# Forming Terrains by Glacial Erosion

GUILLAUME CORDONNIER, Inria, Université Côte d'Azur, France
GUILLAUME JOUVET, University of Lausanne, Institute of Earth Surface Dynamics, Switzerland
ADRIEN PEYTAVIE, Univ Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR5205, France
JEAN BRAUN, Helmholtz Centre Potsdam, GFZ German Research Centre for Geosciences and Institute of Earth and Environmental Sciences, University of Potsdam, Germany
MARIE-PAULE CANI, École Polytechnique, CNRS (LIX, UMR7161), Institut Polytechnique de Paris, France
BEDRICH BENES, Purdue University, Computer Science, USA
ERIC GALIN, Univ Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR5205, France
ERIC GUÉRIN, Univ Lyon, INSA Lyon, CNRS, UCBL, LIRIS, UMR5205, France
JAMES GAIN, University of Cape Town, South Africa

We introduce the first solution for simulating the formation and evolution of glaciers, together with their attendant erosive effects, for periods covering the combination of glacial and inter-glacial cycles. Our efficient solution includes both a fast yet accurate deep learning-based estimation of high-order ice flows and a new, multi-scale advection scheme enabling us to account for the distinct time scales at which glaciers reach equilibrium compared to eroding the terrain. We combine the resulting glacial erosion model with finer-scale erosive phenomena to account for the transport of debris flowing from cliffs. This enables us to model the formation of terrain shapes not previously adequately modeled in Computer Graphics, ranging from U-shaped and hanging valleys to fjords and glacial lakes.

CCS Concepts: • **Computing methodologies** → **Shape modeling**.

Additional Key Words and Phrases: Glacial Erosion, Landscape, Terrain, Erosion, Simulation of Natural Phenomena

**ACM Reference Format:**
Guillaume Cordonnier, Guillaume Jouvet, Adrien Peytavie, Jean Braun, Marie-Paule Cani, Bedrich Benes, Eric Galin, Eric Guérin, and James Gain. 2023. Forming Terrains by Glacial Erosion. *ACM Trans. Graph.* 42, 4, Article 1 (August 2023), 14 pages. https://doi.org/10.1145/3592422

## 1 INTRODUCTION

Natural terrains exhibit a wide range of forms with varying details, from flat deserts to high-altitude mountains and from smooth to fractal-like, dendritic structures. Geomorphological processes determine their formation, but due to their variety and complexity, their application to Computer Graphics (CG) is beset with challenges.

Authors' addresses: Guillaume Cordonnier, Inria, Université Côte d'Azur, 2004 Rte des Lucioles, Valbonne, 06902 , France, guillaume.cordonnier@inria.fr; Guillaume Jouvet, University of Lausanne, Institute of Earth Surface Dynamics, Lausanne, Switzerland, guillaume.jouvet@unil.ch; Adrien Peytavie, Univ Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR5205, Lyon, France, adrien.peytavie@univ-lyon1.fr; Jean Braun, Helmholtz Centre Potsdam, GFZ German Research Centre for Geosciences and Institute of Earth and Environmental Sciences, University of Potsdam, Potsdam, Germany, jbraun@gfz-potsdam.de; Marie-Paule Cani, École Polytechnique, CNRS (LIX, UMR7161), Institut Polytechnique de Paris, Palaiseau, France, marie-paule.cani@polytechnique.edu; Bedrich Benes, Purdue University, Computer Science, West Lafayette, IN, USA, bbenes@purdue.edu; Eric Galin, Univ Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR5205, Lyon, France, eric.galin@univ-lyon1.fr; Eric Guérin, Univ Lyon, INSA Lyon, CNRS, UCBL, LIRIS, UMR5205, Lyon, France, eric.guerin@insa-lyon.fr; James Gain, University of Cape Town, Cape Town, South Africa, jgain@cs.uct.ac.za.



Our key observation is that glaciers had and have a characteristic and significant effect on terrain shape, dissimilar to other phenomena. During previous ice ages, which have dominated the Earth's climate over the past three million years, glacial erosion shaped much of the terrains around us, ranging from prairies and tundra to mountain ranges, such as the Cordillera and Alps. Mountain glaciers, in particular, create specific landforms (see Section 7.1), including U-shaped (or tunnel) valleys, cirques (circular depressions), arêtes (jagged crests), hanging valleys at the convergence of two glaciers (e.g., the Bridal Veil Valley in Yosemite), glacial lakes, and smaller-scale moraines and drumlins formed from debris and sediment deposited near the margins of glaciers and ice sheets (see Figure 1). Existing terrain generation methods cannot accurately model these features.

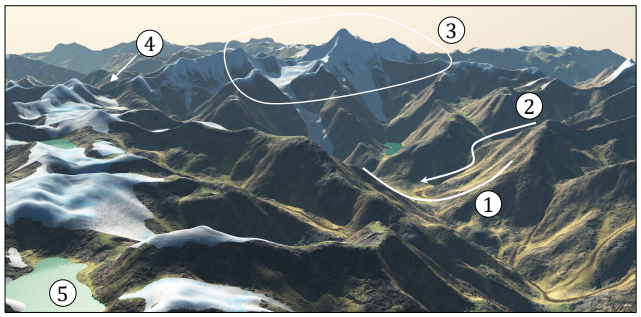



Fig. 1. A landscape carved by our simulated glacier. Specific landforms are (1) U-shaped valleys, (2) hanging valleys, (3) a glacial cirque overhung by arêtes and horns, (4) a pass, and (5) high–altitude lakes.

Simulation of glacial erosion needs to capture the nature of glaciers as bodies of dense ice resulting from a balance of snow accumulation at high altitudes and melting and sublimation at lower altitudes. A critical aspect is the time scale. Glaciers abrade surfaces and transport debris impacting the bedrock. However, bedrock erosion (1-10 mm per year) is much slower than the glacial flow (up to 10km per year), and the carving of U-shaped valleys typically occurs over hundreds of thousands of years (i.e., one to several glacial and inter-glacial cycles). This difference in time scales makes glacial simulation challenging.

Previous work [Argudo et al. 2020] attempts to solve the problem using a shallow-ice approximation (SIA) of the Stokes equation from glaciology. This solution requires small time steps (a few days) for stability, which makes it ill-suited to the large time spans necessary to model glacial erosion (time steps of 10-100 years). Moreover, the SIA does not model the viscous shear stresses of the ice, i.e., moving ice parcels dragging the neighboring ones. These models are unable to capture the formation of U-shaped valleys and related landforms, such as hanging valleys and glacial lakes.

We present a novel, geologically validated, efficient model for glacial erosion. Our solution captures the expansion and retreat of glaciers during glacial and interglacial periods, as well as their erosive impact on the underlying terrain.

Our scientific contributions are as follows: 1) A new, efficient solver for the high-order Blatter ice flow equations based on unsupervised learning used to solve an energy-based ice-flow model; 2) A new, adaptive multi-scale law for ice transport at different time scales, enabling our glacier models to reach equilibrium quickly, and 3) A solution for combining the abrasion and quarrying effects of glacial erosion with slope erosion on surfaces exposed by retreating ice, despite the different space and time scales.

Our method generates characteristic landforms that are visually prominent but impossible to obtain with existing simulations, particularly U-shaped, hanging valleys, eroded cliffs, cirques, ridges, passes, horns, fjords, and glacial lakes. Those shown in Figure 1 were computed in less than 5 minutes on a $16 \times 16\,\mathrm{km}^2$ terrain.

## 2 PREVIOUS WORK

Terrain generation in Computer Graphics is generally tackled using procedural generation, example-based synthesis, and erosion simulation [Galin et al. 2019]. In terms of glacial erosion specifically, the focus has been on geological models and learning-based approaches.

*Procedural generation.* Fractals, subdivisions, and noise-based algorithms are often used to generate terrains that exhibit characteristic landscape features, such as self-similarity. There are currently no procedural methods focused specifically on glacial phenomena. Nevertheless, it might be possible to create a palette of procedural brushes or pens to paint or sketch glacial features [de Carpentier and Bidarra 2009; Guérin et al. 2022]. Unfortunately, this would place the onus of ensuring realism, particularly in the complex interaction between glacial landforms on the digital artist, which would require both significant effort and geomorphological expertise.

*Example-based synthesis.* Zhou et al. [2007] and Guérin et al. [2016] generate terrains based on a corpus of scanned digital elevation models. If the input terrains incorporate glacial features, then this will be reflected in the output. Unfortunately, the source data is almost invariably a single surface, so the terrain below water (e.g., fjords) or ice (e.g., glaciers) is unavailable. It is also not possible to add glacial erosion to existing landscapes because this would require source data showing corresponding before and after states.

*Simulations.* Erosion simulations approximate the geological evolution of terrain through iterations of hydraulic action [Benes et al. 2006; Krištof et al. 2009], thermal erosion [Musgrave et al. 1989], tectonic uplift combined with stream power erosion [Cordonnier et al. 2017a; Schott et al. 2023], or a combination of localized erosion effects [Cordonnier et al. 2017b] applied to an initial base terrain. While methods based on heightfields cannot represent overhangs, other methods approximating wind erosion, hydraulic erosion, or invasion percolation on volumetric structures can result in realistic cliffs, arches, and caves [Beardall et al. 2007; Franke and Müller 2022; Paris et al. 2019, 2021; Peytavie et al. 2009]. Glacier erosion, in contrast, has yet to be tackled.

In terms of glacier simulation, Argudo et al. [2020] use a method based on the Shallow Ice Approximation (SIA), together with procedural amplification, to simulate the flow and retreat of glaciers over hundreds of years. Although this shares our focus on glaciers, there is otherwise significant divergence in terms of goal and methodology. Argudo et al. [2020] do not change the topography of the input terrain but overlay glaciers on top. This is likely because the SIA is entirely unsuited to glacial erosion since it does not model the viscous stress within glaciers. In contrast, our goal is also to generate terrains that evidence landforms arising from glacial erosion. To achieve this, we employ different ice models with different numerical treatments.

*Geological models.* Geologists have explored and validated ice flow and glacial erosion over many decades [Braun et al. 1999; Egholm et al. 2011, 2012b; Harbor et al. 1988; Headley et al. 2012; Mahaffy 1976; Sternai et al. 2013], showing that glacial erosion depends on the speed of the ice in contact with bedrock [Bernard 1979; Herman et al. 2015]. Their motivation is a search for governing equations that match field observations, ranging from measured ice flows to identified and classified landforms.

While glacier flow, covering the range from ice sheets to extensive glaciers, is typically modeled using the Shallow Ice Approximation of the Stokes equations, this does not account for the constriction of ice in narrow alpine valleys and thus cannot predict the formation of characteristic U-shaped valleys [Egholm et al. 2011]. Braun et al. [1999] tackled this problem by adding a factor proportional to the width of the valley while computing sliding ice speed, which in turn dictates erosion of the underlying bedrock. In our work, we rely on a more accurate model from glaciology that uses the highly non-linear Glen-Stokes equation for modeling ice flow [Glen 1955].

*Learning-based approaches.* One promising avenue to accelerate ice flow computation, as advocated by Jouvet et al. [2022], is to apply ML trained on a corpus of high-fidelity simulations to predict vertically averaged horizontal velocities. They use a Convolutional Neural Network (CNN) that makes predictions based on initial conditions relating to the rock-ice interface and the basal topography of ice thickness and surface slope. Unfortunately, prediction quality degrades when the topography is not a good fit for the learning dataset, as often occurs when erosion is factored in. While we share the idea of using ML to accelerate the simulation of glaciers, we use non-supervised learning based on a physical loss, which avoids their costly supervised training on a precomputed dataset. Another alternative is Physics Informed Neural Networks (PINN) [Raissi et al. 2019], which offer a meshless approach with the network trained to map a position in space to a physical variable, *e.g.,* the velocity. These are easy to implement and their variational counterparts [Li et al. 2021] show that, instead of optimizing a loss made of the sum of the

residual form of the PDE and the boundary conditions, minimizing an energy improves convergence and simplifies hyper-parameter search. This technique has been used for glacier modeling [Cui et al. 2022] and motivated our energy formulation, but it requires the network to be fully retrained for any change in the glacier geometry. In contrast, we input the geometry of the glacier in the network, which allows an efficient fine-tuning strategy when the computation of ice velocities is coupled with the motion of the glacier.

## 3 OVERVIEW

In this section, we first cover the fundamental background of ice flow models and introduce the necessary notation before presenting our framework.

### 3.1 Background on ice flow models

Let us consider a glacier on a bedrock of height $z$, of local ice thickness $h$, elevation $s = z + h$, and ice velocity $\bar{\mathbf{u}}$, standing for the horizontal velocity averaged over a vertical column at any $(x, y)$ position in the domain (see Figure 2).

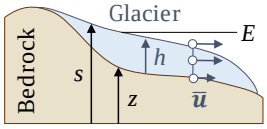


Fig. 2. Notations.

The change in ice thickness over time is governed by a mass conservation equation, where $M$ is the mass balance denoting the local difference between snow precipitation and melting at a point:

$$\frac{\partial h}{\partial t} + \nabla\cdot(\bar{\mathbf{u}}\, h) = M. \tag{1}$$

More specifically, $M$ is computed as a proportion $\beta$ of the difference between the local ice surface elevation and the equilibrium altitude $E$ at which melting counterbalances precipitation:

$$M = \min(p_{\max},\ \beta\,(s - E)). \tag{2}$$

$p_{\max}$ denotes the maximum accumulation rate. The mass balance through parameters $p_{\max}$, $\beta$, and $E$ is critical to the size and shape of the glacier, and ultimately the extent and depth of erosion. Eqn. 1 only holds when $h > 0$; otherwise, $h$ is set to zero as a boundary condition. Given vertically averaged velocity $\bar{\mathbf{u}}$ it is possible using this mass balance to solve for the change in ice thickness $h$ over time (Sec. 5). Thus, the question becomes how to compute the velocity of ice within a glacier.

Ice flow exhibits non-Newtonian fluid behavior characterized by incompressibility and non-linear stress response as described by the Glen-Stokes model [Glen 1955]:

$$\tau = 2\mu\mathbf{D}(\mathbf{v}). \tag{3}$$

The parameter $\tau$ is the deviatoric stress tensor, $\mathbf{D}(\mathbf{v}) = \frac{1}{2}(\nabla\mathbf{v} + \nabla\mathbf{v}^{\mathrm{T}})$ is the strain rate tensor response, and $\mu$ is the viscosity, defined as:

$$\mu = \frac{1}{2}B\,|\mathbf{D}(\mathbf{v})|^{p-2}. \tag{4}$$

$B$ and $p \in (1, 2]$ are parameters that depend on atmospheric conditions and ice composition.

A closed form of the equation can be derived by exploiting incompressibility ($\nabla \cdot \mathbf{v} = 0$) and momentum conservation, in which acceleration terms are discarded:

$$-\nabla \cdot (\tau - P\,\mathbf{I}) = \rho\,\mathbf{g}. \tag{5}$$

$P$ is the pressure, I the identity matrix, $\rho$ the ice density and $\mathbf{g}$ the gravitational force. Glacial erosion is driven by the sliding of glaciers over bedrock, with the sliding being modeled by a non-linear friction law at the ice-bed interface [Weertman 1957]:

$$((\mathbf{I} - \mathbf{n}\mathbf{n}^{\mathrm{T}})\tau)\cdot\mathbf{n} = -C|(\mathbf{I} - \mathbf{n}\mathbf{n}^{\mathrm{T}})\cdot\mathbf{v}|^{\sigma-2}(\mathbf{I} - \mathbf{n}\mathbf{n}^{\mathrm{T}})\cdot\mathbf{v}. \tag{6}$$

The friction parameters $C$ and $\sigma$ are tuned to observations, and $\mathbf{n}$ is the vector normal to the ice-bedrock interface.

Solving the full Glen-Stokes formulation is computationally expensive and challenging, mostly because of the inherent incompressibility constraints. As a consequence, computational geologists typically resort to one of three simplifications [Jouvet 2016]: Shallow Ice Approximation (SIA), Shallow Shelf Approximation (SSA), or the Blatter Model. SIA is a zero-order 2D model that removes longitudinal and transverse stresses, as well as vertical stress gradients from $\mathbf{D}(\mathbf{v})$. SSA incorporates horizontal stresses but assumes that the ice velocity does not vary vertically. The least simplifying in its assumptions, the Blatter model, is a higher-order approach that solves only for the horizontal components of $\mathbf{D}(\mathbf{v})$.

While it provides an efficient closed-form solution for vertically averaged velocities, the SIA suffers from two key disadvantages. First, stability depends on the rate of diffusion, which is particularly high in glacial dynamics, necessitating short time steps. Second, the simplification strips away the forms of stress necessary to effectively calculate erosion on non-flat topography. It is mostly used in practice where the ice flow is dominated by vertical shearing, which is the case in ice caps but not on mountain glaciers. The SSA suffers from the opposite problem: accurate for thin shelf-like glaciers; it underpredicts the velocity of ice caps and deep glaciers occupying primary valleys. This leaves the Blatter model as the most appropriate approximation scheme for our purposes.

### 3.2 Computational framework

Simulating glacial erosion requires solving three interleaved challenges (see Figure 3): 1) computing the ice flow (velocity), 2) transporting the ice with time steps that account for the differing time scales required for flow and erosion, over geological time spans and 3) computing the combined effect of glacial erosion and of more classical erosion phenomena.

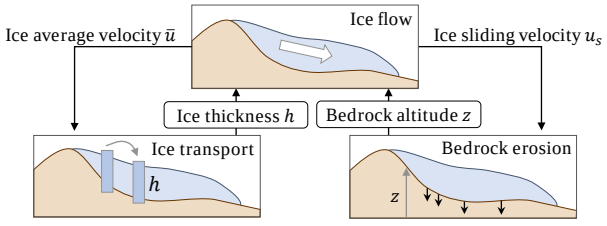


Fig. 3. At each time step, our simulation uses a fast, learning-enhanced prediction of the ice flow. The resulting vertically averaged ice velocity is used in an ice transport algorithm that models the advance of glaciers and the sliding velocity to determine the bedrock abrasion. The updated ice thickness and bedrock altitude are fed back to the ice flow model for the next time step.

Our approach is as follows. First, we rely on the highly non-linear Glen-Stokes equation for modeling the ice flow. To accelerate the evaluation, we use an energy formulation for the Blatter model and enhance it with a new unsupervised learning technique that enables efficient velocity prediction without any need for a training dataset (Figure 3: ice flow, and Section 4).

Second, we introduce a new, adaptive multi-scale transport mechanism to efficiently simulate the ice at different time scales (Figure 3: ice transport, and Section 5). Central to this is the observation that glaciers often retain a stable shape for extended periods. While the ice moves continuously within, the geometric hull of the glacier itself quickly reaches an equilibrium driven by the balance between snow accumulation, ice transport, and melting. A standard, explicit Euler simulation would require small time steps, resulting in lengthy simulations with no visual change in shape. Although less accurate during transitions between equilibrium states, our novel solution allows the glacier to reach equilibrium very quickly for long periods of steady state.

Third, glacial erosion through the abrasion and quarrying effects of the glacier sliding on the bedrock needs to be coupled with classical hydraulic and thermal erosion laws. This is essential in capturing the evolution of glaciated landforms over long periods and the formation of the steep slopes left behind by retreating glaciers. We also extend our model to account for erosion due to debris flow and small landslides (see Figure 3: bedrock erosion, and Section 6).

## 4 ACCELERATED ICE FLOW EVALUATION

Inspired by Jouvet et al. [2022], we use ML to accelerate the evaluation of the Glen-Stokes equation for ice flow. In contrast to previous work, we, however, propose an on-the-fly, unsupervised method that achieves simultaneous training and prediction. This has two critical advantages: first, the model exhibits a surefooted adaptation to eroded topography without the need for the corresponding data; second, it avoids both the lengthy dataset construction and the pre-training phase.

### 4.1 Deep energy minimization

We use a Convolutional Neural Network (CNN) to infer ice velocities from a map representing the glacier geometry (thickness and surface gradients). Our goal is to optimize the weights of the CNN so that it minimizes the following energy, corresponding to Blatter's formulation of the Glen-Stokes equation [Blatter 1995; Jouvet 2016]:

$$\mathcal{E}(\mathbf{u}) = \int_V \frac{B}{p} |\underline{\mathrm{D}}(\mathbf{u})|^p \, dV + \int_{\Gamma_m} \frac{C}{\sigma} |\mathbf{u}|_M^\sigma \, dS + \rho g \int_V (\nabla s \cdot \mathbf{u}) \, dV. \quad (7)$$

This energy is derived from Eqns. 3 to 6. The focus here is on the $(x, y)$ horizontal velocity components, denoted by $\mathbf{u} = \mathbf{v}|_{\mathbf{R}^2} \in \mathbf{R}^2$. Integration is over the volume of the glacier $V$ in the first and third terms and over the basal surface area of the sliding section of the glacier $\Gamma_m$ in the second term. The constants $B$, $C$, $p$, $\sigma$, $g$, and $\rho$ depend on initial conditions (values used in our experiments appear in Table 1). The $M$-norm is defined as: $|\mathbf{u}|_M^2 = \mathbf{u}_x^2 + \mathbf{u}_y^2 + (\mathbf{u}_x \, \partial_z \, \mathbf{u}_x + \mathbf{u}_y \, \partial_z \, \mathbf{u}_y)^2$. Crucially, the strain $\mathbf{D}(\mathbf{v}) = (\nabla \mathbf{v} + \nabla \mathbf{v}^T)/2$ is replaced by a first-order approximation of the height/width ratio of the glacier, which discards the vertical component of the velocity:

$$\underline{\mathrm{D}}(\mathbf{u}) = \begin{pmatrix} \partial_x \mathbf{u}_x & \frac{1}{2}(\partial_y \mathbf{u}_x + \partial_x \mathbf{u}_y) & \frac{1}{2}\partial_z \mathbf{u}_x \\ \frac{1}{2}(\partial_y \mathbf{u}_x + \partial_x \mathbf{u}_y) & \partial_y \mathbf{u}_y & \frac{1}{2}\partial_z \mathbf{u}_y \\ \frac{1}{2}\partial_z \mathbf{u}_x & \frac{1}{2}\partial_z \mathbf{u}_y & -\partial_x \mathbf{u}_x - \partial_y \mathbf{u}_y \end{pmatrix}$$

Note that while the velocity is reduced to a $2D$ vector $\mathbf{u} = (\mathbf{u}_x, \mathbf{u}_y)$, this is still a $3D$ equation over the volumetric domain $V$, defined above the sliding interface surface $\Gamma_M$. In practice, we set the same sliding conditions everywhere at the bedrock interface.

A fundamental difference from PINNs [Cui et al. 2022; Raissi et al. 2019] is that our network does not take sample coordinates as input, but instead uses the geometry of the glacier: ice thickness and surface gradients organized in a regular grid. This approach allows iterative fine-tuning of the network as the transport progresses. Thanks to relatively small variations in the glacier after each time step and large spatial self-similarities in the ice geometry, The network converges sufficiently fast to correct the error in the training before it has any significant impact on ice transport.

### 4.2 Discretization and Implementation

Even though the velocity vectors $\mathbf{u}$ in our energy formulation are two-dimensional they are located within the three-dimensional volume of the glacier. Thus, in order to numerically evaluate the integrals in Eqn. 7 we discretize the domain of the glacier into a lattice as follows (see Figure 4):

(1) Regular basal grid. The basal domain is regularly and evenly subdivided horizontally in $x$ and $y$. This grid aligns exactly in terms of resolution and horizontal vertex positioning with the bedrock elevation and ice thickness maps.
(2) Fixed number of vertical layers. $N$ grid layers are then placed between the bedrock and the ice surface in such a way that they are evenly separated vertically. This means adjusting the z-value of vertices in the resulting lattice depending on the local variation in bedrock $z$ and ice thickness $h$ values.

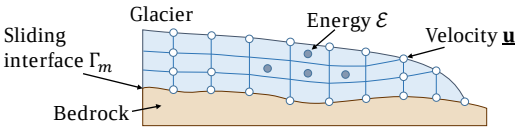



Fig. 4. Discretization of the 3D glacier volume.

This defines a lattice that exhibits square faces in the $x-y$ plane but trapezoids in the $x-z$ and $y-z$ planes. To guard against numerical ill-conditioning (diagnosed in our matrix-free case as excessively large gradients), a minimum thickness $dz_{min}$ is enforced between adjacent layers. The upper layers' spacings are set to zero in cases where the ice is overly thin ($h < N \, dz_{min}$). This correction may collapse some vertical edges of the lattice, in which case we set the velocities stored at the endpoints of the vertical edge to be equal.

In this lattice, velocity values $\mathbf{u}$ are located at the vertices, while energy computations are undertaken in the center of each distorted cell, strictly a trapezoidal prism. Interpolation of values and computation of differentials is based on approximating the cells as linear finite elements. Likewise, any planar quantities, such as the bedrock

and ice surface gradients required by Eqn. 7, can be derived at grid centers via interpolation (see Appendix).

For the CNN velocity prediction, we use as input a Gaussian stack constructed from the ice thickness, bedrock slope in each of the 4 cardinal directions ($+x, -x, +y, -y$), and local sliding parameters. All of these values are zeroed if there is no local glacial ice present. The Gaussian stack is a common strategy [Lefebvre and Hoppe 2005] for incorporating surrounding information in a multiresolution neighborhood. We implement advection with alias-free upsampling and downsampling operators using a convolution of up/down-sampling and a Gaussian kernel at different levels. Every level $i$ is a sequence of $i$ downsampling operators followed by $i$ upsampling steps. We use five levels, a standard deviation of one grid cell, and a Gaussian kernel size of ten grid cell. The output of the CNN is the velocities at vertices in the lattice.

In summary, ice flow computation proceeds as follows: 1) The CNN predicts horizontal velocities at vertices in the volumetric lattice. 2) These velocities are refined with a single step of gradient descent on the energy functional in Eqn. 7 calculated at cell centers. 3) The gradients of the velocities are propagated back to the CNN to update the internal weights. 4) Velocities are summed vertically and normalized by ice thickness to obtain the vertically-averaged velocity $\bar{u}$ required by Eqn. 1. We now can calculate the transformation of the glacier's shape over time.

## 5 ADAPTIVE, MULTI-SCALE ICE TRANSPORT

Unfortunately, the process of ice transport, while seemingly straightforward, is also bedevilled by numerical issues. Solving Eqn. 1 to obtain the change in glacier shape, represented by ice thickness $h$, given vertically averaged velocities $\bar{\mathbf{u}}$, is an advection process. Evaluating this process with an implicit time-step scheme is infeasible because of the complex relationship between the velocities and glacier geometry. Rather, we use an explicit (forward Euler) scheme, which means that for stability reasons, the timestep $\mathrm{d}t$ is limited by grid-cell size $\mathrm{d}x$ and the velocity magnitude $\|\bar{\mathbf{u}}\|$, according to the CFL conditions $\mathrm{d}t < c\,\mathrm{d}x/\|\bar{\mathbf{u}}\|$ (we found that $c = 0.1$ suppressed all instability in the ice model). In normal circumstances, with $\mathrm{d}x = 100m$ and a maximum velocity of about $100m/y$, this limits timesteps to about $1/10$ of a year. This is orders of magnitude below our erosion timestep target of 10 to 100 years. Moreover, we have experienced situations with unrealistically excessive ice velocity, for instance, when modeling very steep slopes where the theoretical time step would be proportionally small. In addition, we target spatial resolutions smaller than $100m$.

Fortunately, we can exploit two characteristics of glaciers to increase timestep size and hence computational efficiency: (1) Excessively high velocities tend to only occur in localized and extreme circumstances, such as thin ice on steep slopes, and (2) in the timeframe of erosion, Eqn. 1 rapidly settles into a steady state.

Broadly speaking, this is achieved by adjusting the timestep in both the spatial domain, according to the local velocity, and in the frequency domain, according to Gaussian residuals. We first cap the user-provided target timestep $\mathrm{d}t$, typically $10 - 100$ years, according to a position-dependant maximum that ensures stability *locally*: $\delta t(x) = \min(\mathrm{d}t, c\,\mathrm{d}x/\|\bar{\mathbf{u}}(x)\|)$. This adapted time step caters for outliers and accelerates the convergence to a steady state, not sufficiently, however as the stability condition still heavily limits ice transport in fast-moving regions. Therefore, we propose a new method to converge rapidly to the steady state, possibly by sacrificing accuracy during transient periods.

Inspired by multigrid techniques, we use frequency decomposition to separate ice transport into layers at different scales. The advection timestep can be set to correspond to the scale of transport of a given layer. We collect the input terms of Eqn. 1 as:

$$T = M - \nabla\cdot(\bar{\mathbf{u}}\,h),$$

where $T$ represents the change of elevation per unit of time, and the divergence is computed *upwind*: the velocities are averaged at the grid edges in the $x$ and $y$ directions, then multiplied by the ice thickness $h$ taken from the cell in the opposite direction to the velocity. To ensure that an excessive negative mass balance does not swamp the lower frequencies, we clamp the minimum $T$ to zero outside the glacier. We construct a stack of $n$ Gaussian residuals $R_i$, $i = 0, \ldots, n-1$ by successive iterations of Gaussian smoothing and up/down sampling (similar to Section 4) starting from $T = T_0$ to generate $T_1, \ldots, T_n$. Each layer of the residual stack represents the difference between successive levels of Gaussian smoothing $R_i = T_i - T_{i+1}$. We then recombine the residual stack to form a stabilized transport term:

$$T' = \sum_i \min(f^i\,\delta t(x), \mathrm{d}t)\,R_i.$$

Recall that $\mathrm{d}t$ is the initial target timestep, and $\delta t(x)$ is the locally adapted timestep in the base pre-smoothed grid. Note that $\delta t(x) = \mathrm{d}t$ where the velocity is sufficiently small, in particular at the exterior of the glacier, which is critical for preserving the boundary. The scaling factor $f \in [1, 2]$ controls the efficiency of this multi-scale strategy. We set it experimentally to $f = 1.7$ by observing that larger values introduce oscillations that do not stabilize between different frequencies. The variable $T'$ now holds an approximation of the change in ice thickness after a timestep $\mathrm{d}t$, which we can use to advance the glacier:

$$h(t + \mathrm{d}t) = \max(0, h(t) + T').$$

In practice, this strategy succeeds because the high-frequency layers move and diffuse ice over large spatial scales, very quickly converging to zero, leaving room for the lower frequencies to correct the resulting approximation error progressively. We propose an evaluation of the behavior of this model in Section. 7. We note that this seems to work particularly well for glaciers because destabilization from the steady state due to a change in climate or bedrock generally affects glacial features proportionally to their scale.

## 6 GLACIAL EROSION AND FLUVIAL PROCESSES

The combination of glacial erosion with other phenomena is essential to generate plausible terrains. Indeed, the typical period of mountain formation is several orders of magnitude longer than the cycles of glaciation, so most landscapes have been exposed to various climates, including interglacial periods where other types of erosion phenomena predominate. The latter do not cease during an

ice age, as the glacierless lower foothills of a mountain continue to be shaped by other prevailing erosion processes.

We incorporate glacial, hill slope, debris flow, and fluvial erosion due to their impact on landscapes. Ice shields the terrain from non-glacial erosion processes, so we do not apply them when the ice thickness is above a threshold (3 m in our experiments). In our simulations, erosion is applied in the following order: glacial erosion, then, where ice is sufficiently thin, hill slope, followed by combined fluvial and debris flow erosion. Figure 5 shows a comparison of the impact of individual erosive processes on the final shape of the terrain.

## 6.1 Glacial erosion

Glacial erosion is an open research topic in geomorphology, especially at the finest spatial resolutions, primarily due to a lack of efficient models that adequately capture viscous stresses in ice. However, two primary forms of erosion are widely studied [Beaud et al. 2014]: abrasion caused by the sliding action of glaciers, and quarrying (or plucking), where the ice infiltrates the bedrock, cracking large chunks of rock that are then transported by the glacier.

We model abrasion following a power law of the sliding velocity $\mathbf{u_s}$ [Herman et al. 2015]:

$$\frac{\partial z}{\partial t} = -k_e \, \|\mathbf{u_s}\|^l.$$

We use the approach of Egholm et al. [2012b] for glacial quarrying:

$$\frac{\partial z}{\partial t} = -k_q \, \|\mathbf{u_s}\| \, Q\left(\mathbf{u_s}, \nabla z\right).$$

$Q$ focuses the quarrying efficiency on slopes aligned with the sliding velocity acting on the bedrock altitude $z$:

$$Q\left(\mathbf{u_s}, \nabla z\right) = \frac{1}{2}\left(\operatorname{erf}\left(-\frac{\mathbf{u_s} \cdot \nabla z}{0.4\mathbf{u_s}}\right) + 1\right).$$

Here, $k_e$, $k_q$, and $l$ represent various erosion constants (see Table 1), and erf is the Gaussian error function.

We obtain the sliding velocities $\mathbf{u_s}$ by selecting the velocities output by the network at the bottom-most layer of the velocity lattice and compute the gradients of the bedrock $\nabla z$ downward in the $x$ and $y$ directions to prevent aliasing artifacts.

More accurate formulations for quarrying exist [Magrani et al. 2022], but they depend heavily on subglacial hydrology, which is beyond the scope of this paper. Quarrying and abrasion give rise to visually distinct outcomes: abrasion carves deeply and enlarges the main glacial valleys, while quarrying flattens high altitudes and produces small lakes just below the ridges.

## 6.2 Debris flow

Rock and mudslides on slopes gain a lot of inertia and have a vast erosive power responsible for gouging cliffs at small to medium scales. This phenomenon is crucial because retreating glaciers leave exposed cliff faces susceptible to debris flow. Our method improves on classical thermal erosion [Musgrave et al. 1989], which reduces bedrock slopes toward an equilibrium angle, by adding an important term referenced in geology [Stock and Dietrich 2003] to account for drainage:

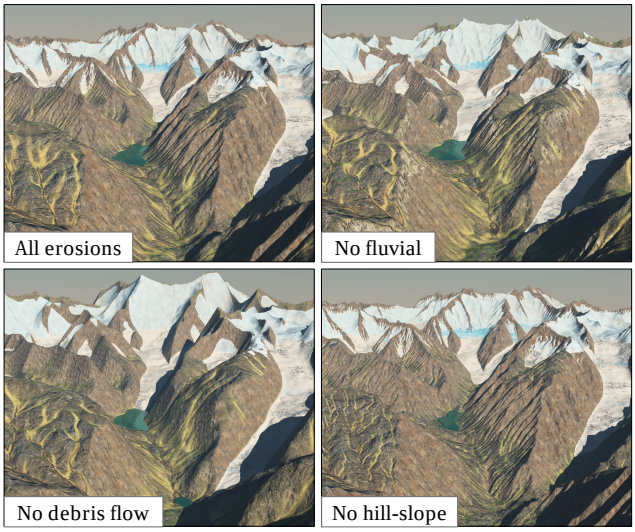



Fig. 5. Compared to the complete model (top left), disabling each erosive effect shows their relative influence on the final result. Debris flow (bottom left) erodes the steepest slopes, fluvial erosion (top right) sharpens the base of U-shaped valleys, and hill slope (bottom right) smoothes the sharp ridges at the drainage divide.

$$\frac{\partial z}{\partial t} = -k_{df} \, \left(1 + k_{da} A^q\right) \max\left(0, \left(\|\nabla z\| - s_c\right)\right)^{q'}. \tag{8}$$

The parameter $k_{df}$ controls erosive strength, $k_{da}$ and $q$ weight the influence of drainage on erosion, $q'$ adds non-linearity in the influence of the slope, and $s_c$ is a critical gradient below which the erosion has no effect (Table 1 in Appendix shows the typical values).

Drainage area $A$ is computed similarly to water flows: from highest to lowest elevation by accumulating cell area $dx^2$ consistently in the direction of steepest descent [Braun and Willett 2013]. Eqn. 8 is solved explicitly, using the Euclidean norm of the downward slope (the $x$ and $y$ components of the bedrock gradient are chosen as the steepest downward directions in $x$ and $y$ directions, respectively).

To avoid sink nodes in the ordering graph, we compute it on a virtual monotonically decreasing surface obtained after depression filling [Barnes et al. 2014; Cordonnier et al. 2019].

## 6.3 Fluvial erosion

Fluvial erosion is the fundamental phenomenon that shapes mountain ranges [Cordonnier et al. 2017b]. The erosion at a cell $\mathbf{c}$ is a power function of the local bedrock slope and the drainage area $A$, which is related to the water flux that flows through $\mathbf{c}$:

$$\frac{\partial z}{\partial t} = -k_f \, A^m \|\nabla z\|^n,$$

where $k_f$, $m$, and $n$ are real coefficients (see Table 1 in Appendix). The equation is similar to Eqn. (8), and we apply the same algorithm to solve it, which makes combining these effects a natural and efficient choice.

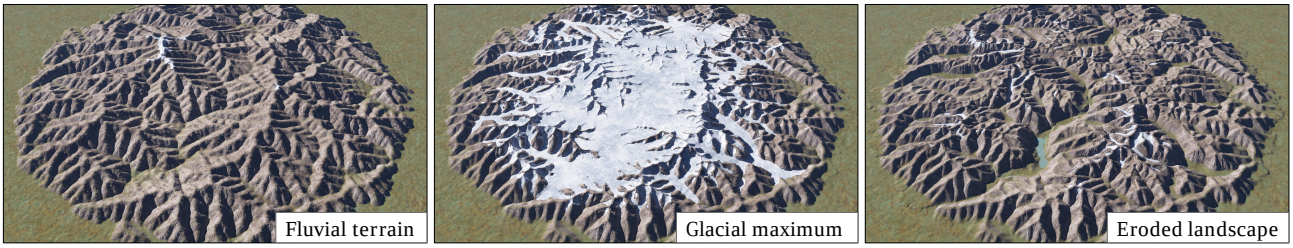



Fig. 6. Far view of a $64 \times 64\,\mathrm{km}^2$ terrain, generated with fluvial erosion only (left). After glaciation reaches its maximum after 100,000 years (center), the glacier retreats over a final 20,000 years and leaves a landscape marked by glacial erosion.

### 6.4 Hill-slope erosion

While *fluvial erosion* is valid for large drainage areas, *hill-slope erosion* is dominant along ridges and accounts for more subtle erosion due to wind, rain, or gravel and sediment accumulating throughout long-standing erosion of the terrain [Braun and Sambridge 1997]. It can be modeled as a simple diffusion equation with constant $k_h$:

$$\frac{\partial z}{\partial t} = k_h \, \Delta z.$$

We employ this equation by using an Alternating Direction Implicit Scheme [Wachspress and Habetler 1960] with long time steps when we want to generate terrains without glaciers or explicitly on the GPU to accompany glacial erosion.

## 7 RESULTS AND DISCUSSION

All components of our simulation, including automatic differentiation and optimization, are written in PyTorch. Thus, most of the algorithms execute on the GPU, except drainage computation (see Sections 6.2 and 6.3). To compensate for this use of CPU, we run 20 steps of ice flow prediction, ice transport, and erosion on the GPU before updating the drainage and elevation of the lakes on the CPU.

All results and timings were produced on a single workstation equipped with a 16-core Intel Xeon E5-2650 CPU (2.20GHz, 64 GB RAM) and an NVidia RTX A500 GPU (24 GB). For the CNN prediction, we use a simple CNN with ReLU activation functions and six layers with 64 features each. This architecture, together with the Gaussian stack input (Section 5), performed better than more advanced architectures such as UNet or DenseNET and result in a 93% accuracy at steady state compared to directly optimizing the energy in Eqn. 7. We use the ADAM optimizer with a learning rate of $10^{-4}$ to perform the optimization. Our implementation is available at: https://gitlab.inria.fr/landscapes/glacial-erosion.

Unless specified otherwise, the parameters used throughout our evaluation are given in Table 1. The alternation between warm interglacial periods and glaciation was simulated on a $64 \times 64\,\mathrm{km}^2$ terrain with 64 m grid cells.

We used multi-fractal noise to simulate variation in uplift and rock resistance and parameterize a fluvial erosion model that initializes the mountains. To demonstrate the impact of glacial erosion, warm climatic conditions were first simulated on a fractal terrain over a 3 million years period, with $\mathrm{d}t = 25,000$ years, using an implicit implementation of fluvial erosion [Cordonnier et al. 2016] (see Figure 6, left). We then applied fast cooling conditions that lowered $E$ from 3,000m (mountain tops) to $1,200m$ over 100,000 years (Figure 6, center) and reduced $\mathrm{d}t$ to 100 years. Then, we set the climate to fast warming conditions ($E$ rose to $1,700\,m$ in 20,000 years) to allow other processes to progressively smooth glacier-dominated landforms (Figure 6, right). The video shows a second glaciation of the same climatic magnitude but with reduced erosion impact, which is consistent with results in geomorphology [Egholm et al. 2009] since glacial erosion carves most of the relief above the snowline altitude $E$.

### 7.1 Validation of landforms

Our end goal is a unified simulation of glacial erosion that enables the emergence of the most prominent glacial features in real mountain ranges [Benn and Evans 2014]. Our validation strategy is to run simulations with default settings and observe if the main patterns of glacial erosion emerge without any intervention from the user. In practice, we run the climatic scenario described earlier but on a smaller scale terrain ($16 \times 16\,\mathrm{km}^2$ with 32 m grid cells) to enable close inspection. The following characteristic features appear consistently in our results (see Figure 8):

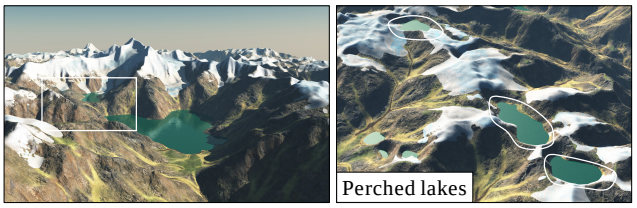



Fig. 7. Two types of glacial lakes produced by our method: over-deepenings at the bottom of valleys (left), and high-altitude perched lakes (right).

**U-shaped valleys:** Their characteristic flat-bedded, steep-walled shape differentiates them from V-shaped valleys carved by fluvial erosion. This shape originates from the viscous stresses of the ice: the fast velocity at the glacier core and the slower-paced ice constrained by the walls interplay to produce a parabolic shape [Svensson 1959].

**Passes:** Passes often originate from a glacier overflowing a valley into its neighbor, resulting in smooth breaches in the ridge line.

**Lakes:** Also called over-deepening, lakes are common footprints of glaciers, ranging from vast lakes, such as the Great Lakes in the US, to high-altitude sequences of pocket lakes. In our results, we observe lakes at the bottom of the main valleys (Figure 7, left) but

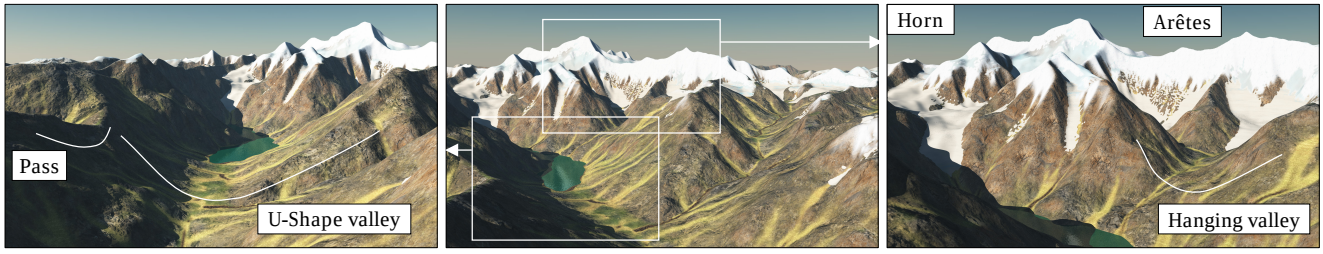
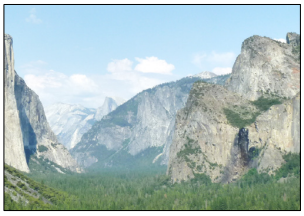



Fig. 8. The retreating glacier forms a typical glacial landscape (center), marked by U shape valleys, lakes and passes (left), hanging valleys, as well as horns and ridges above glacial cirques (right).

also close to ridges at higher elevations (Figure 7, right). Smaller lakes tend to form more often than vast ones due to a different local rock resistance to quarrying, which provides a powerful control tool.

**Hanging valleys:** The velocity of the glacier, even at the bedrock interface, is mainly driven by gravity and by the slope. Thus, if a small glacier flows into a larger one, it will connect in a way that ensures continuity of the ice surface, but not necessarily continuity of the ice-bedrock interface.



After erosion, this differential basal carving forms characteristic hanging valleys. See the inset picture of the Bridalveil Fall in Yosemite National Park for reference. Note that this is different from fluvial erosion, which acts with comparatively small river streams that tend to join and connect valleys, but are unable to create features on the scale of hanging valleys.

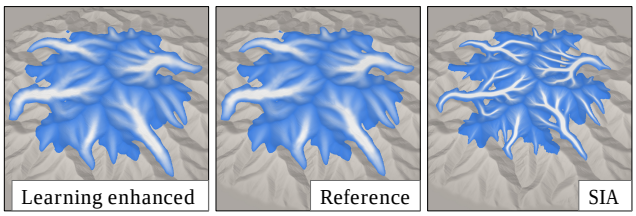



Fig. 9. We compare our learning-enhanced optimization to a direct optimization (reference) and the SIA, at a 32 m cell resolution after reaching steady-state (1,000 time steps in our case, 10,000 for the SIA). Our method predicts both ice flow and ice volume comparable to the reference (at 93% and 97%, respectively). In comparison, the SIA loses about 49% of the glacier volume. Furthermore, the magnitude of the velocities, indicated with a blue-white ramp, show that, due to the lack of viscous stresses, the SIA velocities concentrate in the center of the valleys and cannot model glacial erosion correctly.

**Cirques, horns, and ridges:** The top of the glacier progressively attacks the base of its supporting slope and carves steep cliffs, eventually reaching an angle where the ice can no longer cling. Other erosion processes, such as debris flow, take over, and the cliff retreats until it joins the other side. This results in large circular openings in the mountain called cirques. The connection between the two cirques leads to ridges and horns at the top (see Figure 8).

## 7.2 Evaluation and Comparisons

A direct numerical comparison of the fidelity of our erosion processes against the geomorphological reality of mountain ranges is infeasible. The geological time scales in play extend beyond current measurement capabilities and involve a complex mixture of interleaved phenomena, such as tectonics, climate, lithology, erosion, and ecosystems. Instead, we formulate a range of experiments to validate the different components of our method independently.

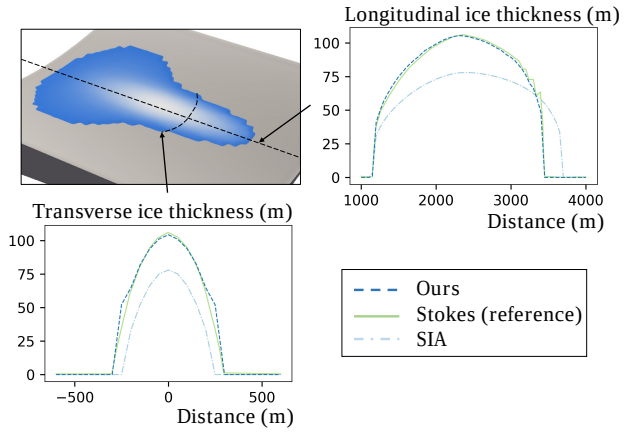



Fig. 10. A comparison of our approach against reference models from glaciology: We simulated a glacier on a sine-shaped terrain (top left), and compared the longitudinal (top right) and lateral (bottom left) profiles of ice thickness with the results from a reference Stokes solver [Le Meur et al. 2004], and the shallow ice approximation [Argudo et al. 2020]. Our results are similar to the Stokes reference, which validates our use of an efficient approximation in place of a more accurate solver.

*Ice flow prediction.* To begin, we assess how faithfully our approach matches Blatter's model (Eqn. 7). We disable erosion, fix constant climate conditions, and simulate the progress of a glacier toward a steady-state on a 32 m resolution terrain. Figure 9 shows a comparison of our learning method, a reference solution to Eqn. 7, and the Shallow Ice Approximation used by Argudo et al. [2020]. Our method has relatively minimal divergence from the reference

(achieving 93% of ice flow, 97% of ice thickness at steady-state), while the *SIA* predicts 49% less volume and evidence ice-flow acceleration along valley centers due to a lack of viscous stress.

Next, we exploit glaciology benchmarks to validate both the correctness of our implementation and the choice of Blatter's model over higher-order solutions to the Glen-Stokes equations. In Figure 10, we recreate a sine-shaped validation case due to Le Meur et al. [2004]. We reproduce the topography, simulate a glacier, and compare the lateral and longitudinal curves of ice thickness arising from our model, a Gen-Stokes solver [Le Meur et al. 2004], and the SIA. Results show a strong equivalence between our model and the Glen-Stokes solver, despite the absence of vertical velocity components in our model.

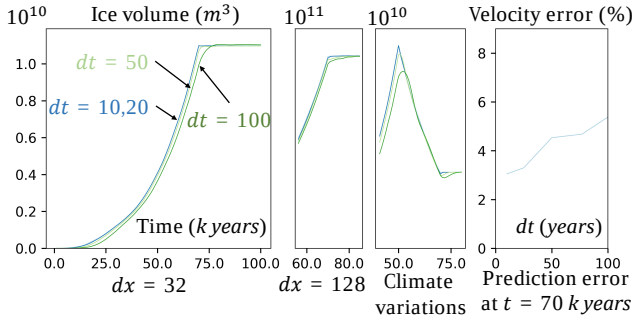



Fig. 11. Evaluation of our time-step scheme: ice volume for a typical cooling climate $dx = 32$ m (left), $dx = 128$ m (center left, with elevations scaled accordingly), and $dx = 32$ m with extreme temperature variation (center right). Time steps of 10 to 20 years are already enormous for glacier simulation at this resolution, yet they result in a low error (< 1%), while a time step of 100 years induces an error of about 10%. We further show the impact of the timestep on the accuracy of the velocity prediction (right).

*The accuracy of our time-step scheme.* Our time-step scheme allows time steps of $10 - 100$ years, thereby enabling the interlinking of ice flow and erosion. We evaluate its accuracy with the following experiment: On the $dx = 32$ m resolution terrain from Figure 9 we reproduce a typical cooling climate scenario, where the ELA drops from 1600 m to 900 m over 70,000 years, with the glacier subsequently stabilizing for another 30,000 years. The total volume of ice during this period is shown in Figure 11 (left) for time steps of 10, 20, 50, and 100 years. The figure validates our assumption that glaciers are close to equilibrium at the scale of climate evolution, with little variation in mass after the cooling period. It also supports our use of a time step of $dt = 100$ years, which corresponds to a loss in mass of 10.5% at $t = 70,000$ years (time steps of 10 or 20 years show errors below 1%).

We also include the result of an equivalent experiment for a $dx = 128$ m (middle left) where we multiplied all elevations and ELA by 4 to produce similar glaciers. Although this corresponds to an extreme drop in ELA of 3,800 m the error is still of the same order. We also pushed our system to its limits with a more aggressive climate change where the temperature decreases over 50,000 years, followed by a quick increase over 20,000 years to a final ELA of $1,100m$. This shows the limits of our method: when faced with overly strong variations in the climate, our time-step scheme favors the transport of low frequencies and over-predicts the glacier response.

The error in ice prediction depicted in Figure 11 (left) is likely a consequence of our adaptive time steps, but may also arise from prediction inaccuracies in the network. With a large time step, the network sees more variation in the ice geometry, which hinders convergence. To settle this, we stop the simulation at $t = 70,000$ years and plot in Figure 11 (right) the accuracy of the ice velocity prediction compared to the velocity obtained by minimizing Eqn. 7 directly. As expected, we see a dependency of the precision on the time step, yet with accuracy above 94% for $dt = 100$ years.

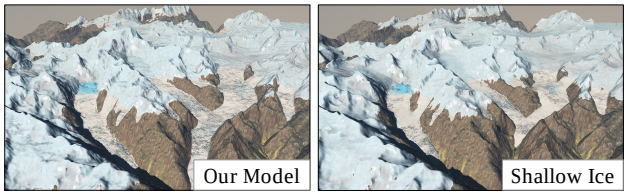



Fig. 12. Our model vs. the SIA for showing surface ice flow. Although the main glacier is similar, the SIA under-predicts glacial extent and our model is more strongly influenced by the valley floor.

*SIA and erosion.* In the absence of erosion, a simple visual inspection of the glacial ice envelope for our method and the SIA [Argudo et al. 2020] is not likely to favour either approach. In Figure 12 both methods are used to overlay glaciers on real topography from Vignemal in the Pyrenees and the only notable differences are the smaller ice extent of the SIA and a stronger correspondence with bedrock undulations for our model.

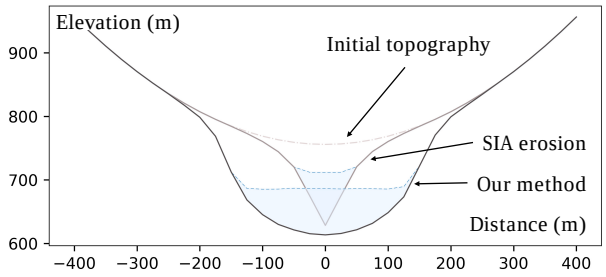



Fig. 13. We erode the terrain from Figure 9: while our method produces characteristic U-shaped valleys, the SIA carves deep narrow defiles that ultimately lead to numerical instability.

It is difficult to couple the SIA to glacial erosion. As observed by geologists [Egholm et al. 2012a], the lack of viscous stress in the SIA results in a narrowing of the principle ice velocities (Figure 9, right) and, therefore, of the eroded valleys, resulting in discontinuities in the eroded bedrock, which leads to numerical instability. We demonstrate this by eroding the toy terrain from Le Meur et al. [2004] (Figure 10) until the point of numerical divergence. While our method produces a typical U-shape, the erosion by SIA mainly erodes the valley base in an unrealistically narrow V-shaped pattern.

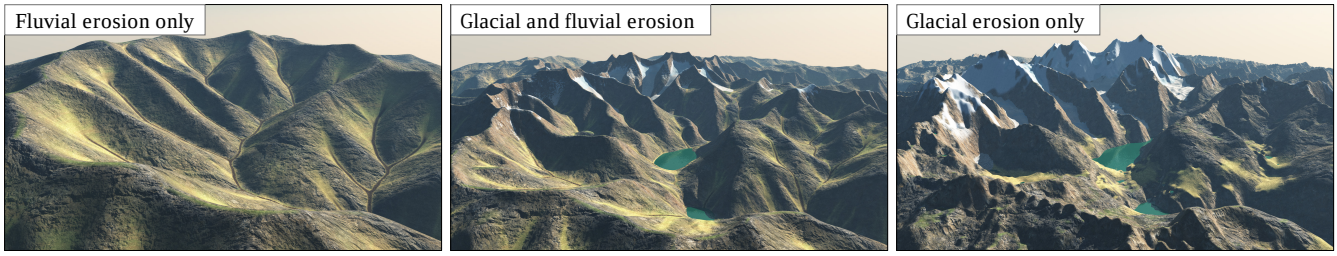



Fig. 14. While fluvial erosion on its own (left) lacks most of the main features from glacial erosion (right), it is still necessary to model the interplay between the two processes (center). Glacial erosion alone results in excessively steep cliffs, and hydraulically disconnected lakes and valleys.

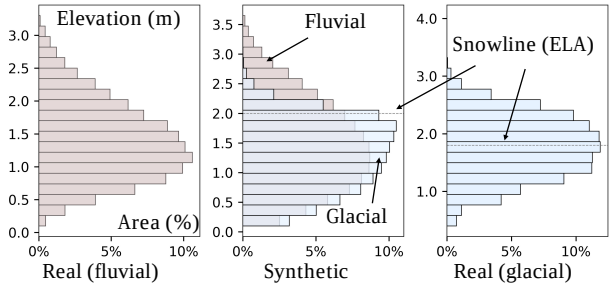



Fig. 15. The hypsometry (distribution of elevation) of a fluvial landscape (Sivalik Hills in Nepal, left), a glacial terrain (Vignemal, Pyrenees, right), and our synthetic terrain before and after glacial erosion. Note the hypsometric maximum just above the snowline (ELA) at the glacial maximum.

*Hypsometric analysis.* Geomorphological studies demonstrate that glacially eroded terrains have a characteristic hypsometric signature (distribution of elevations) that is distinct from fluvial landscapes [Egholm et al. 2009]. In particular, the hypsometric maximum (the most common elevation) is found just above the equilibrium line (ELA, or snowline) in the glacial case. Figure 15 compares the hypsometry of a real fluvial landscape (Sivalik Hills in Nepal, Figure 20), a terrain with clear indications of glacial erosion (Vignemal, Pyrenees, Figure 12), and our synthetic terrain before and after glacial erosion. We uniformly scaled the histograms to match the highest elevations for comparison. We note similarities in the hypsometric signature between real and synthetic fluvial landscapes, as well as the shift of the hypsometric maximum toward the ELA, as predicted by Egholm et al. [2009], in the synthetics glacial case.

## 7.3 Performance

Profiling on a $1024 \times 1024$ terrain shows that the ice flow takes 87% of the total 300 *ms* required to run one step, the ice transport 3% and mass balance and erosion, the remaining 10%. The computation of the drainage takes 1.7 *s* on the CPU (once every twenty GPU steps).

We compared the ice flow estimation with the direct optimization of Eqn. 7 (reference) and the Shallow Ice Approximation (SIA). The reference requires about 100 steps of 130*ms* to achieve similar accuracy: our method is 50× faster. The SIA needs only 4.7*ms* for one step, but the unfavorable stability conditions require smaller time steps (divided by 100 for a cell size of 32 m), even with our time-stepping algorithm, which leads to a similar runtime to reach a steady state. Overall, $\approx 2,000$ steps are required to produce the results presented in this article. Computations take 3.7 min at 512×512 resolution, 9.8 min at $1,024 \times 1,024$ and 37.5 min at $2,048 \times 2,048$ respectively. CPU drainage computation and GPU-CPU transfers represent almost 50% of the total computational time.

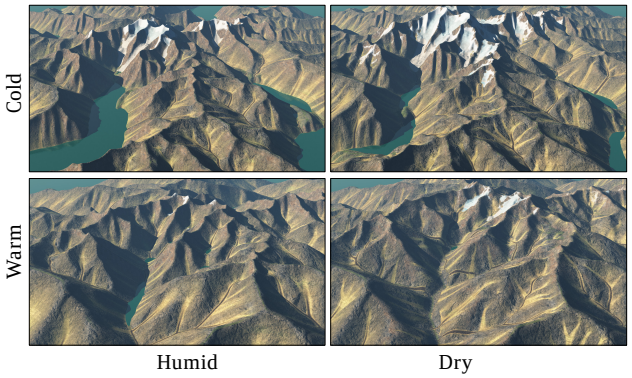



Fig. 16. Effect of the climate on erosion. While the difference in temperature controls the extent of the eroded valleys, the variations in precipitation affect the strength of the erosion patterns.

## 7.4 Control Parameters

We investigated the role of the different parameters of our model and their influence on the shape of the resulting terrain.

The first, most important parameter is the climate, encoded in the mass balance equation (Eqn. 2), as it dictates the overall shape and intensity of the glacier. In particular, the Equilibrium Line Altitude $E$ depends on the average temperature at sea level, which can vary by up to $10^o$C between glacial and interglacial periods in some locations. The choice of $E$ dictates the positioning of glaciers (the area above is fully covered in ice) and their volume, as decreasing $E$ increases the accumulation area. The precipitation, $\beta$, expresses the maximum amount of ice accumulating in the glacier each year and increases the glacier velocity. Figure 16 shows the difference in erosion patterns between warmer, colder, dryer, and more humid climates. In particular, colder glaciers carve to a greater extent than in temperate settings, and a high precipitation rate is associated with more pronounced erosion.

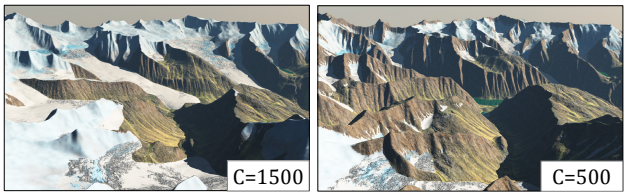



Fig. 17. Slow glaciers with high friction (left) produce larger, smoother valleys than their fast-flowing counterparts (right).

The sliding law of glaciers is poorly constrained and depends on several external factors, such as the subglacial hydrology, the temperature of the ice at the interface, and the concentration of sediments and loose rocks beneath the glacier. It is therefore natural to use sliding parameters as a proxy to express the large variety of glacier shapes. In Figure 17, we fix the sliding exponent to $\sigma = 3/2$ and vary the sliding coefficient between $C = 1,500\ Pa.m^{-\sigma}.y^{\sigma}$ (large friction, low sliding) and $500\ Pa.m^{-\sigma}.y^{\sigma}$ (fast-flowing glaciers). The basal velocity being smaller with higher friction, we multiplied by a factor of 4 the effects of glacier erosion on the case with $C = 1,500$ to produce comparable landscapes. We observe that the sliding coefficient impacts the width of the glacier and carved valleys: a higher $C$ results in wider, smoother and shallower valleys.

Another important factor is the balance between the two main modes of glacial erosion: abrasion and quarrying (Eqs. 6.1 and 6.1), through the parameters $k_e$ and $k_q$. By activating each of these effects independently, Figure 18 shows that the main effect of abrasion is to carve deep valleys where large glaciers reach their maximum extent. In contrast, quarrying is extremely efficient near the top of the glacier and results in flat, small valleys just below the ridges. By extension, the impact of these erosion factors is also seen in the distribution of lakes: abrasion carves lakes in the bottom of valleys, while quarrying produces small high-altitude perched lakes (see Figure 7).


Fig. 18. Terrains eroded with either glacial abrasion or quarrying in isolation. Quarrying erodes near the ridges and produces flatter, high-altitude valleys, while abrasion carves deep and large valleys at the glacier core.

## 7.5 Interaction with Fluvial Erosion

Mountain ranges are shaped by the interplay of fluvial and glacial erosion, with the dominant factor alternating between glacial and interglacial periods. In Section 6 we propose coupling glacial and fluvial processes, to 1) enable an end-to-end, automatic production of mountain ranges from initial fluvial morphology to current shapes with glacial features and 2) consistently model weathering of rocks exhumed from the ice after a glacier's retreat.

Figure 14 demonstrates that glacial erosion produces a landscape very different from fluvial erosion alone (left and center), but also that modeling glacial erosion only, with all other processes disabled, would result in an unnatural terrain: excessively steep cliffs should be quickly reduced by landslides and debris flow erosion, lakes should be connected by a river network and streams should carve openings on the edge of hanging valleys.

Figure 5 demonstrates the effect of the three phenomena that constitute fluvial erosion, by disabling each in turn. Without fluvial erosion (top right), we notice an absence of irregularity in the base of U-shaped valleys that would otherwise lead to connecting rivers and channels. Disabling debris flow (bottom left) shows its importance in carving the steepest slopes into dendritic gullies bordered by sharp ridges. Disabling hill-slope erosion (bottom right), essentially a diffusion process, produces noise and unrealistically sharp shifts between erosive streams at locations where the drainage is small.

## 7.6 Method Applicability

We have shown how our method can be applied to medium-scale terrains ($16 - 60$ km) initialized with fluvial erosion to provide synthetic conditions that closely match the real formation processes of glacial landforms. We demonstrate now that we can use our method for other applications.

Synthetic terrains are often modeled using a multi-scale noise sequence followed by enhancement based on erosion. We show that our method can introduce glacial features into such procedural terrains. Figure 19 shows the results of short-term glacial erosion on a noise-based terrain. Thus, in contrast to fluvial erosion [Galin et al. 2019], our method does not need a hydrologically consistent input terrain to operate.

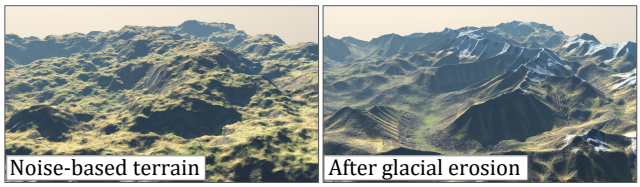



Fig. 19. Glacial erosion (right) applied to a procedural noise-based terrain.

We also show that our method produces believable erosion patterns on real terrains, which opens perspectives for scientific communication or education. We use a 30 m resolution NASA Digital Elevation Model [JPL 2020] of the Sivalik Hills in Nepal. We chose this area from a young mountain range in warm climatic conditions to ensure that the initial terrain was not affected by glacial erosion (Figure 20, left). We show the maximal extent of glaciation (Figure 20, center) and the resulting eroded terrain (Figure 20, right).

Finally, our method can be applied on a much larger scale, for example, to generate fjords. Fjords emerge in conditions very different from mountain glaciers: they are initially buried under very large ice caps, about $1,000$ m thick, and do not need an initial steep topography to appear, as shown by Egholm et al. [2017]. We reproduce this behavior on a $130 \times 130$ km$^2$ terrain with a grid spacing of $256m$. We initialize the terrain's fluvially eroded topography with

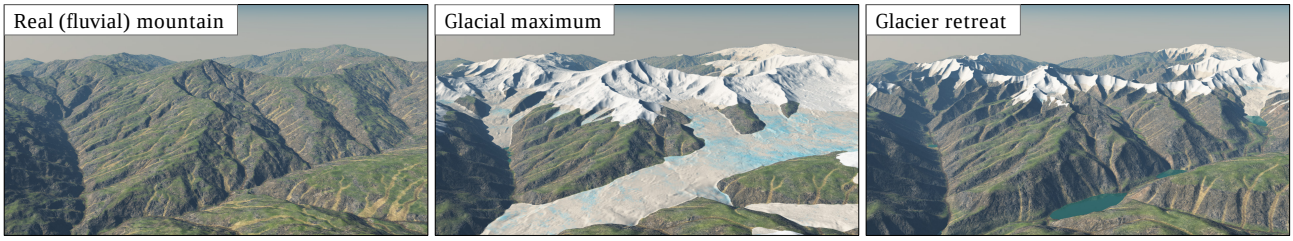



Fig. 20. Real terrain from the Sivalik Hills in Nepal, mostly carved by fluvial erosion (left). After a cooling event (middle), the glaciers retreat and leave deep eroded valleys (right).

a gentle north slope of about 1% (Figure 21, left), and we cover it with 1,000 m of ice. We show the retreat of the ice cap (Figure 21, center) and the carved fjords, a result consistent with the patterns observed in [Egholm et al. 2017].

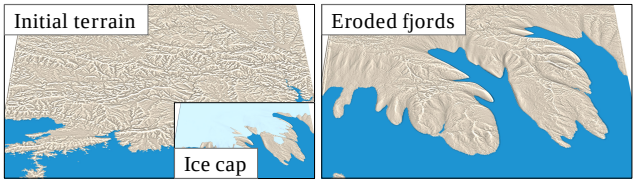



Fig. 21. From a mostly flat topography (left), covered by $1,000m$ of ice, ice streams progressively emerge and carve deep fjords (right). We show an intermediate state during the retreat of the ice cap in the middle.

### 7.7 Limitations

In terms of limitations, our method relies on a physically-based simulation and generally does not provide the direct and intuitive control that would be needed for authoring applications, particularly in the entertainment industry. Furthermore, although we use laws from glaciology and geomorphology, our evaluation is mostly feature-based. A more thorough study would be needed to confirm that the quantitative properties of the eroded features match real ones. Maybe more importantly, such a study would allow us to specify which parameter settings are needed to achieve a specific target, for example, to mimic specific real mountain ranges.

Simulation of natural phenomena is difficult to validate in general. However, more components would be required if our method was to be used in geomorphology: glaciology simulations often couple the glacier with a thermal model [Gilbert et al. 2014], and we do not consider the complexities of sub-glacial hydrology [Egholm et al. 2012b; Herman et al. 2011], which has a significant impact on erosion.

Finally, our simulation does not model the transport of eroded rocks and sediments. However, the latter may accelerate or dampen the erosion [Ugelvig and Egholm 2018] and give rise to specific visual features such as moraines.

## 8 CONCLUSION

We have introduced a novel algorithm for the efficient simulation of glacial erosion combined with secondary erosive phenomena: fluvial, hillslope, and cliff-based debris flow erosion. Our glacial erosion allows for the simulation of characteristic glacial landforms, such as hanging and U-shaped valleys, cliffs, and glacial lakes. It handles periods covering several glacial and interglacial cycles. The key idea is to accelerate ice flow evaluation and couple this with an adaptive ice transport solver to deal with the time and space variability inherent in glacial erosion. Our method generates large to medium-scale features in landscapes formed by glaciers.

Our large-scale model captures many significant glacial landforms, but it does not model smaller-scale features, such as moraines and drumlins. Future work could address this by combining our algorithms with small-scale, physically-based, or procedural models. Rock abrasion in our model could be used to parameterize the deposition of transported rocks into moraines. The sedimentation of debris flowing from cliffs into glacial lakes could also be integrated.

## ACKNOWLEDGMENTS

This project was sponsored by the Agence Nationale de la Recherche project Invterra ANR-22-CE33-0012-01 to Cordonnier, by the USDA NIFA, Award #2023-68012-38992 grant “Promoting Economic Resilience and Sustainability of the Eastern U.S. Forests” to Benes, by Hi!Paris’s fellowship on Creative AI to Cani, by the National Research Foundation of South Africa (Grant Number 129257) to Gain, and the Agence Nationale de la Recherche project Ampli ANR-20-CE23-0001.

## REFERENCES

Oscar Argudo, Eric Galin, Adrien Peytavie, Axel Paris, and Eric Guérin. 2020. Simulation, Modeling and Authoring of Glaciers. *ACM Transactions on Graphics* 39, 6, Article 177 (2020), 14 pages.

Richard Barnes, Clarence Lehman, and David Mulla. 2014. Priority-flood: An optimal depression-filling and watershed-labeling algorithm for digital elevation models. *Computers and Geosciences* 62 (2014), 117–127.

Matthew Beardall, McKay Farley, David Ouderkirk, Cory Reimschussel, Smith J., Michael Jones, and Parris Karl Egbert. 2007. Goblins by Spheroidal Weathering. In *Proceedings of the 3rd Eurographics Conference on Natural Phenomena*. The Eurographics Association, Goslar, DEU, 7–14.

Flavien Beaud, Gwenn E. Flowers, and Sam Pimentel. 2014. Seasonal-scale abrasion and quarrying patterns from a two-dimensional ice-flow model coupled to distributed and channelized subglacial drainage. *Geomorphology* 219 (2014), 176–191.

Bedrich Benes, Václav Těšínskỳ, Jan Hornyš, and Sanjiv K Bhatia. 2006. Hydraulic erosion. *Computer Animation and Virtual Worlds* 17, 2 (2006), 99–108.

Douglas I. Benn and David J. A. Evans. 2014. *Glaciers & glaciation*. Routledge.

Hallet Bernard. 1979. A theoretical model of glacial abrasion. *Journal of Glaciology* 23, 89 (1979), 39–50.

Heinz Blatter. 1995. Velocity and stress fields in grounded glaciers: a simple algorithm for including deviatoric stress gradients. *Journal of Glaciology* 41, 138 (1995), 333–344.

Jean Braun and Malcolm Sambridge. 1997. Modelling landscape evolution on geological time scales: a new method based on irregular spatial discretization. *Basin Research*

9, 1 (1997), 27–52.
Jean Braun and Sean D. Willett. 2013. A very efficient O(n), implicit and parallel method to solve the stream power equation governing fluvial incision and landscape evolution. *Geomorphology* 180-181 (2013), 170–179.
Jean Braun, Dan Zwartz, and Jonathan H. Tomkin. 1999. A new surface-processes model combining glacial and fluvial erosion. *Annals of Glaciology* 28 (1999), 282–290.
Guillaume Cordonnier, Benoit Bovy, and Jean Braun. 2019. A versatile, linear complexity algorithm for flow routing in topographies with depressions. *Earth Surface Dynamics* 7, 2 (2019), 549–562.
Guillaume Cordonnier, Jean Braun, Marie-Paule Cani, Bedrich Benes, Éric Galin, Adrien Peytavie, and Éric Guérin. 2016. Large Scale Terrain Generation from Tectonic Uplift and Fluvial Erosion. *Computer Graphics Forum* 35, 2 (2016), 165–175.
Guillaume Cordonnier, Marie-Paule Cani, Bedrich Benes, Jean Braun, and Eric Galin. 2017a. Sculpting Mountains: Interactive Terrain Modeling Based on Subsurface Geology. *IEEE Transactions on Visualization and Computer Graphics* 24, 5 (2017), 1756 – 1769.
Guillaume Cordonnier, Eric Galin, James Gain, Bedrich Benes, Eric Guérin, Adrien Peytavie, and Marie-Paule Cani. 2017b. Authoring Landscapes by Combining Ecosystem and Terrain Erosion Simulation. *ACM Transactions on Graphics* 36, 4, Article 134 (2017), 12 pages.
Tiangang Cui, Zhongjian Wang, and Zhiwen Zhang. 2022. A variational neural network approach for glacier modelling with nonlinear rheology. arXiv:2209.02088
Giliam J.P. de Carpentier and Rafael Bidarra. 2009. Interactive GPU-based procedural heightfield brushes. In *Proceedings of the 4th International Conference on Foundations of Digital Games*. Association for Computing Machinery, New York, NY, USA, 55–62.
David L. Egholm, John D. Jansen, Christian F. Brædstrup, Vivi K. Pedersen, Jane L. Andersen, Sofie V. Ugelvig, Nicolaj K. Larsen, and Mads F. Knudsen. 2017. Formation of plateau landscapes on glaciated continental margins. *Nature Geoscience* 10, 8 (2017), 592–597.
David L. Egholm, Mads F. Knudsen, Chris D. Clark, and Jerome E. Lesemann. 2011. Modeling the flow of glaciers in steep terrains: The integrated second-order shallow ice approximation (iSOSIA). *Journal of Geophysical Research: Earth Surface* 116, F2 (2011).
David L. Egholm, S. B. Nielsen, Vivi K. Pedersen, and J-E Lesemann. 2009. Glacial effects limiting mountain height. *Nature* 460, 7257 (2009), 884–887.
David L. Egholm, Vivi K. Pedersen, M.F. Knudsen, and N.K. Larsen. 2012a. On the importance of higher order ice dynamics for glacial landscape evolution. *Geomorphology* 141-142 (2012), 67–80.
David L. Egholm, Vivi K. Pedersen, Mads F. Knudsen, and Nicolaj K. Larsen. 2012b. Coupling the flow of ice, water, and sediment in a glacial landscape evolution model. *Geomorphology* 141-142 (2012), 47–66.
Kai Franke and Heinrich Müller. 2022. Procedural generation of 3D karst caves with speleothems. *Computers & Graphics* 102 (2022), 533–545.
Eric Galin, Eric Guérin, Adrien Peytavie, Guillaume Cordonnier, Marie-Paule Cani, Bedrich Benes, and James Gain. 2019. A Review of Digital Terrain Modeling. *Computer Graphics Forum* 38, 2 (2019), 553–577.
A. Gilbert, O. Gagliardini, C. Vincent, and P. Wagnon. 2014. A 3-D thermal regime model suitable for cold accumulation zones of polythermal mountain glaciers. *Journal of Geophysical Research: Earth Surface* 119, 9 (2014), 1876–1893.
John W Glen. 1955. The creep of polycrystalline ice. *Proceedings of the Royal Society A* 228, 1175 (1955), 519–538.
Eric Guérin, Julie Digne, Adrien Peytavie, and Eric Galin. 2016. Sparse representation of terrains for procedural modeling. *Computer Graphics Forum* 35, 2 (2016), 177–187.
Eric Guérin, Adrien Peytavie, Simon Masnou, Julie Digne, Basile Sauvage, James Gain, and Eric Galin. 2022. Gradient Terrain Authoring. *Computer Graphics Forum* 41, 2 (2022), 85–95.
Jonathan M Harbor, Bernard Hallet, and Charles F Raymond. 1988. A numerical model of landform development by glacial erosion. *Nature* 333, 6171 (1988), 347.
Rachel M. Headley, Gerard Roe, and Bernard Hallet. 2012. Glacier longitudinal profiles in regions of active uplift. *Earth and Planetary Science Letters* 317-318 (2012), 354 – 362.
Frédéric Herman, Flavien Beaud, Jean-Daniel Champagnac, Jean-Michel Lemieux, and Pietro Sternai. 2011. Glacial hydrology and erosion patterns: a mechanism for carving glacial valleys. *Earth and Planetary Science Letters* 310, 3-4 (2011), 498–508.
Frédéric Herman, Olivier Beyssac, Mattia Brughelli, Stuart N Lane, Sébastien Leprince, Thierry Adatte, Jiao YY Lin, Jean-Philippe Avouac, and Simon C Cox. 2015. Erosion by an Alpine glacier. *Science* 350, 6257 (2015), 193–195.
Guillaume Jouvet. 2016. Mechanical error estimators for shallow ice flow models. *Journal of Fluid Mechanics* 807 (2016), 40–61.
Guillaume Jouvet, Guillaume Cordonnier, Byungsoo Kim, Martin Lüthi, Andreas Vieli, and Andy Aschwanden. 2022. Deep learning speeds up ice flow modelling by several orders of magnitude. *Journal of Glaciology* 68, 270 (2022), 651–664.
NASA JPL. 2020. NASADEM Merged DEM Global 1 arc second V001 [Data set]. NASA EOSDIS Land Processes DAAC. Accessed 2020-12-30.
Peter Krištof, Bedrich Benes, Jaroslav Křivánek, and Ondřej Šťava. 2009. Hydraulic Erosion Using Smoothed Particle Hydrodynamics. *Computer Graphics Forum (Proceedings of Eurographics 2009)* 28, 2 (mar 2009), 219–228.
Emmanuel Le Meur, Olivier Gagliardini, Thomas Zwinger, and Juha Ruokolainen. 2004. Glacier flow modelling: a comparison of the Shallow Ice Approximation and the full-Stokes solution. *Comptes Rendus Physique* 5, 7 (2004), 709–722.
Sylvain Lefebvre and Hugues Hoppe. 2005. Parallel Controllable Texture Synthesis. In *ACM SIGGRAPH* (Los Angeles, California) *(SIGGRAPH'05)*. Association for Computing Machinery, New York, NY, USA, 777–786.
Wei Li, Martin Z Bazant, and Juner Zhu. 2021. A physics-guided neural network framework for elastic plates: Comparison of governing equations-based and energy-based approaches. *Computer Methods in Applied Mechanics and Engineering* 383 (2021), 113933.
Fabio Magrani, Pierre G Valla, and David Egholm. 2022. Modelling alpine glacier geometry and subglacial erosion patterns in response to contrasting climatic forcing. *Earth Surface Processes and Landforms* 47, 4 (2022), 1054–1072.
Miriam Mahaffy. 1976. A three-dimensional numerical model of ice sheets: Tests on the Barnes Ice Cap, Northwest Territories. *Journal of Geophysical Research* 81, 6 (1976), 1059–1066.
Forest Kenton Musgrave, Craig E. Kolb, and Robert S. Mace. 1989. The synthesis and rendering of eroded fractal terrains. *ACM SIGGRAPH Computer Graphics* 23, 3 (1989), 41–50.
Axel Paris, Eric Galin, Adrien Peytavie, Eric Guérin, and James Gain. 2019. Terrain Amplification with Implicit 3D Features. *ACM Transactions on Graphics* 38, 5 (2019), 147:1–147:15.
Axel Paris, Eric Guérin, Adrien Peytavie, Pauline Collon, and Eric Galin. 2021. Synthesizing Geologically Coherent Cave Networks. *Computer Graphics Forum* 40, 7 (2021), 277–287.
Adrien Peytavie, Eric Galin, Jérôme Grosjean, and Stéphane Mérillou. 2009. Arches: a Framework for Modeling Complex Terrains. *Computer Graphics Forum* 28, 2 (2009), 457–467.
Maziar Raissi, Paris Perdikaris, and George E Karniadakis. 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics* 378 (2019), 686–707.
Hugo Schott, Axel Paris, Lucie Fournier, Eric Guérin, and Eric Galin. 2023. Large-scale terrain authoring through interactive erosion simulation. *ACM Transactions on Graphics* 42 (2023), 15 pages.
Pietro Sternai, Frédéric Herman, Pierre G Valla, and Jean-Daniel Champagnac. 2013. Spatial and temporal variations of glacial erosion in the Rhône valley (Swiss Alps): Insights from numerical modeling. *Earth and Planetary Science Letters* 368 (2013), 119–131.
J. Stock and William E. Dietrich. 2003. Valley incision by debris flows: Evidence of a topographic signature. *Water Resources Research* 39, 4 (2003).
Harald Svensson. 1959. Is the cross-section of a glacial valley a parabola? *Journal of Glaciology* 3, 25 (1959), 362–363.
Sofie V. Ugelvig and David L. Egholm. 2018. The influence of basal-ice debris on patterns and rates of glacial erosion. *Earth and Planetary Science Letters* 490 (2018), 110–121.
Eugene L. Wachspress and G. J. Habetler. 1960. An alternating-direction-implicit iteration technique. *J. Soc. Indust. Appl. Math.* 8, 2 (1960), 403–423.
J. Weertman. 1957. On the Sliding of Glaciers. *Journal of Glaciology* 3, 21 (1957), 33–38.
Howard Zhou, Jie Sun, Greg Turk, and James M. Rehg. 2007. Terrain Synthesis from Digital Elevation Models. *IEEE Transactions on Visualization and Computer Graphics* 13, 4 (2007), 834–848.

## APPENDIX A: VELOCITY GRADIENTS

The evaluation of Eqn.7 in Section 4 requires the computation of the areas of the lattice cell areas and the gradients of the velocity. We assume that the velocities follow piecewise trilinear functions inside the cell, and we evaluate their values and gradients at the center of the cells. For a cell at indices $i, j, k$ in the lattice, We note $\mathbf{u}(i-\frac{1}{2}, j-\frac{1}{2}, k-\frac{1}{2})$ the (horizontal) velocity stored at the corresponding corner of the cell. Similarly, $\zeta$ denotes the vertical position of the cell corner.

The values at the center of the cells are obtained by either differentiation or averaging of the values in the corner, in all $x$, $y$, and $z$ directions. We note $a_x$ and $d_x$ the corresponding averaging and difference operators here in the $x$ direction:

$$(d_x \circ \mathbf{u})(i, \cdot, \cdot)) = \mathbf{u}(i+\frac{1}{2}, \cdot, \cdot) - \mathbf{u}(i-\frac{1}{2}, \cdot, \cdot).$$

Recall that $dx$ denotes the regular grid spacing. We obtain the average cell thickness $c_t$ and the inter-cell face area $f_a$:

$$c_t = a_x \circ a_y \circ d_z \circ \zeta$$

$$f_a = \sqrt{dx^2 + \left((a_x \circ d_y \circ \zeta)\right)^2}\sqrt{dx^2 + \left((a_y \circ d_x \circ \zeta)\right)^2},$$

The cell volume $c_v$ is:

$$c_v = (a_z \circ f_a)\, c_t.$$

The velocity in the cell is given by:

$$\mathbf{u}_{\text{cell}} = a_x \circ a_y \circ a_z \circ \mathbf{u}.$$

The vertical gradient of the velocity is:

$$\partial_z \mathbf{u} = \frac{a_x \circ a_y \circ d_z \circ \mathbf{u}}{c_t}.$$

The horizontal gradients, however, need to account for the deformation of the lattice:

$$\partial_x \mathbf{u} = \frac{d_x \circ a_y \circ a_z \circ \mathbf{u}}{dx} + \frac{(a_x \circ a_y \circ \mathbf{u}_b)(d_x \circ a_y \circ \zeta_t) - (a_x \circ a_y \circ \mathbf{u}_t)(d_x \circ a_y \circ \zeta_b)}{dx\, c_t} + \frac{(d_x \circ a_y \circ d_z \circ \zeta)\mathbf{u}_{\text{cell}}}{dx\, c_t}, \quad (9)$$

where $\mathbf{u}_b$ and $\mathbf{u}_t$ are the velocity at the grid corner at the top and bottom of the cell: $\mathbf{u}_t(\cdot,\cdot,k) = \mathbf{u}(\cdot,\cdot,k+\frac{1}{2})$, $\mathbf{u}_b(\cdot,\cdot,k) = \mathbf{u}(\cdot,\cdot,k-\frac{1}{2})$, and similarly for $\zeta_b$ and $\zeta_t$.

The gradient in the $y$ direction is symmetric:

$$\partial_y \mathbf{u} = \frac{a_x \circ d_y \circ a_z \circ \mathbf{u}}{dx} + \frac{(a_x \circ a_y \circ \mathbf{u}_b)(a_x \circ d_y \circ \zeta_t) - (a_x \circ a_y \circ \mathbf{u}_t)(a_x \circ d_y \circ \zeta_b)}{dx\, c_t} + \frac{(a_x \circ d_y \circ d_z \circ \zeta)\mathbf{u}_{\text{cell}}}{dx\, c_t}. \quad (10)$$

## APPENDIX B: NOTATION AND DEFAULT VALUES

Table 1 details the default value and units of all parameters used in the paper. For erosion parameters, we show the range of values that allow us to simulate the various erosion resistance of various rock types.

The sliding coefficient $C$ is allowed to take values up to $\infty$. In practice, $\infty$ represents a frozen boundary condition (no sliding), and we parameterize the network with the inverse of $C$ to keep values in a reasonable range.

Table 1. Parameters

| | | Name | Value | Unit |
|---|---|---|---|---|
| Glaciers | Climate | $\beta$ | $1.0\,10^{-2}$ | $y^{-1}$ |
| | | $p_{\max}$ | 2 | $m\,y^{-1}$ |
| | | $E$ | $5.0\,10^{2} - 2.0\,10^{3}$ | $m$ |
| | Ice flow | $B$ | $2.9\,10^{5}$ | $Pa\,y^{1/3}$ |
| | | $C$ | $5.0\,10^{2} - \infty$ | $Pa\,m^{-\sigma}\,y^{\sigma}$ |
| | | $p$ | $4/3$ | |
| | | $\sigma$ | $3/2$ | |
| | | $g$ | 9.8 | $m\,s^{-2}$ |
| | | $\rho$ | 917 | $kg\,m^{-3}$ |
| Erosion | Glacial | $l$ | 2.02 | |
| | | $k_e$ | $1.3\,10^{-7} - 1.1\,10^{-6}$ | $m^{1-l}\,y^{l-1}$ |
| | | $k_q$ | $3.2\,10^{-5} - 2.6\,10^{-4}$ | 1 |
| | Fluvial | $k_f$ | $5\,10^{-6} - 1.5\,10^{-6}$ | $m^{2m-1}\,y^{-1}$ |
| | | $m$ | 0.4 | |
| | | $n$ | 1.0 | |
| | Hill slope, debris | $k_h$ | $1\,10^{-2} - 5\,10^{-2}$ | $m^{2}\,y^{-1}$ |
| | | $k_{df}$ | $10^{-4} - 2.5\,10^{-4}$ | $m^{2q-1}\,y^{-1}$ |
| | | $k_{da}$ | 10 | $m^{-2q}$ |
| | | $q$ | 0.8 | |
| | | $q'$ | 2.0 | |
| | | $s_c$ | 37 | Degrees |