

# Theory of lattice Boltzmann simulations of glacier flow

DAVID B. BAHR

*Cooperative Institute for Research in Environmental Sciences, University of Colorado, Boulder, Colorado 80309, U.S.A.*

JOHN B. RUNDLE

*Cooperative Institute for Research in Environmental Sciences and Department of Geological Sciences, University of Colorado, Boulder, Colorado 80309, U.S.A.*

**ABSTRACT.** A lattice Boltzmann technique for modeling Navier–Stokes fluid flow is extended to allow steady-state simulations of glaciers and other slow-flowing solids. The technique is based on a statistical mechanical representation of flowing ice as a set of particles (populations) which translate and collide on a face-centered cubic lattice. The average trajectories of the populations give the velocities of the ice at any point in the glacier. The method has considerable advantages over other techniques, including its ability to handle complex realistic geometries without additional complications to the code. Examples are presented for two-dimensional simulations.

## INTRODUCTION

A common, if unstated, goal of many glaciological field programs is to collect surface-velocity and geometry information suitable for numerical inversions for conditions at the unobserved bed. Two classic examples from the recent past (among many) are Columbia and Variegated Glaciers with volumes of published information and numerous papers devoted to numerical inversions (e.g. Balise and Raymond, 1985; Meier and others, 1985; Raymond and others, 1987; Van der Veen and Whillans, 1993). In fact, during the last century, dozens of other glaciers and ice streams have also been repeatedly surveyed to gather information suitable for inversions aimed at elucidating sliding laws and mechanical controls at or near the bed. Typical numerical procedures have focused on finite-element and finite-difference formulations of the inversion problem, and with the advent of work stations, these methods have become increasingly accurate, especially in their descriptions of the deformational components of flow. However, the model's efficiency and accuracy in describing complex geometries have lagged far behind the more impressive developments in continuum flow. Finite-element techniques can take advantage of new methods for automatic generation of appropriate modeling meshes and finite-difference techniques can utilize specialized curvilinear coordinate systems. In both cases, however, the increased ability to deal with realistic geometries has forced a corresponding increase in computational and coding complexity.

Within the last few years a number of exciting advances in statistical mechanics have opened up new approaches to modeling low-viscosity fluid flows around complex geometric structures. One of the most recent and successful of these new techniques is the lattice gas automaton (LGA) which excels in its ability to handle complex boundary conditions with a surprisingly simple

code (Frisch and others, 1986). LGA models represent a fluid as a set of colliding gas particles which on average reproduce incompressible Navier–Stokes fluid flow. Boundary conditions are handled by simple particle-collision rules which reproduce the desired fluid velocities at any point. Using LGA, for example, flows through complex, even fractal, porous media geometries have been successfully modeled at a level of detail not previously possible (e.g. Chen and others, 1991). Likewise, simulations of flow around dissolving miscible solids and simulations of turbulent flow around arbitrary obstacles has become routine (e.g. d'Humières and Lallamand, 1987; Lawniczak and others, 1991).

These automaton techniques have been slow to excite interest in disciplines which focus on the slow-creeping flow of solids, like glaciers, primarily because the bulk of LGA research has developed models suitable for low viscosities and high Reynolds number simulations. These constraints, however, can be removed, and using LGA to simulate glaciers requires only a shift in modeling philosophy. Rather than focusing on finite-difference or finite-element models of the macroscopic equations of state, LGAs focus on building much simpler models formulated at a microscopic level. The microscopic models are constructed to conserve mass, momentum and energy but do not attempt completely faithful reproductions of the microscopic world. This is a significant departure from traditional modeling philosophy but, although inexact, this type of simulation is successful because hydrodynamics and thermodynamics describe large-scale systems which behave independently of their precise microscopic formulation (Salem and Wolfram, 1986). So, while the macroscopic models are typically limited to a small number of spatial "cells" where information on the flow state is updated based on the complex continuum equations, these microscopic models instead use tens of thousands of cells which are updated by simple rules



chosen for their computational efficiency. Proper continuum behaviour is reconstructed from large-scale averages of the many microscopic cells. In other words, very complex systems are studied by looking at simplified microscopic models which reproduce the correct mathematical behaviour at macroscopic scales.

The shift from a macroscopic to a simplified microscopic paradigm will allow significant advances in our ability to model the flow of glaciers, ice sheets and other solids around complex realistic obstacles. In the following sections, therefore, we describe a variant of LGA, called the lattice Boltzmann method, and demonstrate a new technique for implementing constitutive relationships appropriate for ice and other non-Newtonian slow-flowing solids. The lattice Boltzmann method described here is appropriate for isothermal steady-state problems or for time-varying velocities with fixed glacier geometries. The technique can be used to invert surface velocities for information at the bed or to extract velocity and stress profiles from complicated, realistic glacier channels. This approach represents the first step towards a more general lattice Boltzmann model of time-dependent and thermally inhomogeneous glacier flow.

**THE LATTICE BOLTZMANN METHOD**

Historically, the lattice Boltzmann method evolved from LGA and is most easily understood in the context of its automaton predecessors. Automata correspond closely with von Neumann’s original concept of a computer and are any machine with a finite set of input and output states and a fixed finite set of rules which map each input to an output (Lewis and Papadimitriou, 1981, p. 222). A simple example uses finite-length binary strings as input and output states; a look-up table converts a binary input string to a binary output string. The set of rules, or the look-up table, is the automaton’s program.

Lattice gas automata have input and output states which correspond to particle positions and velocities on a triangular lattice. At each site on the lattice, up to six particles of unit mass are assigned unit velocities  $\mathbf{e}_i$  in one of the six lattice directions ( $i = 1, \dots, 6$ ) (Fig. 1). At each point on the lattice there is at most one particle with a given velocity (in other words, no two particles are allowed to have the same momentum at the same lattice site). If  $n_i = 1$  and  $n_i = 0$  represent occupied and unoccupied momentum states, then the particle positions and velocities at lattice site  $\mathbf{x}$  and time  $t$  are given by the binary string  $n(\mathbf{x}, t) = (n_1 n_2 n_3 n_4 n_5 n_6)$ . Figure 1 illustrates  $n(\mathbf{x}, t) = (100001)$ .

At each time step in the model’s evolution, particles in the LGA are translated one lattice unit in the direction of their velocity. Particles which arrive at the same site are said to collide and are redistributed according to a fixed set of rules which conserve mass and momentum (these are the automaton rules which convert the binary input to the binary output states). After colliding, the particles are translated again, and the process repeats. Figure 2 illustrates the translation and collision rules for two and three particles.

This basic lattice gas automaton was first proposed by Frisch and others (1986) in a paper in which they

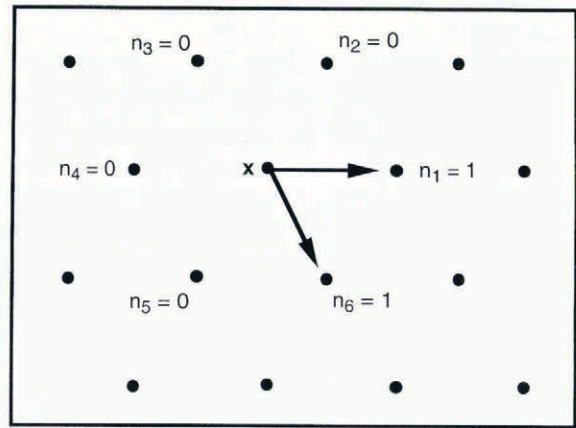


Fig. 1. Momentum state  $n(\mathbf{x}, t) = (100001)$  on a triangular lattice.

demonstrated that the Navier–Stokes equation is reproduced in the continuum limit by the movement of the particles on the triangular lattice. If the local density  $\rho$  and local velocity  $\mathbf{v}$  of the fluid at any point in time are given by

$$\rho(\mathbf{x}, t) = \sum_{i=1}^6 n_i(\mathbf{x}, t) \tag{1}$$

and

$$\rho(\mathbf{x}, t)\mathbf{v}(\mathbf{x}, t) = \sum_{i=1}^6 n_i(\mathbf{x}, t)\mathbf{e}_i, \tag{2}$$

then they showed that macroscopic quantities like the fluid velocity are obtained by spatial averages of  $\mathbf{v}$ . Boundary conditions are implemented by adjusting the directions of any particles which collide with a boundary; and at each time step, body forces, like gravity, can be specified by altering the direction (momentum) of random particles to align with the force (Kadanoff and others, 1987). A number of review papers have given details of the derivations which lead from the collision rules to the full Navier–Stokes equations of fluid flow (e.g. Wolfram, 1986; Frisch and others, 1987).

LGA models of fluid flow are stable because the look-up tables are exact Boolean operators (round-off errors are eliminated). The translation and collision process is also local in nature, so that the updates can be performed simultaneously at all the nodes, making LGA well suited for fast parallel processing. The discrete nature of the particle momenta, however, adds noise to the system, requiring large spatial averages of the particle trajectories to get reasonable fluid approximations. This noise clearly outweighed the advantages of an LGA implementation of the glacier-flow problem constructed by the authors, and along with a problem in adjusting the viscosity (controlled by the selection of specific collision rules) motivated others to develop the original lattice Boltzmann technique (Higuera and Jiménez, 1989).

The lattice Boltzmann method considers populations of particles ( $0 \leq n(\mathbf{x}, t) \leq 1$ ) rather than individual particles on the lattice. In essence, this is the same as simultaneously running a large ensemble of LGA simulations and then determining the probability of a particle having a particular momentum state (i.e. probability of occupying



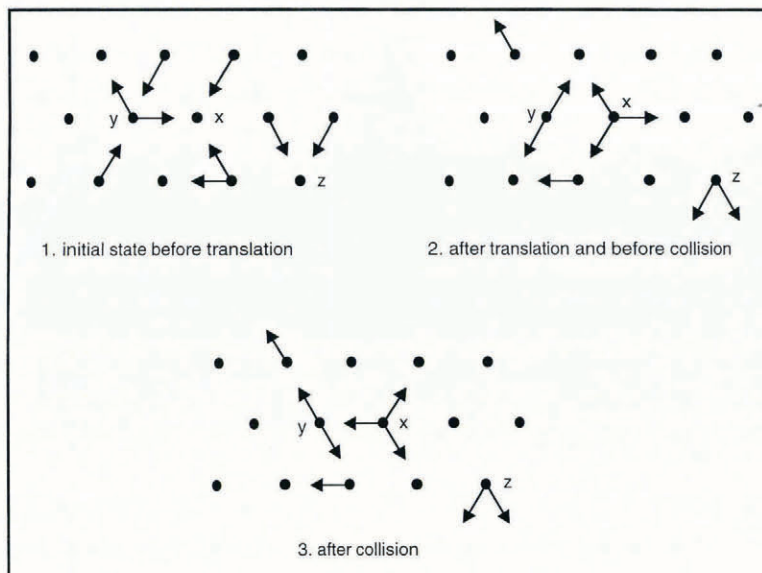


Fig. 2. An example of the translation and collision process for LGA particles. The three-way collision at site  $\mathbf{x}$  re-arranges velocities in the only manner which will conserve momentum and mass. The two particles colliding collinearly at site  $\mathbf{y}$  have zero total momentum and can be redistributed in any other collinear directions (the choice can be made randomly). All other possible collision configurations are ignored and the velocities are left unchanged (for example, grid site  $\mathbf{z}$ ).

a particular lattice site with a particular velocity). (In statistical mechanics, this collection of LGA simulations would be referred to as the canonical ensemble which has a Boltzmann distribution of possible states, hence the name.) Practically speaking, the lattice Boltzmann technique averages out the noise in the system while maintaining the fast parallel architecture. However, the arithmetic is now real, rather than Boolean, and the ultimate stability of the LGA is compromised, though stable behavior is still possible for almost all flow scenarios of interest. (Benzi and others (1992) discussed details of stability to numerical dispersion, discretization and other types of error.)

The lattice Boltzmann collision rules can be implemented in the same manner as the LGA rules, although a more efficient method is discussed here. In the original LGA, there were six unit velocities  $\mathbf{e}_i$  ( $i = 1, \dots, 6$ ) on a triangular lattice, but more recent analyses (Skordos, 1993) have shown that the Navier–Stokes equations can be reproduced with two velocities on a more easily coded square lattice (velocities along edges and diagonals). In particular, for  $i = 1, \dots, 8$

$$\mathbf{e}_i = \begin{cases} \left( \cos \frac{i-1}{4} \pi, \sin \frac{i-1}{4} \pi \right) & \text{for } i \bmod 2 = 1 \\ \sqrt{2} \left( \cos \frac{i-1}{4} \pi, \sin \frac{i-1}{4} \pi \right) & \text{for } i \bmod 2 = 0. \end{cases} \quad (3)$$

In addition,  $\mathbf{e}_0 = (0, 0)$  corresponds to a “rest” particle with zero velocity. Also, rather than using look-up tables, the collision step can be more compactly represented as a relaxation from an equilibrium state. Let  $0 \leq N_i(\mathbf{x}, t) \leq 1$  be the probability of finding a particle at node  $\mathbf{x}$  and time  $t$  with unit velocity  $\mathbf{e}_i$ , and let  $N_i^{\text{eq}}(\mathbf{x}, t)$  be the equilibrium distributions. Then, for a unit time step ( $\Delta t = 1$ ),

$$N_i(\mathbf{x} + \mathbf{e}_i \Delta t, t + 1) - N_i(\mathbf{x}, t) = -\frac{1}{\tau} (N_i(\mathbf{x}, t) - N_i^{\text{eq}}(\mathbf{x}, t)). \quad (4)$$

The lefthand side of Equation (4) is just the rate of change of the particle distributions due to collisions.  $\tau$  is a unitless relaxation parameter which controls this change and adjusts the rate at which the equilibrium is approached.

The equilibrium states  $N_i^{\text{eq}}$  are derived from a Chapman–Enskog expansion of Equations (1), (2) and (4) (Skordos, 1993) and for a square lattice are given by

$$N_0^{\text{eq}} = \frac{4}{9} \rho \left( 1 - \frac{3}{2} \mathbf{v} \cdot \mathbf{v} \right)$$

$$N_i^{\text{eq}} = \frac{1}{9} \rho \left( 1 + 3(\mathbf{e}_i \cdot \mathbf{v}) + \frac{9}{2} (\mathbf{e}_i \cdot \mathbf{v})^2 - \frac{3}{2} \mathbf{v} \cdot \mathbf{v} \right)$$

for  $i \bmod 2 = 1$

$$N_i^{\text{eq}} = \frac{1}{36} \rho \left( 1 + 3(\mathbf{e}_i \cdot \mathbf{v}) + \frac{9}{2} (\mathbf{e}_i \cdot \mathbf{v})^2 - \frac{3}{2} \mathbf{v} \cdot \mathbf{v} \right)$$

for  $i \bmod 2 = 0. \quad (5)$

The density  $\rho$  and macroscopic velocity  $\mathbf{v}$  are given by Equations (1) and (2) without averaging.

The relaxation time parameter is also derived from the Chapman–Enskog expansion and is related to the kinematic viscosity  $\nu$  (measured in lattice units squared per time step) by

$$\nu = \frac{(2\tau - 1)}{6}. \quad (6)$$

$\tau$  has a lower bound of  $\frac{1}{2}$  to prevent unphysical negative viscosities. Intuitively, the relationship between  $\tau$  and  $\nu$  is a consequence of the relaxation parameter’s influence on the rate of change of particle distributions at each collision step. Thinking of the Boltzmann method as an ensemble of LGAs, a larger  $\tau$  decreases the change in  $N$  with time (Equation (4)) or, in other words, decreases the probability of a collision in the ensemble of LGAs. When collisions are less probable, the mean free path of a particle is increased and Maxwell (1859) demonstrated that the mean free path



of particles in a gas is proportional to the viscosity (Huang, 1987). Therefore, increasing  $\tau$  increases the mean free path and increases the viscosity.

**NON-NEWTONIAN FLOW**

The ability to adjust the viscosity with the relaxation parameter is our basis for constructing non-Newtonian constitutive relationships suitable for modeling glacier flow. When  $\tau$  has a constant value, the viscosity is fixed and stress is linearly related to strain. In other words  $\sigma'_{ij} = 2\mu\dot{\epsilon}_{ij}$  where  $\sigma'_{ij}$  and  $\dot{\epsilon}_{ij}$  are components of the deviatoric stress and strain-rate tensors and  $\mu = \rho\nu$  is the dynamic viscosity (e.g. Mase, 1970). Water is the classic example of such a linear or Newtonian fluid. Ice, magma, metals and many other solids are characterized, however, by an effective viscosity,  $\mu_{\text{eff}}$ , which is a function of the strain rate (in addition to other dependencies such as chemical impurities and temperature). In fact, for ice, magma and most metals, the effective viscosity is a power of the second strain-rate invariant,

$$\dot{\epsilon} = \sqrt{\sum_{i,j} \dot{\epsilon}_{ij} \dot{\epsilon}_{ij}}$$

(Turcotte and Schubert, 1982; Paterson, 1994), i.e.

$$\sigma'_{ij} = \mu_{\text{eff}} \dot{\epsilon}_{ij} \tag{7}$$

where

$$\mu_{\text{eff}} \propto \dot{\epsilon}^{n-1} \tag{8}$$

for some positive  $n$ .

To handle these generalized viscosities,  $\tau$  is simply expressed as a function of the effective viscosity,

$$\tau = \frac{6(\mu_{\text{eff}}/\rho) + 1}{2} \tag{9}$$

Now the strain rates at each lattice site can be estimated by finite differences of the velocities at their neighbors. This determines the effective dynamic viscosity  $\mu_{\text{eff}}$  and an appropriate  $\tau$  at each node on the lattice. Aharonov and Rothman (1993) have developed a similar but more complicated scheme for non-Newtonian flow based on an earlier implementation of the lattice Boltzmann method. Our results compare favorably with their work and require less complicated coding.

**ELIMINATING COMPRESSIBILITY AND INERTIAL EFFECTS**

Solids typically have kinematic viscosities on the order of  $1 \times 10^{15} \text{ m}^2 \text{ s}^{-1}$ , about 21 orders of magnitude higher than the typical viscosities for a fluid ( $1 \times 10^{-6} \text{ m}^2 \text{ s}^{-1}$ ). For a lattice Boltzmann model, this has two important consequences for flow simulations of a solid. First, the mean free path of a particle is so long that infeasibly large model grids are necessary for realistic simulations. This can only be countered by taking advantage of dynamic similarity and using dynamically equivalent flow scenarios with much smaller viscosities. Secondly, large viscosities imply negligibly small inertial terms (i.e.

for Newtonian flow, the  $\mathbf{v} \cdot \nabla \mathbf{v}$  term in the Navier–Stokes equation should be set to zero). However, these two consequences tend to work against each other; decreasing the viscosity causes increasingly dominant inertial effects in the model, leading to high Reynolds numbers and turbulent flow. To prevent this behavior, it is best to eliminate the inertial terms from the glacier-model formulation.

To do this, consider the macroscopic flow behavior of the lattice Boltzmann model as derived using the Chapman–Enskog procedure, which is essentially a Taylor expansion of Equation (4). The expanded  $N_i$  are rewritten in terms of  $\rho$  and  $\rho\mathbf{v}$  using the mass and momentum relationships given in Equations (1) and (2). The equilibrium distribution is treated as an expansion about small velocities (Hou, 1995),

$$N_i^{\text{eq}} = A_i + B_i(\mathbf{e}_i \cdot \mathbf{v}) + C_i(\mathbf{e}_i \cdot \mathbf{v})^2 - D_i \mathbf{v} \cdot \mathbf{v} \tag{10}$$

with higher-order terms neglected. The general parameters  $A_i, B_i, C_i$  and  $D_i$  are then selected to ensure that to first order the expansion gives the Navier–Stokes equations (see Equation (5)). In our case, we choose a different set of parameters which gives the same solution but eliminates the inertial terms.  $A_i$  and  $B_i$  remain the same as before, but  $C_i = D_i = 0$ . Intuitively, the terms in Equation (10) which are non-linear in  $\mathbf{v}$  have been set to zero because the Navier–Stokes pdes are linear in  $\mathbf{v}$  except for the inertial term,  $\mathbf{v} \cdot \nabla \mathbf{v}$ .

In our model, one additional correction is made to ensure incompressibility. Without some further manipulation, the equilibrium distributions given in Equations (5) and (10) predict that  $\nabla \cdot (\rho\mathbf{u}) = 0$  for steady-state flows with velocity  $\mathbf{u}$ . Truly incompressible flows require  $\nabla \cdot \mathbf{u} = 0$ . Therefore, Hou (1995) has suggested redefining the velocity to be

$$\mathbf{v} = \rho\mathbf{u} = \sum_{i=1}^8 N_i(\mathbf{x}, t)\mathbf{e}_i \tag{11}$$

Using the same Chapman–Enskog expansion as outlined above, this leads to a correction in the equilibrium distributions given by

$$\begin{aligned} N_0^{\text{eq}} &= \frac{4}{9}\rho \\ N_i^{\text{eq}} &= \frac{1}{9}(\rho + 3(\mathbf{e}_i \cdot \mathbf{v})) \quad \text{for } i \text{ mod } 2 = 1 \\ N_i^{\text{eq}} &= \frac{1}{36}(\rho + 3(\mathbf{e}_i \cdot \mathbf{v})) \quad \text{for } i \text{ mod } 2 = 0. \end{aligned} \tag{12}$$

A more complete motivation of the incompressibility corrections has been explained in Hou (1995).

**GENERALIZATION TO THREE DIMENSIONS**

A three-dimensional generalization of the lattice Boltzmann model is straightforward. Instead of a square lattice, the particle populations move on a face-centered cubic lattice with six nearest neighbors at a distance of  $|\mathbf{e}_i \Delta t| = 1$  lattice unit and eight next-nearest at a distance of  $|\mathbf{e}_i \Delta t| = \sqrt{3}$ . All other equations are unchanged except



for the equilibrium distributions which become

$$\begin{aligned}
 N_0^{\text{eq}} &= \frac{2}{9}\rho \\
 N_i^{\text{eq}} &= \frac{1}{9}(\rho + 3(\mathbf{e}_i \cdot \mathbf{v})) \quad \text{for } |\mathbf{e}_i \Delta t| = 1 \\
 N_i^{\text{eq}} &= \frac{1}{72}(\rho + 3(\mathbf{e}_i \cdot \mathbf{v})) \quad \text{for } |\mathbf{e}_i \Delta t| = \sqrt{3} \quad (13)
 \end{aligned}$$

for  $i = 0$  to 14.

**SCALING ANALYSIS AND NON-DIMENSIONALIZATION**

In order to run the lattice Boltzmann model at low viscosities and have dynamically similar behavior to a highly viscous glacier, a set of non-dimensional numbers needs to be derived from the macroscopic equations of glacier flow. To guarantee similar dynamics, these non-dimensional numbers must be the same for both the high- and low-viscosity scenarios. For Newtonian flows, the appropriate non-dimensional numbers are derived by dividing each term of the Navier–Stokes equation by one of the other terms (Welty and others, 1984). Reynolds number, for instance, is the ratio of the inertial term to the viscous term. However, the equations of glacier flow are complicated by a non-linear rheology (Equation (7)) and cannot be written as a single partial differential equation. In that case, ratios of terms are impossible and a more sophisticated scaling analysis is necessary to derive the non-dimensional parameters (Logan, 1987).

Scaling analyses start with a set of relevant variables  $\{V_1, V_2, \dots, V_i\}$  and replace each by a rescaled quantity  $\{\lambda^{m_1} \bar{V}_1, \lambda^{m_2} \bar{V}_2, \dots, \lambda^{m_i} \bar{V}_i\}$ . The rescaled variables are substituted into the relevant equations (describing glacier flow in our case) and the scaling parameter  $\lambda$  is factored out. This factoring ensures that the equations are unchanged by the rescaling and is used to derive a set of constraints on  $\{m_1, m_2, \dots, m_i\}$ . These constraints determine how each variable must be related to its rescaled quantity and therefore determine how different scale simulations must be related to each other in order for the relevant equations to describe the same physical process. In other words, the constraints provide the relationships necessary to construct non-dimensional quantities.

Consider the equations describing isothermal steady-state glacier flow in two dimensions. From the definition of strain rates,

$$\dot{\epsilon}_{xx} = \frac{\partial u}{\partial x}, \quad \dot{\epsilon}_{zz} = \frac{\partial v}{\partial z} \quad \text{and} \quad \dot{\epsilon}_{xz} = \frac{1}{2} \left( \frac{\partial u}{\partial z} + \frac{\partial v}{\partial x} \right)$$

where  $x$  and  $z$  are along coordinate directions perpendicular and parallel to gravity, and  $u$  and  $v$  are deformational velocity components in the  $x$  and  $z$  directions. From these definitions,

$$\frac{\partial^2 \dot{\epsilon}_{xx}}{\partial z^2} + \frac{\partial^2 \dot{\epsilon}_{zz}}{\partial x^2} - 2 \frac{\partial^2 \dot{\epsilon}_{xz}}{\partial x \partial z} = 0 \quad (14)$$

which is a compatibility requirement ensuring that the velocities are unique (e.g. Mase, 1970, p.92). The strain rates are related to deviatoric stresses by Glen’s flow law

for ice (Equation (7)), which can be written explicitly as

$$\dot{\epsilon}_{xx} = A \left( \sigma'_{xx}{}^2 + \sigma'_{xz}{}^2 \right)^{\frac{n-1}{2}} \sigma'_{xx} \quad (15)$$

and

$$\dot{\epsilon}_{xz} = A \left( \sigma'_{xx}{}^2 + \sigma'_{xz}{}^2 \right)^{\frac{n-1}{2}} \sigma'_{xz} \quad (16)$$

with  $\sigma'_{xz} = \sigma_{xz}$  and  $\sigma'_{xx} = (\frac{1}{2})(\sigma_{xx} - \sigma_{zz})$ . ( $\sigma_{ij}$  is a stress-tensor component.) Conservation of momentum requires

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xz}}{\partial z} = -\rho g \sin \alpha \quad (17)$$

and

$$\frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{zz}}{\partial z} = -\rho g \cos \alpha \quad (18)$$

where  $g$  is gravity and  $\alpha$  is the angle of the ice surface relative to the horizontal.

The variables in these equations can be rescaled by the following substitutions:  $\bar{x} = \lambda^a x$ ,  $\bar{z} = \lambda^b z$ ,  $\bar{\sigma}_{xx} = \lambda^c \sigma_{xx}$ ,  $\bar{\sigma}_{xz} = \lambda^d \sigma_{xz}$ ,  $\bar{\sigma}_{zz} = \lambda^e \sigma_{zz}$ ,  $\bar{A} = \lambda^f A$ ,  $\bar{\rho} = \lambda^j \rho$ ,  $\bar{g} = \lambda^k g$ ,  $\bar{u} = \lambda^p u$  and  $\bar{v} = \lambda^q v$ , where the overbar represents a scaled quantity and  $\lambda$  is an arbitrary scaling factor. After substitution, the rescaled equations are consistent with the original equations only if the  $\lambda$ s can be factored out. This places a set of constraints on the scaling exponents. For example, using the chain rule,  $(\partial \bar{\sigma}_{xx})/(\partial \bar{x}) = \lambda^{c-a} (\partial \sigma_{xx})/(\partial x)$  and  $(\partial \bar{\sigma}_{xz})/(\partial \bar{z}) = \lambda^{d-b} (\partial \sigma_{xz})/(\partial z)$ , so

$$\lambda^{c-a} \frac{\partial \sigma_{xx}}{\partial x} + \lambda^{d-b} \frac{\partial \sigma_{xz}}{\partial z} = -\lambda^{j+k} \rho g \sin \alpha. \quad (19)$$

This gives back the original Equation (17) if and only if  $\lambda^{c-a} = \lambda^{d-b} = \lambda^{j+k}$  or  $c - a = d - b = j + k$ .

Similar constraints derived from each of the other flow equations leads to a set of relationships which can be solved to give  $a = b$ ,  $c = d = e = a + j + k$  and  $p = q = a + i + n(a + j + k)$ . Using  $\lambda^a = \bar{x}/x$ , etc., these solutions imply that

$$\left( \frac{\bar{x}}{x} \right) = \left( \frac{\bar{z}}{z} \right) \quad (20)$$

$$\left( \frac{\bar{\sigma}_{xx}}{\sigma_{xx}} \right) = \left( \frac{\bar{\sigma}_{xz}}{\sigma_{xz}} \right) = \left( \frac{\bar{\sigma}_{zz}}{\sigma_{zz}} \right) = \left( \frac{\bar{x}}{x} \right) \left( \frac{\bar{\rho}}{\rho} \right) \left( \frac{\bar{g}}{g} \right) \quad (21)$$

and

$$\left( \frac{\bar{u}}{u} \right) = \left( \frac{\bar{v}}{v} \right) = \left( \frac{\bar{x}}{x} \right)^{n+1} \left( \frac{\bar{A}}{A} \right) \left( \frac{\bar{\rho}}{\rho} \right)^n \left( \frac{\bar{g}}{g} \right)^n. \quad (22)$$

Equations (20) through (22) give constraints that will be satisfied when two simulations are dynamically similar.

By solving for the original unscaled variables, Equations (20) through (22) can be rewritten as non-dimensional numbers,

$$\pi_1 = \frac{x^{n+1} A \rho^n g^n}{u}, \quad (23)$$

$$\pi_2 = \frac{x \rho g}{\sigma_{xz}} \quad (24)$$

and

$$\pi_3 = \frac{x}{z}. \quad (25)$$

Each of the variables now represents characteristic values (e.g. the glacier thickness for  $x$ ). Note that the characteristic shear stress is given by the basal shear, or



$\rho g x \sin \alpha$ , so the non-dimensional parameter  $\pi_2$  just requires the slopes to be the same in dynamically similar simulations.  $\pi_3$  requires geometric similarity and  $\pi_1$  is a generalization of Reynolds number divided by Froude's from Newtonian flow ( $n = 1$ ). Note that the inclusion of a third dimension in the glacier-flow equations does not alter  $\pi_1$  or  $\pi_2$  and leads to the obvious generalizations of geometric similarity. The axes orientations are also irrelevant to the derivation.

As a typical example, fix  $n = 3$  and let the glacier thickness be  $x = 100$  m, let the characteristic velocity be  $u = 1/100\,000$  m s<sup>-1</sup>,  $g = 9.8$  m s<sup>-2</sup>,  $\rho = 900$  kg m<sup>-3</sup> and  $A = 1 \times 10^{-24}$  m<sup>3</sup> s<sup>5</sup> kg<sup>-3</sup>. Then  $\pi_1 = 6.86$ . A dynamically similar model could have the values  $x = 30$ ,  $u = 0.1$ ,  $g = 4.73 \times 10^{-4}$ ,  $\rho = 2$  and  $A = 1000$  where each variable is measured in model units. These values are typical for the lattice Boltzmann model and note that the viscosity is kept relatively small, as required.

### TWO-DIMENSIONAL GLACIER-FLOW SIMULATIONS

A glacier-flow simulation starts by occupying each lattice site and lattice direction with some initial population of particles. These populations are then translated in the direction of their velocities. Populations arriving at the same lattice site are redistributed by the rules which conserve mass and momentum (Equations (4) and (12)). As in the lattice gas models, gravity is implemented by changing the momentum of particle populations by some small increment at each lattice site at each time step. The translations, redistributions and body-force momentum adjustments iterate until the system reaches a steady state. In our simulations, a change in velocity (in a least-squares sense) of less than  $1 \times 10^{-5}$  between iterations is required for steady state. Velocities are calculated from Equation (11).

Equation (12) gives the basis for specifying velocity-boundary conditions (Skordos, 1993). Rather than calculating the velocity from Equation (11), the velocity at any boundary site is fixed to the desired value (for example, to that of a surveyed velocity). At each iteration, the specified velocity is used to calculate an equilibrium distribution via Equation (12). The specified equilibrium distribution is then used to adjust the populations in the collision step (Equation (4)). Specifying zero velocity, for example, leads to a no-slip condition. A free-slip (or frictionless) condition is also possible by adjusting Equation (4) so that the angle of incidence equals the angle of reflectance for any populations approaching the free-slip boundary. Note that both specified velocities and free-slip conditions can be used at different sites in the same simulation.

Two examples of this simulation process are presented below. An implementation of the three-dimensional model is in progress and examples will be presented in future publications.

#### Laminar flow

The simplest possible glacier is a uniform thickness of ice sitting on an inclined plane. Assuming a no-slip condition at the bed and a free-slip condition at the surface, the

velocity profile is given by the laminar-flow solution,  $v = 0$  and  $u(z) = u_s - 2A(\rho g \sin \alpha)^n z^{n+1}/(n + 1)$  where  $u_s$  is the surface velocity and is  $z$  measured perpendicular to the surface, positive downwards (Paterson, 1994, p. 251). For  $n = 1$ ,  $A = 3$ ,  $\rho = 1.0$ ,  $g = 2.8 \times 10^{-5}$ ,  $\alpha = 0.46$  rad and a bed at  $z = 30$  this gives  $u(z) = 0.034 - 3.7 \times 10^{-5} z^2$ . For this case,  $\pi_1 = 2.22$ , so one choice of corresponding physical values is  $A = 1 \times 10^{-19}$  m s kg<sup>-1</sup>,  $\rho = 900$  kg m<sup>-3</sup>,  $g = 9.8$  m s<sup>-2</sup>, a depth of 1000 m and a slow surface velocity of  $0.0125$  m a<sup>-1</sup>. Figure 3 compares the results of the Boltzmann model to the analytical expression. A similar simulation with  $n = 3$ ,  $g = 1.2 \times 10^{-3}$  and  $A = 1000$  illustrates the characteristic flattening of the velocity profile caused by a strain-thinning solid (Fig. 3). In this case,  $\pi_1 = 22.86$ , so one choice of corresponding physical values is  $A = 1 \times 10^{-24}$  m<sup>3</sup> s<sup>5</sup> kg<sup>-3</sup>,  $\rho = 900$  kg m<sup>-3</sup>,  $g = 9.8$  m s<sup>-2</sup>, a depth of 100 m and a surface velocity of  $94.66$  m a<sup>-1</sup>.

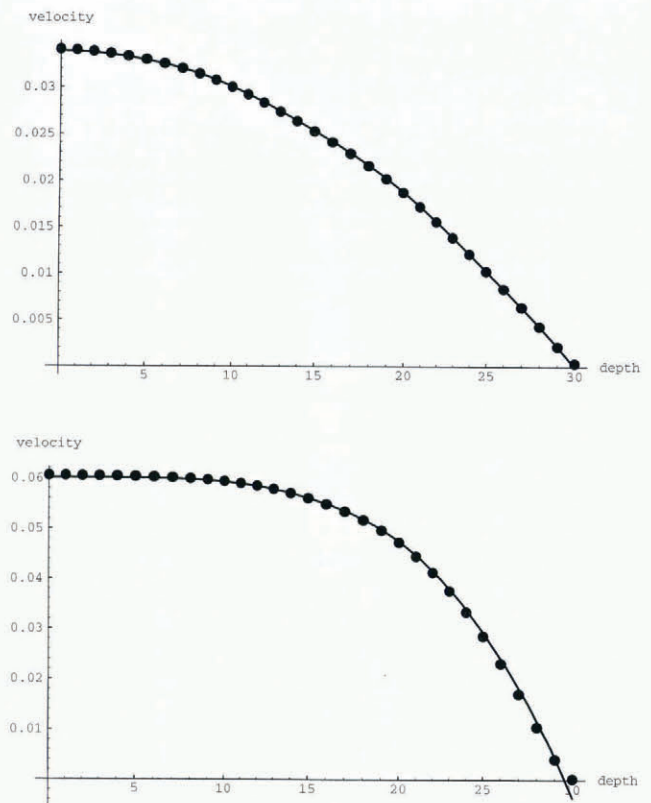


Fig. 3. Laminar flow with  $n = 1$  (top) and  $n = 3$  (bottom). Point data are from the lattice Boltzmann simulations and solid lines are the analytical solutions.

#### Valley-glacier flow

Figure 4 shows non-Newtonian ( $n = 3$ ) flow through a more typical two-dimensional cross-section of a valley glacier. A no-slip condition is specified at the bed and surface velocities are fixed at the surface (roughly downwards in an accumulation zone and roughly upwards in an ablation zone). Other parameters (and potential choices for characteristic physical values) are the same as above. Notice that the bed is undulating and overdeepened near the terminus and that the cross-section has an icefall and a steep bedrock cliff near the equilibrium line. These geometries can cause significant meshing difficulties for



non-Newtonian finite-difference and finite-element techniques but pose no difficulty for the lattice Boltzmann model.

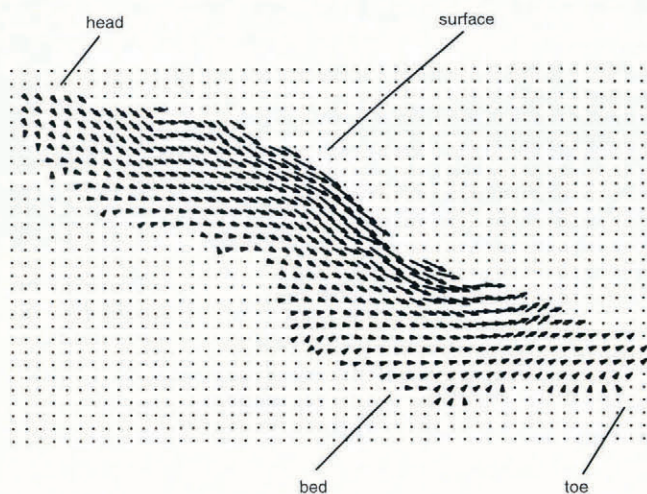


Fig. 4. Velocity vectors calculated by the lattice Boltzmann model for a two-dimensional vertical cross-section through an irregularly shaped valley glacier. Gravity is acting vertically downwards. The small dots are lattice sites which are not a part of the glacier and have zero velocity. The vectors are magnified 100 times and are measured in lattice units per time step. The grid is 50 by 30 lattice units.

## CONCLUSIONS

The code for a lattice Boltzmann model is simple and short, with an intuitive motivation — the translation and collision of particles. Glacier-flow geometries can be as complicated as necessary with no additional complexity imposed on the model code. The same technique can be used to model the slow creeping flow of any solid, such as glass, cold tar, salt diapers or lava. The only restrictions are a quantifiable constitutive relationship (substituted in place of Equation (8)), flows which have negligible inertia, steady-state behavior and isothermal conditions.

The preliminary success of the lattice Boltzmann method for steady-state isothermal glacier flows suggests that it would be profitable to focus additional research on fully time-dependent three-dimensional thermomechanical Boltzmann techniques. The current model can already handle time-varying boundary conditions, so the primary difficulty is allowing free-surface geometries which can evolve with time. The authors are currently developing this free-surface condition as well as techniques for incorporating the thermodynamic equations of state.

## ACKNOWLEDGEMENTS

D. Bahr was supported by a Visiting Fellowship and J. Rundle was supported under U.S. Department of Energy grant number DE-FG03-95ER14499, both to the

Cooperative Institute for Research in Environmental Sciences at the University of Colorado. We appreciate S. Peckham's assistance in the scaling analysis and would like to thank G. Doolen for bringing S. Hou's dissertation to our attention. S. Marshall's review of the mathematics improved the clarity of this paper.

## REFERENCES

- Aharonov, E. and D. H. Rothman. 1993. Non-Newtonian flow (through porous media): a lattice-Boltzmann method. *Geophys. Res. Lett.*, **20**(8), 679–682.
- Balise, M.J. and C.F. Raymond. 1985. Transfer of basal sliding variations to the surface of a linearly viscous glacier. *J. Glaciol.*, **31**(109), 308–318.
- Benzi, R., S. Succi and M. Vergassola. 1992. The lattice Boltzmann equation: theory and applications. *Phys. Rep.*, **222**(3), 145–197.
- Chen, S. and 6 others. 1991. Lattice gas automata for flow through porous media. *Physica D*, **28**, 72–84.
- Frisch, U., B. Hasslacher and Y. Pomeau. 1986. Lattice-gas automata for the Navier-Stokes equation. *Phys. Rev. Lett.*, **56**(14), 1505–1508.
- Frisch, U., D. d'Humières, B. Hasslacher, P. Lallemand, Y. Pomeau and J. Rivet. 1987. Lattice gas hydrodynamics in two and three dimensions. *Complex Systems*, **1**, 649–707.
- Higuera, F.J. and J. Jiménez. 1989. Boltzmann approach to lattice gas simulations. *Europhys. Lett.*, **9**(7), 663–668.
- Hou, S. 1995. Lattice Boltzmann method for incompressible, viscous flow. (Ph.D. dissertation, Kansas State University.)
- Huang, K. 1987. *Statistical mechanics. Second edition.* New York, John Wiley and Sons.
- Humières, D. d' and P. Lallemand. 1987. Numerical simulations of hydrodynamics with lattice gas automata in two dimensions. *Complex Systems*, **1**, 599–632.
- Kadanoff, L., G. McNamara and G. Zanetti. 1987. A Poiseuille viscometer for lattice gas automata. *Complex Systems*, **1**, 791–803.
- Lawniczak, A., D. Dab, R. Kapral and J.P. Boon. 1991. Reactive lattice gas automata. *Physica D*, **28**, 132–158.
- Lewis, H.R. and C.H. Papadimitriou. 1981. *Elements of the theory of computation.* Englewood Cliffs, NJ, Prentice-Hall.
- Logan, J. D. 1987. *Applied mathematics, a contemporary approach.* New York, John Wiley and Sons.
- Mase, G.E. 1970. *Theory and problems of continuum mechanics.* New York, McGraw-Hill.
- Maxwell, J. C. 1859. Illustrations of the dynamical theory of gases. In Brush, S.G. 1966. *Kinetic theory. Volume 1. The nature of gases and heat.* Oxford, Pergamon Press, 148–171. [Reprinted.]
- Meier, M.F., L.A. Rasmussen, R.M. Krimmel, R.W. Olsen and D. Frank. 1985. Photogrammetric determination of surface altitude, terminus position, and ice velocity of Columbia Glacier, Alaska. *U.S. Geol. Surv. Prof. Pap.* 1258-F.
- Paterson, W.S.B. 1994. *The physics of glaciers. Third edition.* Oxford, etc., Elsevier.
- Raymond, C., T. Jóhannesson, T. Pfeffer and M. Sharp. 1987. Propagation of a glacier surge into stagnant ice. *J. Geophys. Res.*, **92**(B9), 9037–9049.
- Salem, J. and S. Wolfram. 1986. Thermodynamics and hydrodynamics of cellular automata. In Wolfram, S., ed. *Theory and application of cellular automata.* Singapore, World Scientific, 362–365.
- Skordos, P.A. 1993. Initial and boundary conditions for the lattice Boltzmann method. *Phys. Rev. E*, **48**(6), 4823–4842.
- Turcotte, D.L. and G. Schubert. 1982. *Geodynamics, applications of continuum physics to geological problems.* New York, John Wiley and Sons.
- Van der Veen, C.J. and I.M. Whillans. 1993. Location of mechanical controls on Columbia Glacier, Alaska, U.S.A., prior to its rapid retreat. *Arct. Alp. Res.*, **25**(2), 99–105.
- Welty, J.R., C.E. Wicks and R.E. Wilson. 1984. *Fundamentals of momentum, heat, and mass transfer. Third edition.* New York, John Wiley and Sons.
- Wolfram, S. 1986. Cellular automata fluids I: basic theory. *J. Stat. Phys.*, **45**(3/4), 471–526.

MS received 10 March and accepted in revised form 19 June 1995