, Universität Bremen, Bremen, Germany*

**ABSTRACT.** Various spatial discretizations for the ice sheet are compared for accuracy against analytical solutions in one and two dimensions. The computational efficiency of various iterated and non-iterated marching schemes is compared.

The stability properties of different marching schemes, with and without iterations on the non-linear equations, are compared. Newton–Raphson techniques permit the largest time steps. A new technique, which is based on the fact that the dynamics of unstable iterated maps contain information about where the unstable root lies, is shown to improve substantially the performance of Picard iteration at a negligible computational cost.

## INTRODUCTION

The ice-sheet thickness-evolution equation (where the mechanics are computed according to the shallow-ice approximation of Hutter (1983)) is of a highly non-linear diffusion-reaction type and generally needs to be solved numerically. The time steps used in the solution of this equation are limited by the requirements of stability: if the time steps are too large, then spurious oscillations are created in the ice-sheet surface. Unlike the linear diffusion equation, this does not necessarily result in blow-up because non-linear coupling of the instability prevents unbounded growth.

A number of approaches exist for the numerical solution of this equation. They can be grouped as explicit, semi-implicit (where there is no sub-time step non-linear iteration) and fully implicit (where there is full sub-time step non-linear iteration). The former two schemes are most often used in operational ice-sheet models. Their stability characteristics limit the maximum time step used in most practical applications to 1–10 years. This is considerably shorter than the natural time-scale of ice sheets and limits the number of experiments that can be made with continental ice-sheet models.

Budd and Jenssen (1976) have investigated the stability of explicit schemes for solving the ice-sheet equations. However, since then, considerable progress has been made in the theory of iterated maps (which is what numerical solutions are), and there has been widespread adoption of semi-implicit schemes (e.g. Mahaffy, 1976; Huybrechts, 1992), and implicit or Crank–Nicholson

schemes which iterate to solve the non-linear simultaneous equations using Picard or Newton–Raphson iteration (e.g. Hindmarsh and others, 1987; Hindmarsh and Hutter, 1988). All of these methods have time-step limits, which should be contrasted with the linear diffusion equation where Crank–Nicholson and implicit schemes are uniformly stable.

In this paper, we will look at the theory and practice of time-step limitations for the ice-sheet equation. First, we set out several of the more commonly used discretizations of the ice-sheet equation. We then present comparisons between these different discretizations in terms of accuracy, stability and efficiency. This is done in two idealized experiments for which analytical solutions are available: the one-dimensional case flow-law exponent ( $n$ ) is equal to 3 and the two-dimensional case where the exponent is equal to 1. Finally, we present the concept of the iterated map and introduce a method for improving the stability properties of the time-stepping method used to solve the ice-sheet equation.

## 1. BACKGROUND: THE ICE-SHEET EQUATION AND ITS FINITE-DIFFERENCE DISCRETIZATIONS

The ice-sheet equation is

$$\frac{\partial H}{\partial t} = \nabla \cdot (CH^{n+2} |\nabla H|^{n-1} \nabla H) + a \quad (1)$$

where  $H$  is ice thickness (throughout we deal only with a

flat basal topography) and  $a$  is the accumulation rate.  $C$  is related to the flow law parameter  $A$  by

$$C = \frac{2A(\rho g)^n}{n + 2}$$

where  $\rho$  is the density of ice,  $g$  is acceleration due to gravity,  $A$  is the ice-flow constant,  $n$  is its exponent and the flow relation between strain rate  $e$  and deviator stress  $\tau$  is given by

$$e = A\tau^n.$$

We deal only with constant  $A$  (i.e. with isothermal ice).

$\nabla$  is the two-dimensional horizontal gradient operator

$$\nabla \equiv \left( \frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$$

and the magnitude of the gradient of  $|\nabla H|$  is computed by

$$|\nabla H| \equiv \left| \left( \frac{\partial H}{\partial x} \right)^2 + \left( \frac{\partial H}{\partial y} \right)^2 \right|^{\frac{1}{2}}.$$

The non-linear diffusion Equation (1) is solved by discretizing the independent variables as

$$x = i\Delta_x, i \in (0, N_x); \quad y = j\Delta_y, j \in (0, N_y)$$

for the horizontal dimensions  $x$  and  $y$  over the domain  $-L_x \leq x \leq L_x, -L_y \leq y \leq L_y$ ; and

$$t = k\Delta_t, \quad k \in (i, \infty)$$

for time  $t$ , where  $N_x$  and  $N_y$  are the number of grids in the  $x$  and  $y$  directions.

### 1.1. The horizontal space discretization

The discretization of the non-linear diffusion term

$$\nabla \cdot \left( CH^{n+2} |\nabla H|^{n-1} \nabla H \right) \tag{2}$$

calls for some comment. There are many ways of carrying out a discretization of this operator. Here we define three methods that are in common use and which are schematized in Figure 1. It is convenient to define a coefficient  $D$  somewhat analogous to the diffusivity of a thermal medium

$$D = CH^{n+2} |\nabla H|^{n-1} \tag{3}$$

so that Equation (1) becomes

$$\frac{\partial H}{\partial t} = \nabla \cdot (D\nabla H) + a \equiv -\nabla \cdot \mathbf{q} + a. \tag{4}$$

In *method 1*,  $D$  is evaluated at grid centres which are

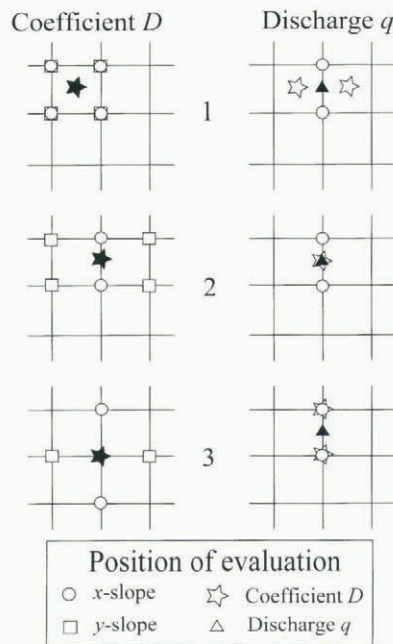


Fig. 1. Illustrating the different grids used to compute the coefficient  $D$  and the discharge  $q$ . The lefthand column shows which points are used in the computation of the average value of  $H$  and  $|\nabla H|$  used in the computation of  $D$ , while the righthand column shows the location of  $D$  and points used in the evaluation of  $\partial H/\partial x$  required to compute  $q$ .

staggered in both  $x$  and  $y$  according to

$$D_{i+\frac{1}{2},j+\frac{1}{2}} = CH_{i+\frac{1}{2},j+\frac{1}{2}}^{n+2} \left| (\nabla H)_{i+\frac{1}{2},j+\frac{1}{2}} \right|^{n-1} \tag{5}$$

where

$$H_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{4}(H_{i,j} + H_{i+1,j} + H_{i,j+1} + H_{i+1,j+1})$$

and

$$\begin{aligned} (\nabla H)_{i+\frac{1}{2},j+\frac{1}{2}} = & \frac{1}{2\Delta_x} (H_{i+1,j} + H_{i+1,j+1} - H_{i,j} - H_{i,j+1}) \\ & + \frac{1}{2\Delta_y} (H_{i,j+1} + H_{i+1,j+1} - H_{i,j} - H_{i+1,j}). \end{aligned}$$

The  $x$ -direction flux  $q^x$  is then evaluated at  $x$ -midway points according to

$$q_{i+\frac{1}{2},j}^x = -\frac{1}{2} \left( D_{i+\frac{1}{2},j+\frac{1}{2}} + D_{i-\frac{1}{2},j+\frac{1}{2}} \right) \frac{H_{i+1,j} - H_{i,j}}{\Delta_x}$$

and similarly for  $q_{i-\frac{1}{2},j}^x$ .

The  $y$ -direction flux  $q^y$  is evaluated at  $y$ -midway points according to

$$q_{i,j+\frac{1}{2}}^y = -\frac{1}{2} \left( D_{i+\frac{1}{2},j+\frac{1}{2}} + D_{i-\frac{1}{2},j+\frac{1}{2}} \right) \frac{H_{i,j+1} - H_{i,j}}{\Delta_y}$$

and similarly for  $q_{i,j-\frac{1}{2}}^y$ .

For *method 2*,  $D$  is evaluated at  $x$ -midway and  $y$ -midway points which are staggered in the direction of



flow only

$$D_{i+\frac{1}{2},j} = CH_{i+\frac{1}{2},j}^{n+2} |(\nabla H)_{i+\frac{1}{2},j}|^{n-1}$$

where

$$H_{i+\frac{1}{2},j} = \frac{1}{2}(H_{i,j} + H_{i+1,j})$$

and

$$(\nabla H)_{i+\frac{1}{2},j} = \frac{1}{2\Delta_x} (H_{i+1,j} - H_{i,j}) + \frac{1}{4\Delta_y} (H_{i,j+1} + H_{i+1,j+1} - H_{i,j-1} - H_{i+1,j-1}).$$

The  $x$ -direction flux  $q^x$  is then evaluated at the same  $x$ -midway points according to

$$q_{i+\frac{1}{2},j}^x = -D_{i+\frac{1}{2},j} \frac{H_{i+1,j} - H_{i,j}}{\Delta_x}$$

and similarly for  $q_{i-\frac{1}{2},j}^x$ .

The  $y$ -direction flux  $q^y$  is evaluated at  $y$ -midway points according to

$$q_{i,j+\frac{1}{2}}^y = -D_{i,j+\frac{1}{2}} \frac{H_{i,j+1} - H_{i,j}}{\Delta_y}$$

and similarly for  $q_{i,j-\frac{1}{2}}^y$ . This scheme derives from Mahaffy (1977).

In *method 3*,  $D$  is evaluated at the grid points themselves

$$D_{i,j} = CH_{i,j}^{n+2} |(\nabla H)_{i,j}|^{n-1}$$

where

$$(\nabla H)_{i,j} = \frac{1}{2\Delta_x} (H_{i+1,j} - H_{i-1,j}) + \frac{1}{2\Delta_y} (H_{i,j+1} - H_{i,j-1}).$$

The  $x$ -direction flux  $q^x$  is then evaluated at the staggered  $x$ -midway points according to

$$q_{i+\frac{1}{2},j}^x = -\left(\frac{D_{i+1,j} + D_{i,j}}{2}\right) \left(\frac{H_{i+1,j} - H_{i,j}}{\Delta_x}\right)$$

and similarly for  $q_{i-\frac{1}{2},j}^x$ .

The  $y$ -direction flux  $q^y$  is evaluated at  $y$ -midway points according to

$$q_{i,j+\frac{1}{2}}^y = -\left(\frac{D_{i,j+1} + D_{i,j}}{2}\right) \left(\frac{H_{i,j+1} - H_{i,j}}{\Delta_y}\right)$$

and similarly for  $q_{i,j-\frac{1}{2}}^y$ . This scheme derives from Oerlemans and Van der Veen (1984) (in one dimension) and Huybrechts (1992) (in two dimensions).

In all three methods, the flux divergence in Equation (4) is computed according to

$$(\nabla \cdot \mathbf{q})_{i,j} = \frac{q_{i+\frac{1}{2},j}^x - q_{i-\frac{1}{2},j}^x}{\Delta_x} + \frac{q_{i,j+\frac{1}{2}}^y - q_{i,j-\frac{1}{2}}^y}{\Delta_y}. \quad (6)$$

All these methods are conservative because the thickness calculations at two adjacent grid points  $(i, j)$  and  $(i + 1, j)$  both use exactly the same formulation for

$q_{i+\frac{1}{2},j}^x$ . All the material leaving one grid point arrives at adjacent grid points, except of course at the boundary. However, while methods 1 and 2 have a 9 point computational molecule (for computing the flux gradient), method 3 has a 13 point computational molecule. This is a very important feature of method 3 and can be interpreted as incorporating an additional, artificial diffusion (or smoothing).

The accuracy of the method 1 and 2 spatial discretizations is not uniform: in the interior of the ice sheet it is second-order, but the singular margins imply that the ice-sheet surface is not analytic and thus that Taylor series do not constitute a valid approximation at these points. Numerical results indicate that the scheme is something like first-order accurate in space.

### 1.2. The time discretization

We now turn to the discretization of Equation (4) in time, which we denote by

$$\frac{H_{i,j}^{k+1} - H_{i,j}^k}{\Delta_t} = -(\nabla \cdot \mathbf{q})_{i,j}^{k+\theta, k+\omega} + a^{k+\theta} \quad (7)$$

where the superscripts refer the time at which the term is calculated. In particular, the two superscripts on the flux  $\mathbf{q}$  refer to the time at which the two factors  $(D^{k+\theta}$  and  $(\nabla H)^{k+\omega}$ ) of  $\mathbf{q}$  are calculated.

We define three time-stepping schemes. When  $\omega = 0$ ,  $\theta = 0$  we have an explicit scheme. When  $\omega = 1$ ,  $\theta = 0$  we have the so-called semi-implicit scheme, which is probably the most widely used scheme in glaciology. When  $\omega = 1$ ,  $\theta = 1$  we have a fully implicit scheme. In addition to these first-order time discretizations, there is the second-order accurate Crank–Nicholson scheme where  $\omega = 1/2$  and  $\theta = 1/2$ .

The semi-implicit and implicit schemes imply a linear iteration (e.g. a relaxation or a conjugate-gradient method) or a direct matrix-inversion technique (e.g. Gaussian elimination) to solve the linear matrix equation implied by Equation (7). The implicit scheme implies a further, non-linear iteration to cope with the non-linear nature of  $D$ . This non-linear iteration may be accomplished by either Picard or Newton–Raphson iteration. These are both well-established techniques which are described in texts on numerical analysis (e.g. Johnson and Riess 1982; Ekman 1987).

We will return to the subject of time stepping in a later section, where the concept of the iterated map will be introduced and used to analyse the stability properties of these time-stepping methods. We now turn to the comparison of the different discretization methods in space and time.

### A comparison exercise

In this section we provide comparisons of the accuracy, efficiency and stability of the spatial and temporal discretizations described above. We use two experiments which have analytical solutions available to investigate

Table 1. Divide thickness found using methods 2 and 3 for various  $\Delta_y$  in experiment I

Grid km	Number of points in $y$	Divide thickness			
		Scheme 2	Scheme 3	Scheme 2 error $\times 10^{-3}$	Scheme 3
10	151	3580.0226	3562.8913	1.4	-3.4
25	61	3587.6580	3546.0067	3.5	-8.1
50	31	3600.5068	3520.0102	7.1	-15.4
75	21	3613.3609	3496.1669	10.7	-22.1

the effects of spatial discretization on the accuracy of the calculated steady-state ice thickness. We then go on to assess the stability and efficiency of the various time-stepping methods.

**Description of the experiments and their analytical solutions**

We first describe the two analytical solutions available for steady-state ice thickness.

*Experiment I: one dimension with  $n = 3$*

In one dimension with constant  $a$ , we have from continuity (Vialov, 1958; Nye, 1959)

$$ay = q^y = -\frac{2A}{n+2} \left| \rho g \frac{\partial H}{\partial y} \right|^n H^{n+2}.$$

Over the half-domain  $y = 0$  (divide) to  $y = L_y$  (margin) with boundary conditions  $H(0) = H_o$  and  $H(L_y) = 0$ , and taking  $n = 3$  we obtain

$$-\left(\frac{2A}{5}\right)^{\frac{1}{3}} (\rho g) \int_H^0 H^{\frac{5}{3}} dH = \int_y^{L_y} (ay)^{\frac{1}{3}} dy$$

which can be integrated to

$$H^{\frac{8}{3}} = \left(\frac{20a}{A}\right)^{\frac{1}{3}} \left(\frac{1}{\rho g}\right) \left(L_y^{\frac{4}{3}} - y^{\frac{4}{3}}\right).$$

At  $y = 0$

$$H_o = \left(\frac{20a}{A}\right)^{\frac{1}{3}} \left(\frac{1}{\rho g}\right)^{\frac{3}{8}} L_y^{\frac{1}{2}}$$

giving

$$\left(\frac{H}{H_o}\right)^{\frac{8}{3}} = 1 - \left(\frac{y}{L_y}\right)^{\frac{4}{3}}. \tag{8}$$

*Experiment II: two dimensions with  $n = 1$*

In this experiment, we consider the ice-sheet equation for a linear rheology. The flux formula is simply

$$\mathbf{q} = -\frac{2A\rho g}{3} H^3 \nabla H$$

and the condition for steady state is

$$\nabla \cdot \mathbf{q} = a$$

with finite flux-boundary conditions on  $x = \pm L_x$ ,  $y = \pm L_y$  of  $H = 0$ . We may write

$$\mathbf{q} = -\frac{A\rho g}{6} \nabla H^4$$

and by defining a new variable  $u = H^4$  we see

$$\mathbf{q} = -\frac{A\rho g}{6} \nabla u$$

and then obtain

$$\nabla^2 u = -\frac{6a}{A\rho g}$$

which is simply the Poisson equation on a rectangle, which has an analytical solution (Carslaw and Jaeger, 1988, p. 171). One may then write down the solution for  $H(x, y)$ , which is

$$H(x, y) = u^{\frac{1}{4}} = \left(\frac{6a}{A\rho g} \left\{ \frac{(L_x^2 - x^2)}{2} - \frac{16L_x^2}{\pi^3} F \right\}\right)^{\frac{1}{4}} \tag{9}$$

where

$$F = \sum_{j=0}^{\infty} \frac{(-1)^j \cos(\{2j+1\}\pi x/2L_x) \cosh(\{2j+1\}\pi y/2L_x)}{(2j+1)^3 \cosh(\{2j+1\}\pi L_y/2L_x)}.$$

**1.3. The accuracy of various spatial discretizations**

The first set of experiments aims at assessing the behaviour of two spatial discretization methods over various grid resolutions. The discretization methods are the 9 point molecule method 2 and the 13 point molecule method 3 defined above.

A constant time step of 10 years was used for method 2 and a time step of 0.1 year was used in method 3. These time steps assure a stable solution but are not the largest stable time step for each scheme. Time stepping was fully implicit with solution of the non-linear algebraic equations by Newton–Raphson iteration in method 2 and was explicit in method 3, although the various time-stepping methods yield the same results for a given spatial discretization and grid spacing, as we are dealing with steady solutions.



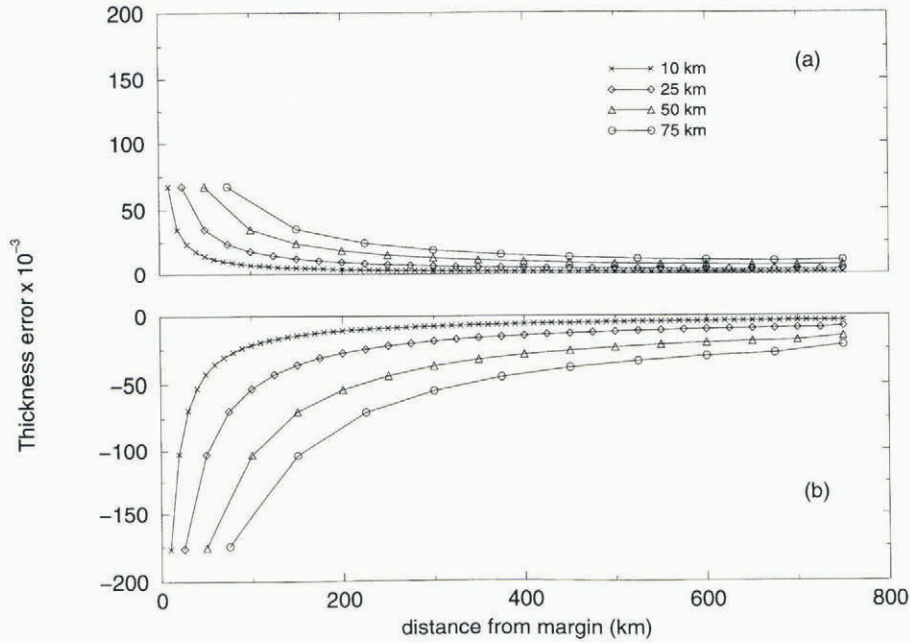


Fig. 2. A comparison of the numerical error from margin to divide using different  $\Delta_y$ , for spatial discretizations 2(a) and 3(b) in experiment I.

In all experiments, the ice-thickness evolution Equation (1) was integrated from zero ice thickness for 100kyears, by which time a steady state had been reached. A constant accumulation rate ( $a$ ) of  $0.3 \text{ m year}^{-1}$  was used and the viscosity of ice ( $A$ ) was set to  $10^{-16} \text{ Pa}^{-3} \text{ year}^{-1}$  in experiment I and to  $2.1 \times 10^{-7} \text{ Pa}^{-1} \text{ year}^{-1}$  in experiment II (resulting in similar ice thicknesses to those of experiment I). In addition,  $\rho$  and  $g$  were taken as  $910 \text{ kg m}^{-3}$  and  $9.81 \text{ m s}^{-2}$ , respectively. A flat topography was used throughout.

Experiment I: one dimension with  $n = 3$

This experiment was conducted on a rectangular grid on which  $\Delta_x = \Delta_y$ . The length of the domain in the  $y$  direction was held constant at 1500 km (i.e.  $2L_y = 1500 \text{ km}$ ). At the upper and lower edges of the grid, thickness was set to zero. Cyclic boundary conditions were employed on the left and right edges of the domain, so that information passed freely from one boundary to the other. The domain was therefore cylindrical. The ice

sheet which evolves should always have symmetry in the  $y$  direction about  $y = 0 \text{ km}$  and will have constant thickness along  $x$  for any given  $y$ . The advantage of this set-up is that, although a two-dimensional problem is solved, the one-dimensional analytical solution above is available for comparison with the results.

Table 1 shows the steady-state divide thicknesses for each run after 100kyears. In this table and elsewhere, the error  $\mathcal{E}$  is defined as

$$\mathcal{E} = \frac{H_{\text{num}}(x, y) - H_{\text{ana}}(x, y)}{H_{\text{ana}}(x, y)}$$

where  $H_{\text{num}}$  is the numerical thickness and  $H_{\text{ana}}$  the analytical one.

The analytical solution for the steady-state divide thickness is 3575.058 m. A linear regression on the four runs for method 2 yielded

$$H_{\text{num}} = 3574.862 + 0.513\Delta_y$$

Table 2. Divide thickness found using methods 2 and 3 for various  $\Delta_{x,y}$  in experiment II

Grid km	Number of points in $x y$	Divide thickness			
		Scheme 2	Scheme 3	Scheme 2 error $\times 10^{-3}$	Scheme 3
10	151	3575.0527	3538.8643	6.5	-3.7
25	61	3607.5904	3519.8241	15.7	-9.0
50	31	3656.7418	3488.8749	29.5	-17.7
75	21	3700.6105	3458.7609	41.9	-26.2

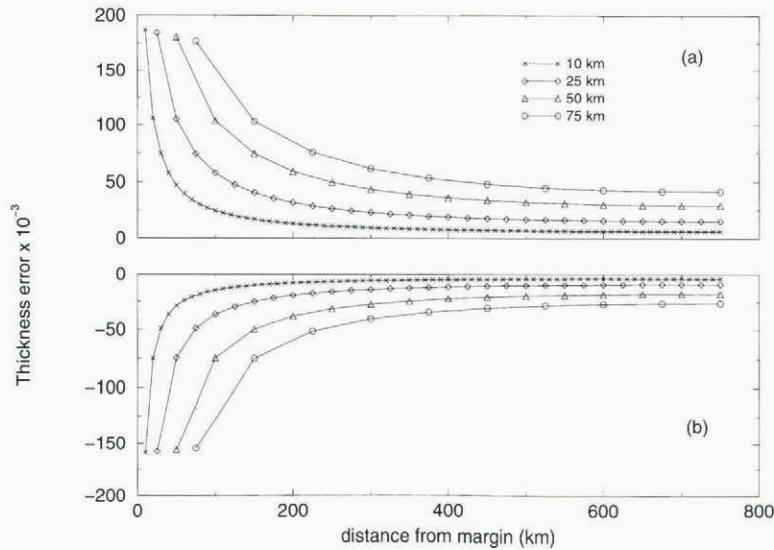


Fig. 3. A comparison of the numerical error from margin to divide using different  $\Delta_{x,y}$ , for spatial discretizations 2(a) and 3(b) in experiment II.

where  $\Delta_y$  is the grid spacing in kilometres. This relation has a correlation coefficient of 1.0. The intercept is within  $5.5 \times 10^{-5}$  (0.196 m) of the analytical solution.

The steady-state divide thicknesses obtained in the method 3 experiments are also shown in Table 1. A linear regression on the four runs yielded

$$H_{\text{num}} = 3572.263 - 1.025\Delta_y$$

with a correlation coefficient of  $-1.0$ . The intercept for method 3 is within  $7.8 \times 10^{-4}$  (2.795 m) of the analytical solution.

The very high correlation coefficients indicate that both spatial discretization methods are first-order accurate and have a linear convergence to the analytical solution as the grid spacing goes to zero. Method 2 is particularly accurate.

Figure 2a and b shows the variation of error from the margin to the divide in each of the experiments using, respectively, methods 2 and 3. The relationship between the two schemes' error remains fairly constant from margin to divide and varies between  $\times 2$  and  $\times 3$ . In all cases, the error grows dramatically towards the margin.

*Experiment II: two dimensions with  $n = 1$*

This experiment was conducted on a rectangular grid on which  $\Delta_x = \Delta_y$ . The length of the domain in the  $x$  and  $y$  directions was held constant at 1500 km. Ice thickness was set to zero along all four edges of the grid.

Table 2 shows the steady-state divide thicknesses for each run after 100kyears. The analytical solution for divide thickness is 3551.862 m. A linear regression on the four runs for method 2 yielded

$$H_{\text{num}} = 3557.883 + 1.928\Delta_{x,y}$$

This relation has a correlation coefficient of 1.0. The intercept is within  $1.7 \times 10^{-3}$  (6.021 m) of the analytical solution.

The steady-state divide thickness obtained in the method 3 experiments are also shown in Table 2. A linear

regression on the four runs yielded

$$H_{\text{num}} = 3550.852 - 1.232\Delta_{x,y}$$

with a correlation coefficient of  $-1.0$ . The intercept for method 3 is within  $2.8 \times 10^{-4}$  (1.010 m) of the analytical solution.

Again, the high correlation coefficients indicate that convergence to the analytical solution is almost linear. However, method 3 is now the more accurate.

Figure 3a and b shows the variation of error from the margin ( $x = 750$  km,  $y = 750$  km) to the divide ( $x = 0$  km,  $y = 750$  km) in each of the experiments using, respectively, methods 2 and 3.

Both experiments indicate that the accuracy of the spatial discretizations is approximately linear with  $\Delta_{x,y}$ . The 9 point molecule discretization (method 2) overestimates steady-state ice thickness mainly because of error in the area of steep surface slopes near the margin. This is particularly true in the  $n = 1$  experiment, where the near-margin surface slopes are much larger than those in the  $n = 3$  experiment. The 13 point molecule discretization (method 3) consistently underpredicts

Table 3. Divide thickness found using methods 2 and 3 for various  $\Delta_{x,y}$  in experiment III

Grid spacing km	Number of points in x y	Divide thickness	
		Scheme 2 m	Scheme 3 m
10	151	3401.9364	3385.4924
25	61	3409.1807	3369.0222
50	31	3420.5050	3342.6250
75	21	3430.6165	3317.1001



Table 4. Comparison of spatial discretization and time-stepping methods for  $\Delta_{x,y} = 75 \text{ km}$

Spatial discret- ionary method	Time-stepping method	Time step in years	Number of time steps	Number of linear iterations	Number of non- linear iterations	Normal CPU time
2	Implicit	10 000	10	537	36	1.0
2	Implicit	100	1000	6989	1233	24.0
2	Semi-implicit	112	892	1840	–	7.0
2	Explicit	23	4347	–	–	14.0
3	Semi-implicit	621	161	1262	–	2.0
3	Explicit	51	1960	–	–	3.0

Table 5. Comparison of spatial discretization and time-stepping methods for  $\Delta_{x,y} = 50 \text{ km}$

Spatial discret- ionary method	Time-stepping method	Time step in years	Number of time steps	Number of linear iterations	Number of non- linear iterations	Normal CPU time
2	Implicit	10 000	10	870	41	1.0
2	Implicit	100	1000	9119	1235	13.4
2	Semi-implicit	45	2222	4059	–	8.0
2	Explicit	10	10 000	–	–	15.2
3	Semi-implicit	331	302	1618	–	1.8
3	Explicit	21	4761	–	–	3.2

Table 6. Comparison of spatial discretization and time-stepping methods for  $\Delta_{x,y} = 25 \text{ km}$

Spatial discret- ionary method	Time-stepping method	Time step in years	Number of time steps	Number of linear iterations	Number of non- linear iterations	Normal CPU time
2	Implicit	10 000	10	1907	57	1.0
2	Implicit	100	1000	14 968	1239	9.2
2	Semi-implicit	10	10 000	15 662	–	15.1
2	Explicit	2	50 000	–	–	30.2
3	Semi-implicit	96	1041	3835	–	2.0
3	Explicit	5	20 000	–	–	5.4

thickness. This is probably because of the apparent artificial diffusion introduced in this discretization (discussed above). In the  $n = 1$  experiment, this diffusion leads to a better overall estimation of steady-state ice thickness.

**1.4. The stability and efficiency of various time-stepping discretizations**

This experiment is a replication of the EISMINT fixed-margin benchmark, which is two-dimensional and takes  $n = 3$  (Huybrechts and others, 1996). First, the accuracy of the two spatial discretizations discussed above is again compared, however without the opportunity of comparison with an analytical solution.

Table 3 shows the steady-state divide thicknesses for

each run after 100 kyears. A linear regression on the four runs for method 2 yielded

$$H_{\text{num}} = 3397.914 + 0.441\Delta_{x,y}$$

with a correlation coefficient of 1.0. The linear regression for method 3 yielded

$$H_{\text{num}} = 3395.606 - 1.051\Delta_{x,y}$$

with a correlation coefficient of  $-1.0$ . These results are close to those found in the  $n = 3$  one-dimensional experiment above.

The main purpose of this experiment is to compare the efficiency of various time-stepping methods. Three methods are used in combination with spatial discretization method 2. These were the explicit, semi-implicit and implicit methods. In addition, the explicit and semi-implicit methods were used with spatial discretization

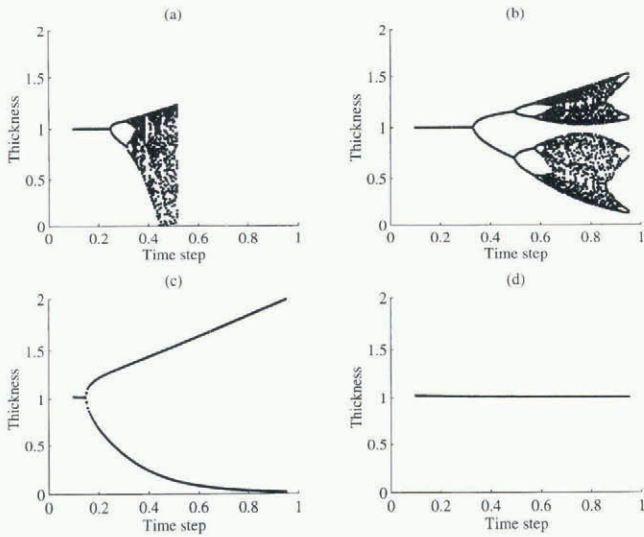


Fig. 4. Bifurcation maps for the zero-dimensional ODE model. Vertical axis is  $I$ , horizontal axis is  $\Delta_t$ . (a) is an explicit scheme, (b) is an implicit scheme, (c) is a Picard iteration and (d) is a Newton–Raphson scheme. The points represent successive values of  $I^k$  as  $k \rightarrow \infty$  (a and b) and successive values of  $I^{k,\ell}$  as  $\ell \rightarrow \infty$  (c and d).

method 3. No vectorization or parallel techniques were used in the computations.

The semi-implicit and implicit methods require the solution of a sparse, linear system because of the two-dimensional nature of the problem. Here, a conjugate gradient algorithm from Press and others (1992) was used. This will be referred to as the linear iteration. In addition, the implicit method requires the solution of a non-linear system because of the nature of  $D$  in Equation (4). A Newton–Raphson partial derivative method was used. This will be called the non-linear iteration. The implicit method therefore has two nested iterations: an outer, non-linear one and an inner, linear one.

The maximum stable time step for each combination of methods is reported in Tables 4–6 for  $\Delta_{x,y} = 75, 50$  and 25 km, respectively. Each maximum stable time step was checked to the nearest 1 year. The implicit method proved to be very stable and testing was stopped after a 10 kyear time step had been reached. Clearly, this time step is far longer than that required in most practical problems. The runs using the implicit scheme were therefore repeated using 100 year time steps, giving a better impression of the efficiency of this scheme for practical applications.

The numbers of linear and non-linear iterations, and the normalized (within each table) CPU requirements are also reported in the tables. Although the individual runs lasted for 100 kyears, the ice sheet took approximately 25–27 kyears to reach equilibrium. After this time the number of linear and non-linear iterations required fell dramatically.

The onset of instability depends on the method of time stepping. For explicit and Newton–Raphson implicit stepping, ice thicknesses grow explosively. However, in the semi-implicit method, thicknesses slowly develop oscillations. The criteria for assessing stability was there-

fore that the divide thickness should not differ by more than  $1 \times 10^{-4}$  m from the stable thicknesses given in Table 3.

The use of the method 3 spatial discretization allows for a 2–10 times increase in the maximum stable time step over method 2, which reduces CPU requirement by 3–4 times. The combination of semi-implicit time stepping with this spatial discretization is particularly efficient.

The use of semi-implicit rather than explicit time stepping increases the maximum stable time step by a factor of 5 for spatial discretization 2, and 10 for spatial discretization 3. In terms of CPU requirement, the method is usually twice as efficient.

The comparison of implicit time stepping with the other methods is difficult. Clearly, when very large time steps are taken, it is much more efficient than either semi-implicit or explicit methods. Much of this advantage is, however, lost when shorter time steps are used, although it remains competitive.

## 2. THE DISCRETIZED ICE-SHEET EQUATION AS AN ITERATED MAP

### 2.1. Scaling

We shall now consider the ice-sheet evolution equations in scaled form. We are still using the equation set (3) and (4). By choosing scales for the thickness  $[H]$ , the span  $[L]$ , the accumulation rate  $[a]$  and  $[C]$  for the quantity  $C = 2A(\rho g)^n / (n + 2)$  which satisfy the relationship

$$[a] = \frac{[C][H]^{2n+2}}{[L]^{n+1}},$$

we can consider the ice-sheet equation

$$\frac{\partial H}{\partial t} = \nabla \cdot (CH^{n+2} |\nabla H|^{n-1} \nabla H) + a \quad (10)$$

in dimensionless form. This scaling is that used in the shallow-ice approximation (e.g. Hutter, 1983) and is described in this volume by Hindmarsh (1996). The time scale is given by

$$[t] = \frac{[H]}{[a]}.$$

We will restrict consideration to one dimension and, in the following section, zero dimensions.

### 2.2. Theory: iterated maps of zero-dimensional models

If we make the very crude assumptions that  $H(x, t) = I(t)J(x)$ , it is easy to show that a scale or zero-dimensional representation of the ice-sheet evolution equation is the ordinary differential equation (ODE)

$$\frac{dI}{dt} = 1 - I^{2n+2}$$

where  $I(t)$  represents the ice-sheet thickness, where we have scaled out the accumulation rate and the rate factor,



We note here that the steady solution  $I = 1$  is easily shown to be stable. For the record, this equation does have analytical solutions for particular values of  $n$ , including 3, where the solution is given implicitly by

$$\frac{1}{4}(\operatorname{arctanh}(I(t)) + \operatorname{arctan}(I(t))) + \frac{\sqrt{2}}{16} \ln\left(\frac{I^2(t) + I(t)\sqrt{2} + 1}{I^2(t) - I(t)\sqrt{2} + 1}\right) + \frac{1}{8}\sqrt{2}(\operatorname{arctan}(I(t)\sqrt{2} + 1) - \operatorname{arctan}(I(t)\sqrt{2} - 1)) = t.$$

A statement of a range of possible integration schemes is

$$I^{k+1} = I^k + \left(1 - (\phi I^{k+1} + (1 - \phi)I^k)^{2n+1} \cdot (\omega I^{k+1} + (1 - \omega)I^k)\right) \Delta_t$$

where  $k$  is a counting index and the parameters  $\phi, \omega \in [0, 1]$  control the iterated map. This map can be written

$$I^{k+1} = I^k \frac{1 - (1 - \omega)D}{1 + \omega D} + \frac{\Delta_t}{1 + \omega D},$$

$$D = \Delta_t (\phi I^{k+1} + (1 - \phi)I^k)^{2n+1}.$$

This will be called “the marching scheme”. Note that schemes with  $\phi > 0$  involve the solution of a non-linear algebraic equation through the dependence of  $D$  on  $I^{k+1}$ . In these schemes, there are two kinds of iterations: the iteration with counting index  $k$ , corresponding to a time sequence, and another sequence with counting index  $\ell$

$$\{I^{k+1,\ell}\}_{\ell=1,\dots,m}$$

where the  $I^{k+1,\ell}$  represent (we hope), better and better approximations to  $I^{k+1}$ , computed from

$$I^{k+1,\ell+1} = I^k \frac{1 - (1 - \omega)D^\ell}{1 + \omega D^\ell} + \frac{1}{1 + \omega D^\ell}$$

with

$$D^\ell = \Delta_t (\phi I^{k+1,\ell} + (1 - \phi)I^k)^{2n+1}.$$

We shall call this the “non-linear iteration”. The computation of sequence is terminated when

$$\left| I^{k+1,m} - I^k \frac{1 - (1 - \omega)D^m}{1 + \omega D^m} - \frac{1}{1 + \omega D^m} \right| \leq \psi$$

where  $\psi \ll 1$  is the convergence criterion of the non-linear iteration. We shall be considering four iterated maps, some corresponding to typical ODE integration schemes, while others are not typically used for ODEs, but all have been used for partial differential equations (PDE) and are incorporated for the sake of comparison.

1. Explicit (forward Euler) schemes:  $\omega = 0, \phi = 0$ .

$$I^{k+1} = I^k(1 - D) + \Delta_t.$$

2. Implicit (backward Euler) non-iterated schemes  $\omega = 1, \phi = 0$ .

$$I^{k+1} = I^k \frac{1}{1 + D} + \frac{\Delta_t}{1 + D}.$$

3. Implicit Picard iterated schemes  $\omega = 1, \phi = 1$ . A residual,  $r$ , is computed, where

$$r^{k+1,\ell} = -(1 + D^\ell)I^{k+1,\ell} + I^k + \Delta_t$$

and used in the iteration as follows:

$$I^{k+1,\ell+1} = I^{k+1,\ell} - \frac{r^{k+1,\ell}}{1 + \omega D^\ell}.$$

4. Implicit Newton–Raphson iterated schemes  $\omega = 1, \phi = 1$ . The same residual is used but now a Taylor expansion is used to update the solution:

$$I^{k+1,\ell+1} = I^{k+1,\ell} - \left(\frac{dr^{k+1,\ell}}{dI^{k+1,\ell}}\right)^{-1} r^{k+1,\ell}.$$

We shall investigate these maps as a function of the time-step parameter  $\Delta_t$ . The results are displayed in Figure 4. The first two cases (a) and (b) are the solutions as  $k \rightarrow \infty$  as a function of the time step. The solution points were computed once transients had decayed. The analytical solution, respected for small time steps, is  $I = 1$ . The maps show that for too large time steps the

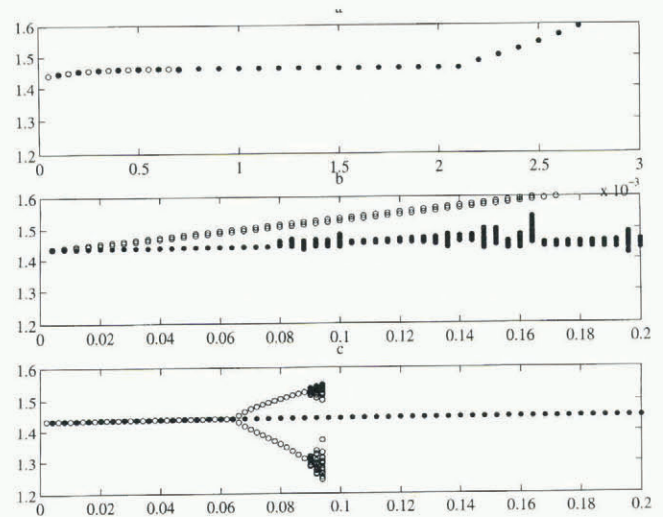


Fig. 5. Bifurcation maps for the iterated maps corresponding to the partial differential equations. Vertical axis is elevation at the divide, horizontal axis is  $\Delta_t$ . (a) is non-iterated schemes; an explicit scheme (circles) and an implicit non-iterated scheme (dots), (b) is Picard schemes without (circles) and with (dots) unstable manifold iteration, and (c) is Newton–Raphson schemes without (circles) and with (dots) unstable manifold iteration. The points represent successive values of  $H^k$  as  $k \rightarrow \infty$  (a) and successive values of  $H^{k,\ell}$  as  $\ell \rightarrow \infty$  (b and c). Note how the unstable manifold iteration substantially improves the stability properties of the iterated schemes, in particular the Picard iteration.



system does not necessarily “blow up” as do related linear equations

$$\frac{dI}{dt} = 1 - I,$$

but go through a period doubling sequence. This occurs at a larger time step for the implicit case than for the explicit case, but not at a very much greater time step. The explicit case blows up eventually (points not plotted). Cases (c) and (d) represent the iterate as  $\ell \rightarrow \infty$ . The Picard iteration has worse stability properties than the simple backward Euler scheme on which it is based, while the Newton–Raphson scheme looks very stable.

**The solution of partial differential equations as iterated maps**

By discretizing in time, we are making a fundamental change to the dynamics of the system under consideration, changing it from a dynamical system to an iterated map, i.e. a system of the form

$$\mathbf{u}^{k+1} = f(\mathbf{u}^k) + \mathbf{a}.$$

The issue of the stability of numerical schemes is based on establishing some kind of equivalence between the dynamics implied by the iterated map and that of the differential equation which is being investigated.

In solving partial differential equations, we typically arrive at a scheme

$$D_I(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^{k+1,\ell+1} = D_E(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^k + \mathbf{a}$$

where the first superscript represents the time step and the second the iterate; this equation is solved repeatedly for increasing  $\ell$  until the residual

$$\mathbf{r} = D_E(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^k + \mathbf{a} - D_I(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^{k+1,\ell}$$

is small in some suitable sense. Certain schemes are not iterated; these can be incorporated in this scheme by considering any residual to be sufficiently small. Certainly, if one maintains a time step short enough to preserve stability and accuracy, the residual will be small. In one dimension, the operators  $D_E, D_I$  are tri-diagonal matrices with rows given by

$$\begin{aligned} [D_{E,i-1,i} \ D_{E,i,i} \ D_{E,i+1,i}] &= \left[ \frac{(1-\omega)\Delta_t D_{i-1/2}^{k+\phi,\ell}}{\Delta_x^2} \ 1 \right. \\ &\left. - \frac{\Delta_t(1-\omega)(D_{i-1/2}^{k+\phi,\ell} + D_{i+1/2}^{k+\phi,\ell})}{\Delta_x^2} \ \frac{(1-\omega)\Delta_t D_{i+1/2}^{k+\phi,\ell}}{\Delta_x^2} \right] \end{aligned}$$

where  $D_{i-1/2}^{k+\phi,\ell}$  etc. are evaluated according to, for example, Equation (5) and the superscript refers to the time level  $k + \phi$  and iteration  $\ell$  of the vector  $H_i$ . In this case we may identify  $\mathbf{u}^{k+\phi,\ell}$  with  $H_i^{k+\phi,\ell}$ . Similarly,

$$\begin{aligned} [D_{I,i-1,i} \ D_{I,i,i} \ D_{I,i+1,i}] &= \left[ -\frac{\Delta_t \omega D_{i-1/2}^{k+\phi,\ell}}{\Delta_x^2} \ 1 \right. \\ &\left. + \frac{\Delta_t \omega (D_{i-1/2}^{k+\phi,\ell} + D_{i+1/2}^{k+\phi,\ell})}{\Delta_x^2} \ -\frac{\Delta_t \omega D_{i+1/2}^{k+\phi,\ell}}{\Delta_x^2} \right]. \end{aligned}$$

Zero margin-elevation boundary conditions simply require that the diagonal entry in the matrix be one.

Then, we can write down the following iterated maps:

1. Explicit schemes

$$\mathbf{u}^{k+1} = D_E(\mathbf{u}^k)\mathbf{u}^k + \mathbf{a}.$$

2. Implicit non-iterated schemes

$$D_I(\mathbf{u}^k)\mathbf{u}^{k+1} = D_E(\mathbf{u}^k)\mathbf{u}^k + \mathbf{a}.$$

3. Implicit Picard iterated schemes

$$D_I(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^{k+1,\ell+1} = D_E(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^k + \mathbf{a}$$

which can also be written

$$\mathbf{u}^{k+1,\ell+1} = D_I^{-1}(\mathbf{u}^{k+\phi,\ell})(D_E(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^k + \mathbf{a})$$

or

$$\begin{aligned} \mathbf{u}^{k+1,\ell+1} &= \mathbf{u}^{k+1,\ell} + D_I^{-1}(\mathbf{u}^{k+\phi,\ell})(D_E(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^k \\ &\quad + \mathbf{a} - D_I(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^{k+1,\ell}). \end{aligned}$$

4. Implicit Newton-Raphson iterated schemes

$$\begin{aligned} \mathbf{u}^{k+1,\ell+1} &= \mathbf{u}^{k+1,\ell} + J^{-1}(\mathbf{u}^{k+\phi,\ell})(D_E(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^k \\ &\quad + \mathbf{a} - D_I(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^{k+1,\ell}) \end{aligned}$$

where

$$J(\mathbf{u}^{k+\phi,\ell}) = \frac{\partial(D_E(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^k + \mathbf{a} - D_I(\mathbf{u}^{k+\phi,\ell})\mathbf{u}^{k+1,\ell})}{\partial\mathbf{u}^{k+1,\ell}}.$$

There is an important distinction between the first two cases, where the index in the iterated map is the time step, and the latter two cases, where the index in the iterated map is the iteration number. We are going to analyse these iterations as a function of  $\Delta_t$  in order to find the time steps where the map becomes unstable.

**2.3. Stability maps for some finite-difference schemes**

We consider some experiments in one dimension, symmetrical about the divide and with a Vialov–Nye fixed margin. The initial elevations are set to be 0.98 (an arbitrary number) times the elevations required for steady state for the discretization under consideration. The schemes are run for various time steps, iterating either as a marching scheme (explicit and semi-implicit schemes) or as non-linear iteration schemes (Picard or Newton–Raphson). For each time step and each scheme, a sequence in  $k$  (explicit and semi-implicit) or  $\ell$  was computed for 1024 steps, and the last 256 entries plotted.

Clearly, if the steady-state solution is reached or the iteration converges, all these points will be coincident. If the iteration fails to converge or the stepping is not stable, then for a given time step there will be several points, often reflecting decay on to a periodic or chaotic attractor. We emphasize that these are manifestations of



numerical instability and have no physical meaning. Linearized theory (Hindmarsh, unpublished) predicts exponential decay on to the analytical solution.

The results are plotted in Figure 5. Figure 5a shows the results from non-iterated schemes. The explicit scheme gives a maximum time step three times smaller than the maximum time step permissible for the non-iterated scheme, which is indicated by the break in slope. This occurs at time step of around  $2 \times 10^{-3}$ ; for somewhere like Greenland, where  $[t] = 10000$  a; this corresponds to a time step of around 20 years. The explicit scheme blows up for too large a time step.

If we turn now to the iterated schemes (Fig. 5b and c) the corresponding cases are for when there is “no unstable manifold correction”. This term will be explained in the next section. The Picard iteration is unstable for the whole range of the graph, while the Newton–Raphson iteration shows a stable time step up to around 0.07, corresponding to 700 years in Greenland and 7000 years in East Antarctica.

Note that, for both Picard iteration and Newton–Raphson iteration, the first manifestation of instability is a Hopf bifurcation to an oscillatory steady state with (implied) an unstable steady state.

**Application: the correction vector and sub-space under-relaxation**

It has been seen that during non-linear iterations, using both the Picard and Newton–Raphson methods, that if the time step is sufficiently large, a Hopf bifurcation occurs and a limit cycle emerges. Associated with this is the fact that if the time step is small enough (presumably relative to the spatial discretization in some unspecified way) then the iterative map “spirals” on to the steady solution. Further increases in the time step result in a typical Feigenbaum-like bifurcation sequence: period doubling, possibly a chaotic region, and, finally (not like a Feigenbaum sequence) blow up.

Detailed investigation not reported here of the iteration sequences shows that only one mode bifurcates as the time step is increased. The other modes remain stable, meaning that after a few (typically two or three iterations) the correction vector simply moves back and forth along the same line in correction space, with the implication that it is overshooting the unstable solution equivalent to zero error. If one assumes that the movement along this correction line is simply proportional to the error (in effect, linearizing around the solution), then it is possible to compute a correction vector of optimal size, which is moving along the correction line and should, if these assumptions are correct, reach the solution exactly. This procedure amounts to a numerical Taylor expansion of the iteration procedure projected on to the unstable manifold.

Of course there are problems: the decay of the iteration on to the correction line is not instantaneous, so one has to decide when to start this correction procedure; also, one infers the correction line from the correction vectors, so the direction will only be approximately correct. Nevertheless, the results are very promising: not only is convergence of the non-linear iteration accelerated but convergence occurs with this sub-space

method when the iteration would otherwise have diverged, because the divergence is along the correction line: the unstable manifold only has dimension one (at least in the region of the solution) while (locally) the remainder of the space is a stable manifold.

We shall work in correction space and not refer to residual space. Let us consider an iterative solution of some non-linear equation which generates a series of approximate solutions  $x^\ell, x^{\ell+1}, \dots$ , sequence being updated by a sequence of correction vectors  $c^\ell, c^{\ell+1}, \dots$ , such that  $x^{\ell+1} = x^\ell + c^\ell$ , and let the error  $e^\ell, e^{\ell+1}, \dots$  in the solution vector  $x^\ell, x^{\ell+1}, \dots$ , refer to that which exists before the correction is made. Our heuristic is simply that  $(c^\ell, c^{\ell+1}, \dots) = \alpha(e^\ell, e^{\ell+1}, \dots)$ . By the assumption that the decay is on to a straight line in the correction space, we can also simply state that

$$\begin{aligned} \|c^\ell\| &= \|e^{\ell+1} - e^\ell\| \Rightarrow \alpha \|c^\ell\| = \alpha \|e^{\ell+1} - e^\ell\| \Rightarrow \alpha \|c^\ell\| \\ &= \|c^{\ell+1} - c^\ell\|, \end{aligned}$$

whence we obtain

$$\alpha = \frac{\|c^{\ell+1} - c^\ell\|}{\|c^\ell\|}$$

and we compute our modified correction vector according to

$$c_*^{\ell+1} \equiv e^{\ell+1} \approx c^{\ell+1}/\alpha.$$

All these norms refer to the two norm. Clearly, this technique will only work if the iteration has converged on to the sub-space. The direction  $\theta$  between successive correction vectors is easily computed according to the normal projection rule

$$\theta = \arccos\left(\frac{c^{\ell+1} \cdot c^\ell}{\|c^{\ell+1}\| \|c^\ell\|}\right)$$

and the heuristic of the iteration is to decide when this angle is near enough 0 or  $\pi$  for the sub-space iteration to be viable. We have only looked at iterations which were overshooting (thus requiring under-relaxation), and we went as far as sub-space iterating when  $\theta \geq 5\pi/6$ . A typical iteration sequence, as reflected in the value of  $\theta$ , the sub-space angle, consists of this angle decaying on to a value greater than  $5\pi/6$  in two or three iterations, at which point the sub-space iteration is applied. Thereafter, the angle showed a certain amount of variation but seemed to have a value around  $\pi/2$ , which is what one would expect: the correct magnitude of  $\alpha$  is being correctly computed but the angle is wrong because of incomplete convergence on to the sub-space, resulting in the solution vector being wrong.

The results of using the unstable manifold correction are shown in Figure 5b and c. Picard iteration is stabilized to a time step of around 0.08, while Newton–Raphson iteration is stable to time steps well beyond 0.2. The unstable manifold correction permits significantly enhanced time steps for a small computational investment. When used with Picard iteration, the programming complexity is far less than that for Newton–Raphson iteration.



### 3. CONCLUSIONS

We have investigated the stability and accuracy of a number of different discretizations of the ice-sheet equation from an empirical and a theoretical point of view.

Our analysis of the 9 and 13 point molecule spatial discretizations indicate that both are first order and that they have comparable accuracies. It should be stressed that this finding is true only of the steady-state ice thickness with constant viscosity. We find that fully implicit time-stepping methods can increase the maximum stable time step very dramatically and that they also remain competitive when used with shorter time steps.

The study of a scaled ODE model of the ice-sheet equation can be used as a tool for understanding the mechanisms behind non-linear instability. For increasingly large time steps, this instability has the period-doubling property typical of a Feigenbaum bifurcation sequence, which eventually (not immediately) leads to numerical blow up. These findings prompted the development of a general technique for enhancing the stability of iterated maps using the unstable manifold correction.

### REFERENCES

- Budd, W.F. and D. Jenssen. 1975. Numerical modelling of glacier systems. *International Association of Hydrological Sciences Publication* 104 (Symposium at Moscow 1971—*Snow and Ice in Mountainous Areas*), 257–291.

- Carslaw, H.S. and J.C. Jaeger. 1983. *Conduction of heat in solids. Second edition*. Oxford, Clarendon Press.
- Ekman, T. 1987. *Numeriska metoder på dator och dosa*. Lund, Studentlitteratur.
- Hindmarsh, R.C.A. 1996. Stochastic perturbation of divide position. *Ann. Glaciol.*, **23** (see paper in this volume).
- Hindmarsh, R.C.A. and K. Hutter. 1988. Numerical fixed domain mapping solution of free-surface flows coupled with an evolving interior field. *Int. J. Numer. Anal. Methods Geomech.*, **12**(4), 437–459.
- Hindmarsh, R.C.A., L.W. Morland, G.S. Boulton and K. Hutter. 1987. The unsteady plane flow of ice-sheets: a parabolic problem with two moving boundaries. *Geophys. Astrophys. Fluid Dyn.*, **39**(3), 183–225.
- Hutter, K. 1983. *Theoretical glaciology; material science of ice and the mechanics of glaciers and ice sheets*. Dordrecht, etc., D. Reidel Publishing Co./Tokyo, Terra Publishing Co.
- Huybrechts, P. 1992. The Antarctic ice sheet and environmental change: a three-dimensional modelling study. *Ber. Polarforsch.* 99.
- Huybrechts, P., T. Payne and EISMINT Intercomparison Group. 1996. The EISMINT benchmarks for testing ice-sheet models. *Ann. Glaciol.*, **23** (see paper in this volume).
- Johnson, L.W. and R.D. Riess. 1982. *Numerical analysis*. Reading, MA, Addison Wesley.
- Mahaffy, M.W. 1976. A three-dimensional numerical model of ice sheets: tests on the Barnes Ice Cap, Northwest Territories. *J. Geophys. Res.*, **81**(6), 1059–1066.
- Nye, J.F. 1959. The motion of ice sheets and glaciers. *J. Glaciol.*, **3**(26), 493–507.
- Oerlemans, J. and C.J. van der Veen. 1984. *Ice sheets and climate*. Dordrecht, etc., D. Reidel Publishing Co.
- Press, W.H., B.P. Flannery, S.A. Teukolsky and W.T. Vetterling. 1992. *Numerical recipes in FORTRAN: the art of scientific computing. Second edition*. Cambridge, Cambridge University Press.
- Vialov, S.S. 1958. Regularities of glacial shields movement and the theory of plastic viscous [sic] flow. *International Association of Scientific Hydrology Publication* 47 (Symposium at Chamonix 1958—*Physics of the Movement of the Ice*), 266–275.