The Utrecht Finite Volume Ice-Sheet Model: UFEMISM (version 1.0)

Constantijn J. Berends[1], Heiko Gölzer[1,2,*], Roderik S. W. van de Wal[1,3]

[1] Institute for Marine and Atmospheric research Utrecht, Utrecht University, Utrecht, 3584 CC, The Netherlands

[2] Laboratoire de Glaciologie, Université Libre de Bruxelles, Brussels, Belgium

5 [3] Faculty of Geosciences, Department of Physical Geography, Utrecht University, Utrecht, The Netherlands

[*] Now at: NORCE Norwegian Research Centre, Bjerknes Centre for Climate Research, Bergen, Norway

Correspondence to: Constantijn J. Berends (c.j.berends@uu.nl)

Abstract. Improving our confidence in future projections of sea-level rise requires models that can simulate ice-sheet evolution both in the future and in the geological past. A physically accurate treatment of large changes in ice-sheet geometry requires a

10 proper treatment of processes near the margin, like grounding line dynamics, which in turn requires a high spatial resolution in that specific region. This leads to a demand for computationally efficient models, where such a high resolution can be feasibly applied in simulations of $10^5 - 10^7$ yr in duration. Here, we present and evaluate a new ice-sheet model that solves the SIA and SSA approximations of the stress balance on a fully adaptive, unstructured triangular mesh. This strongly reduces the number of grid points where the equations need to be solved, increasing the computational efficiency. We show that the model

15 reproduces the analytical solutions or model intercomparison benchmarks for a number of schematic ice-sheet configurations, indicating that the numerical approach is valid. Because of the unstructured triangular mesh, the number of vertices increases less rapidly with resolution than in a square-grid model, greatly reducing the required computation time for high resolutions. A simulation of all four continental ice sheets during an entire 120 kyr glacial cycle, with a 4 km resolution near the grounding line, is expected to take $100 - 200$ wall clock hours on a 16-core system ($1,600 - 3,200$ core hours), implying that this model

20 can be feasibly used for high-resolution paleo-ice-sheet simulations.

1 Introduction

The response of the Greenland and Antarctic ice sheets to the warming climate forms the largest uncertainty in long-term sea-level projections (e.g. Oppenheimer et al., 2019; van de Wal et al., 2019). Since the dynamical evolution of ice sheets has components that are slow compared to the human time scale, observational evidence alone cannot sufficiently reduce this

25 uncertainty. Instead, reconstructions of the evolution of ice sheets during the geological past are required to improve our understanding of the long-term evolution of these systems and the constraints this provides for future ice-sheet retreat. Recent work has focused on using ice-sheet models and climate models, with varying degrees of inter-model coupling, to reproduce different periods of the geological past, with climates that have been both significantly warmer and colder than the present (e.g. Abe-Ouchi et al., 2013; Pollard et al., 2013; DeConto and Pollard, 2016; Stap et al., 2017; Berends et al., 2018, 2019;

30 Willeit et al., 2019). These studies have highlighted and reemphasized the importance of understanding, and properly

modelling, the different physical interactions between the ice sheets, sea level, the solid Earth, and the regional and global climate. Since many of these processes become relevant only when significant changes in ice-sheet geometry occur, studying them requires very long ($10^5 - 10^7$ yr) ice-sheet model simulations.

5   The dynamics of the Antarctic ice sheet at present are strongly influenced by the presence of floating ice shelves (Pattyn, 2018). Different studies have investigated the physical processes affecting these ice shelves, including surface melt induced by atmospheric processes (Bevan et al., 2017; Kuipers Munneke et al., 2018), bottom melt induced by intrusion of warm ocean water (Depoorter et al., 2013; Lazeroms et al., 2018, Reese et al., 2018), brittle fracturing of ice cliffs (Pollard et al., 2015), and the response of the grounding line to changes in sea level and bedrock elevation (Gomez et al., 2013; Barletta et al., 2018).

10 Ice dynamics around the grounding line have been of particular interest, demonstrating the need for a high model resolution (< 1 km) in order to accurately resolve the physical processes involved (Schoof, 2007; Gladstone et al., 2012; Pattyn et al., 2012, 2013). Different approaches have been proposed where semi-analytical solutions (Schoof, 2007; Tsai et al., 2015) are implemented as boundary conditions in numerical models, to maintain physical accuracy at lower resolutions (Pollard & DeConto, 2012; Pattyn, 2017). However, while these approaches achieve good results in situations without buttressing (Pattyn

15 et al., 2013), their performance in simulating realistic ice shelves, where buttressing is usually a significant factor, is poor (Reese et al., 2018).

Ice-sheet model computation time increases rapidly with model resolution, due to the increasing number of grid points, the decreasing time step required for numerical stability, and the increasing number of topographic features that are resolved. This

20 means that the need for both a high model resolution and long simulations results in a computational paradox. Most research groups working on paleo-ice-sheet simulations consider the uncertainties in paleoclimate reconstructions used as forcing to be much larger than the physical inaccuracy resulting from a low model resolution. The ice-sheet models used by these groups therefore typically have a low-resolution, fixed grid, and solve a simplified version of the Navier-Stokes equations, which makes them very computationally efficient, and relatively easy to compile, run, and modify (e.g. SICOPOLIS; Greve et al.,

25 2011; PISM; Winkelmann et al., 2011; ANICE; de Boer et al., 2014; f.ETISh; Pattyn, 2017; GRISLI; Quiquet et al., 2018; CISM; Lipscomb et al., 2019; Yelmo; Robinson et al., 2020). Paleo-simulations of the Antarctic ice sheet with such models have used resolutions of e.g. 10 km (DeConto and Pollard, 2016), 32 km (Robinson et al., 2020), 40 km (Berends et al., 2019), 80 km (Willeit et al., 2019) or 110 km (Abe-Ouchi et al., 2013), too low to properly capture grounding line dynamics. A few existing models, which are mainly intended for relatively short (10^1-10^3 yr) simulations, use high-resolution, adaptive grids

30 (e.g. ISSM; Larour et al., 2012; Elmer/Ice; Gagliardini et al., 2013; MALI; Hoffman et al., 2018). They often solve for more terms in the Navier-Stokes equations (either the higher-order Blatter-Pattyn [Pattyn, 2003] approximation, or even the full Stokes system) using finite element methods, making them more physically accurate. While recent developments have improved their computational efficiency enough to make small-scale (single ice-sheet basin, 10^4 yr) paleo-simulations feasible

(Cuzzone et al., 2018, 2019), they tend to be too computationally demanding for the 10^5-10^7 yr simulations needed for paleo-ice-sheet simulations.

Here, we present and evaluate a new ice-sheet model that constitutes an optimum between these two families: the Utrecht
5 Finite Volume Ice-Sheet Model (UFEMISM). It combines the SIA and SSA simplifications of the stress balance, used in most paleo-ice-sheet models, with a variable-resolution unstructured grid, which allows it to achieve a high (< 5 km) resolution near the grounding line, while retaining the computational speed required for feasibly performing long paleo-ice-sheet simulations. This makes it especially useful for studying the impact of ice-sheet – solid Earth – sea-level interactions on long-term ice-sheet evolution. In Sect. 2, we provide a brief description of the physical equations for ice dynamics and thermodynamics that are
10 solved by the model, as well as a description of the unstructured grid upon which those equations are solved. In Sect. 3, we present results from a number of benchmark experiments performed with UFEMISM, showing that the model output agrees well with different analytical solutions, as well as with output from other ice-sheet models. In Sect. 4, we present results from a detailed analysis of the computational performance of the model.

2 Model description

15 ### 2.1 Overview

UFEMISM is a variable-resolution ice-sheet model. Flow velocities for grounded ice are calculated using the shallow ice approximation (SIA; Morland and Johnson, 1980), while the shallow shelf approximation (SSA; Morland, 1987) is used both for calculating flow velocities for floating ice, and as a "sliding law" for grounded ice, using the hybrid approach by Bueler and Brown (2009). These equations are discretised and solved on an unstructured mesh, which is generated and updated based
20 on the modelled ice-sheet geometry. The resolution can vary from as coarse as 200 km over open ocean or 100 km over the interior of a large ice-sheet, to as fine as 1 km at the grounding line, with the precise numbers specified at run-time through a config file. Using a finite volume approach, ice velocities and fluxes are calculated on cell boundaries, similar to the "staggered" Arakawa C grid used in many ice models based on finite differences. By explicitly calculating the mass of ice moved between individual vertices in every time step, this approach guarantees conservation of mass. The model is
25 thermomechanically coupled, solving for the diffusion and advection of heat, which enters the ice sheet through the surface and base, as well as through strain heating. For this purpose, the ice sheet is divided vertically into 15 unequally spaced layers, decreasing in thickness near the base, where most of the deformation takes place. The resulting three-dimensional englacial temperature field is used to determine the ice viscosity.

2.2 Unstructured triangular mesh

30 The main distinguishing feature of UFEMISM with respect to many other paleo-ice-sheet models is the unstructured triangular mesh on which the physical equations are solved. The mesh generation code in UFEMISM is based on Ruppert's Algorithm

(Ruppert, 1995), which is used to iteratively add vertices to a pre-existing Delaunay triangulation (a process called "refinement"), until the smallest internal angle of all triangles no longer lies below a pre-defined threshold value (typically 25 degrees). Since the stability of many numerical iterative schemes for solving differential equations strongly depends on this particular property of the mesh geometry (Ruppert, 1995), Ruppert's Algorithm is a very useful tool for generating meshes
5   suitable for use in physical models.

The mesh generation code in UFEMISM uses an extended version of Ruppert's algorithm. In the original version, a triangle is marked as "bad" if its smallest internal angle lies below the threshold value. If this is the case, the triangle is "split", by adding a vertex at the triangle's circumcentre, and updating the Delaunay triangulation. Ruppert (1995) showed that, as long
10   as the perimeter of the mesh does not contain any angles sharper than the threshold value, this algorithm always converges, i.e. results in a mesh with no "bad" triangles and a finite, typically small number of vertices. In UFEMISM, several additional conditions have been added which can cause a triangle to be marked as "bad", which depend upon the triangle's "model content" and size. If, for example, the modelled grounding line passes through a triangle whose longest edge exceeds the specified maximum grounding line resolution, that triangle is also marked as "bad". Extensive preliminary experiments showed
15   that this particular approach to mesh generation is robust, resulting in a mesh with resolutions over specified areas (such as the grounding line) that lie below, but not too far below (rarely more than 25 %), the specified maximum value. Other conditions that can be included, but will not be discussed here, include a maximum vertical error in surface elevation based on surface curvature (useful for the interior close to the ice margin), a maximum resolution derived from ice velocity (useful for ice streams). Lastly, a maximum resolution can be prescribed at specific geographic locations, such as ice core drill sites. Shown
20   in Fig. 1 are four meshes generated for both Antarctica and Greenland, based on present-day ice-sheet geometry (BedMachine Greenland version 3, Morlighem et al., 2017, and BedMachine Antarctica version 1, Morlighem et al., 2019), with maximum ice-margin (including both the grounding line and the calving front) resolutions of 100, 30, 10 and 3 km, respectively. For Antarctica, two high-resolution locations are included: one at the South Pole, and one at the EPICA Dome C drill site.
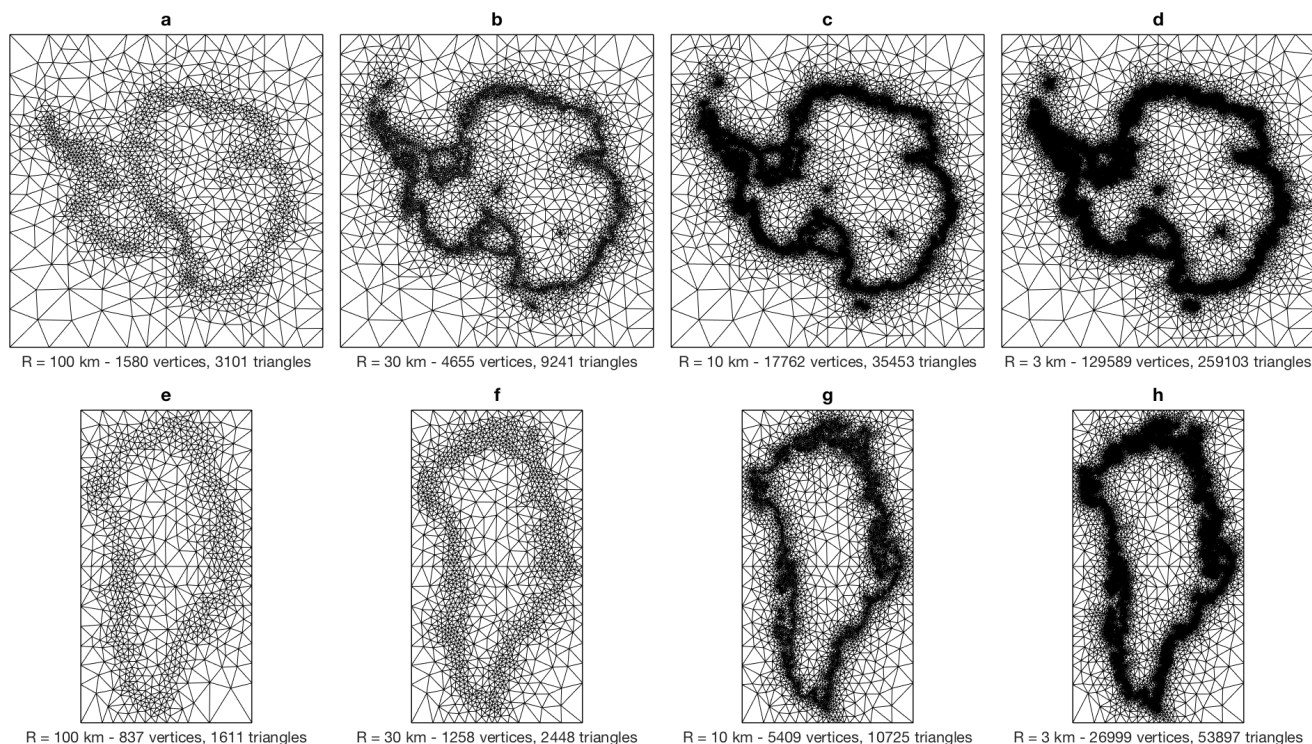
Figure 1: Meshes generated for Antarctica (a-d) and Greenland (e-h), based on present-day ice-sheet geometry, with maximum ice-margin (including grounding line and calving front) resolutions of 100, 30, 10 and 3 km, respectively. For Antarctica, two high-resolution locations are included: one at the South Pole, and one at the EPICA Dome C drill site. Both have been prescribed the same resolution as the ice margin.

Since the ice margin, the calving front and the grounding line are one-dimensional regions, increasing the desired resolution only over these regions results in a total number of vertices and triangles that scales almost linearly with this resolution (though not quite, as increasing the resolution resolves more geographical features, increasing the length of the lines). The way the number of vertices and the computational speed of the model scale with the prescribed resolution, is investigated in more detail in Sect. 4.

Mesh generation has been parallelised by subdividing the model domain into separate regions for each processor, which generates a mesh for only that section. These "submeshes" are then merged to produce one single mesh that covers the entire model domain. An example of four such submeshes that have been generated in parallel for Antarctica is shown in Fig. 2. Although these particular submeshes have equally-sized domains, in general the size of each domain is chosen such that they contain approximately equal numbers of vertices, to ensure proper load balancing. The results presented in Sect. 4 show that this approach results in a computation time that scales with the number of processors to order 0.47.
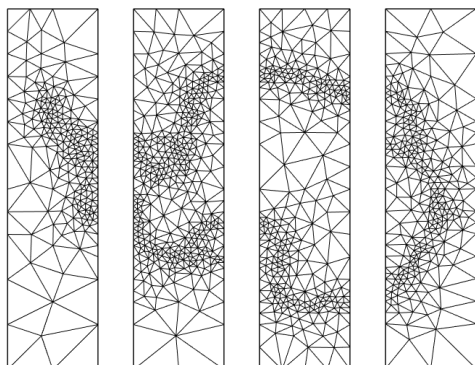
Figure 2: four Antarctic submeshes that will be merged into a single, domain-wide mesh

The mesh adapts to the ice-sheet geometry as it evolves through time. In every model time step, a "mesh fitness factor" is calculated, defined as the fraction of triangles that still meet all the fitness criteria in the extended version of Ruppert's

5 algorithm. For a newly generated mesh, this fraction is, by definition, 100 %. As the ice-sheet geometry evolves over time, the fitness factor slowly decreases. When it falls below a prescribed threshold value (typically 95 %), a mesh update is triggered, and an entire new mesh is generated. In glacial cycle simulations, the typical time between mesh updates is 50 – 500 model years. Although Ruppert's algorithm provides a very intuitive rule for refining a mesh, no such rules can be easily defined for "un-refining" a mesh. The simplest approach is therefore to generate an entirely new mesh once the fitness of the current mesh

10 falls below the threshold value. Once this is done, the model data fields (ice thickness, velocity, temperature, etc.) are projected from the old mesh to the new one, using trilinear interpolation for diagnostic fields such as ice velocity, surface albedo, etc., and second-order conservative remapping (Jones, 1999) for ice thickness and englacial temperature.

2.3 Ice dynamics

Ice velocities over land are calculated in UFEMISM using the SIA. The horizontal ice velocities in the vertical column, $u(z)$

15 and $v(z)$, can be expressed in terms of the depth-dependent ice diffusivity $D(z)$:

$$D(z) = 2(\rho g H)^n |\nabla h|^{n-1} H \int_0^z A(T^*)\zeta^n d\zeta, \tag{1}$$

$$u(z) = D(z)\frac{\partial h}{\partial x},$$

$$v(z) = D(z)\frac{\partial h}{\partial y}. \tag{2}$$

The depth-dependent ice diffusivity $D(z)$ is defined as a function of the ice density ρ, gravitational acceleration g, ice thickness H, surface gradient ∇h, Glen's flow law exponent n, and the temperature-dependent ice flow factor $A(T^*)$. For a comprehensive derivation of Eqs. 1 and 2, see e.g. Bueler and Brown (2009).

In order to solve these equations numerically, these different quantities need to be discretised on a grid. In many ice models, this is done using a combination of a standard central differencing scheme, and a "staggered" grid, where either the ice thickness H is averaged and the surface slopes $\frac{\partial h}{\partial x}$ and $\frac{\partial h}{\partial y}$ are differenced (Type I models; Huybrechts et al., 1996), or the other way around (Type II models). UFEMISM is, in this sense, a Type I model. The discretisation of spatial derivatives on an

5 unstructured triangular mesh is derived in Appendix A. Ice thickness changes over time are calculated using a finite volume approach: by calculating both the diffusivity D and surface slopes $\frac{\partial h}{\partial x}$ and $\frac{\partial h}{\partial y}$, and the resulting ice velocities u and v, on the boundaries between vertices (using the "staggered mesh" approach described in Appendix A), ice fluxes between individual vertices are calculated explicitly, and moved from one vertex to the other in every time step. This guarantees conservation of mass. UFEMISM also solves the SSA, to determine ice velocities for floating ice shelves, and sliding velocities for grounded

10 ice, following the approach of Bueler and Brown (2009). The iterative scheme for discretising and solving the set of partial different equations describing the SSA on the unstructured triangular mesh, is derived in Appendix B.

2.4 Thermodynamics

UFEMISM uses a mixed implicit-explicit finite differencing scheme with a fixed time step to solve the heat equation in a flowing medium:

$$\frac{\partial T}{\partial t} = \frac{k}{\rho c_p}\nabla^2 T - \boldsymbol{u} \cdot \boldsymbol{\nabla} T + \frac{\Phi}{\rho c_p},$$

(3)

$$\Phi = 2\left(\dot{\varepsilon}_{xz}\tau_{xz} + \dot{\varepsilon}_{yz}\tau_{yz}\right) = -\rho g(h - z)\left[\frac{\partial u}{\partial z}\frac{\partial h}{\partial x} + \frac{\partial v}{\partial z}\frac{\partial h}{\partial y}\right]$$

(4)

15 The three terms on the right-hand side of Eq. 3 respectively represent diffusion, advection and strain heating. In UFEMISM, horizontal diffusion of heat is neglected, simplifying Eq. 3 to:

$$\frac{\partial T}{\partial t} = \frac{k}{\rho c_p}\frac{\partial^2 T}{\partial z^2} - \boldsymbol{u} \cdot \boldsymbol{\nabla} T + \frac{\Phi}{\rho c_p},$$

(5)

This equation is discretised on an irregular grid in the vertical direction; all vertical derivatives are discretised implicitly, whereas horizontal derivatives are discretised explicitly. This mixed approach, which was first used in the ice-sheet model SICOPOLIS (Greve, 1997), is numerically stable (since both the steepest gradients and shortest grid distances are in the vertical

20 direction), relatively easy to implement, and fast to compute. Using 15 layers in the vertical direction, a time step of 10 years is typically sufficient to maintain numerical stability, independent of the horizontal resolution. The iterative scheme for solving this equation is derived in Appendix C.

3 Model verification and benchmark experiments

In order to verify the numerical solution to the ice dynamical equations, we performed several benchmark experiments, where we compare our model output to analytical solutions (Halfar, 1981; Bueler et al., 2005), or to results from the EISMINT intercomparison exercise (Huybrechts et al., 1996).

3.1 Verification using analytical solutions

For several schematic, simplified ice-sheet configurations, analytical solutions exist for the time evolution of the ice sheet. One of these was derived by Halfar (1981), describing a "similarity solution" for the time evolution of a radially symmetrical, isothermal ice sheet lying on top of a flat bed, with a uniform zero mass balance. For an ice sheet which, at time t_0, has a thickness at the dome H_0 and margin radius R_0, the time-dependent solution to the SIA, with Glen's flow law exponent n, reads:

$$H(r,t) = H_0 \left(\frac{t_0}{t}\right)^{\frac{2}{5n+3}} \left[1 - \left(\left(\frac{t_0}{t}\right)^{\frac{1}{5n+3}} \frac{r}{R}\right)^{\frac{n+1}{n}}\right]^{\frac{n}{2n+1}}, \tag{6}$$

$$t_0 = \frac{1}{(5n+3)\Gamma}\left(\frac{2n+1}{n+1}\right)^n \frac{R_0^{n+1}}{H_0^{2n+1}}, \Gamma = \frac{2A}{5}(\rho g)^n \tag{7}$$

Since the surface mass balance is zero, any change in ice thickness is caused only by ice dynamics, making this a useful experiment for verifying ice-sheet model numerics.

We performed several simulations with UFEMISM of an ice-sheet that starts at $t = t_0$ with the Halfar solution for $H_0 = 5000$ m, $R_0 = 300$ km, $A = 10^{-16}$ Pa^3 yr^{-1}, and $n = 3$. The ice-margin resolutions for the different simulations were set to 50, 16, 8, and 4 km. Shown in Fig. 3 are transects of the simulated ice sheet at different points in time, compared to the analytical solution.
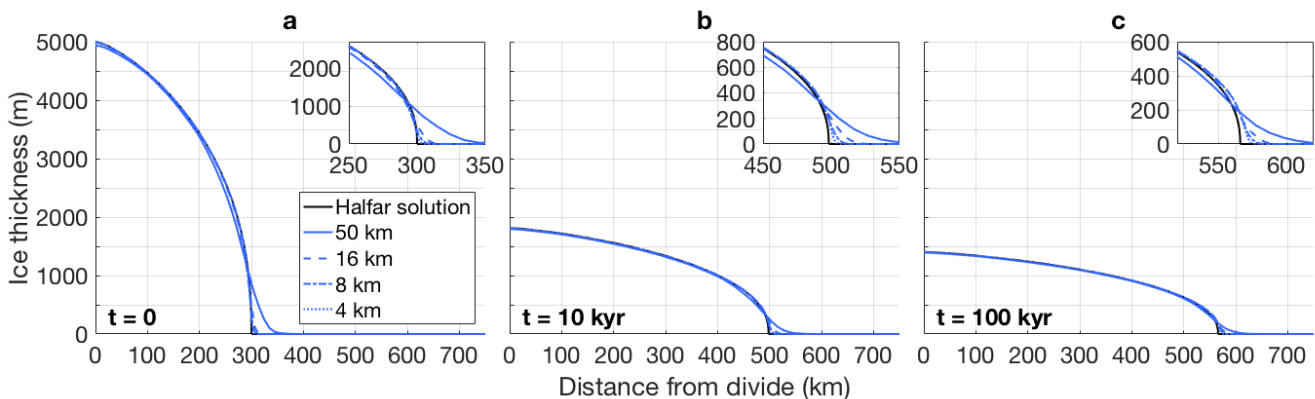
Figure 3: The evolution through time of a schematic, radially symmetric, isothermal ice sheet, as simulated by UFEMISM at different ice-margin resolutions, compared to the Halfar solution starting at t = t₀. The small sub-panels in the top-right corner of the panels show a zoomed-in view of the ice margin, showing how the simulated ice margin converges to the analytical solution as the model resolution increases.

5   UFEMISM reproduces the analytical solution well, with the largest errors occurring at the margin, and decreasing with resolution. As was shown by Bueler et al. (2005), this is the case for all spatially discrete SIA models, and is due to the fact that the SIA is intrinsically unable to accurately describe ice flow at the margin, where the surface slope becomes very large. They also show that these errors do not "corrupt" the model solution over the interior. This matches our results, where the modelled ice-sheet interior after 100,000 years is still close to the analytical solution. At that time, the modelled ice-sheet

10   margin at 4 km resolution differs from the analytical solution by about 10 km.

A generalisation of the solution by Halfar (1981), applicable to problems including a simple elevation-dependent accumulation rate, was derived by Bueler et al. (2005). For an accumulation given by:

$$M_\lambda(r,t) = \frac{\lambda}{t} H(r,t),$$

(8)

the solution for the ice thickness over time reads:

$$H_\lambda(r,t) = H_0 \left(\frac{t_0}{t}\right)^\alpha \left\{ 1 - \left[\left(\frac{t}{t_0}\right)^{-\beta} \frac{r}{R_0} \right]^{\frac{n+1}{n}} \right\}^{\frac{n}{2n+1}},$$

(9)

$$\alpha = \frac{2 - (n+1)\lambda}{5n+3}, \beta = \frac{1 + (2n+1)\lambda}{5n+3}, t_0 = \frac{\beta}{\Gamma} \left(\frac{2n+1}{n+1}\right)^n \frac{R_0^{n+1}}{H_0^{2n+1}}, \Gamma = \frac{2A}{5}(\rho g)^n.$$

(10)

15   The special case of zero mass balance, described by $\lambda = 0$, gives the solution by Halfar (1981). The results of this experiment are shown in Fig. 4, for the case of $\lambda = 5$, which describes a positive accumulation rate, resulting in a rapidly expanding ice sheet. Here, too, UFEMISM reproduces the analytical solution well, with the largest errors occurring at the ice-sheet margin, and decreasing with resolution.
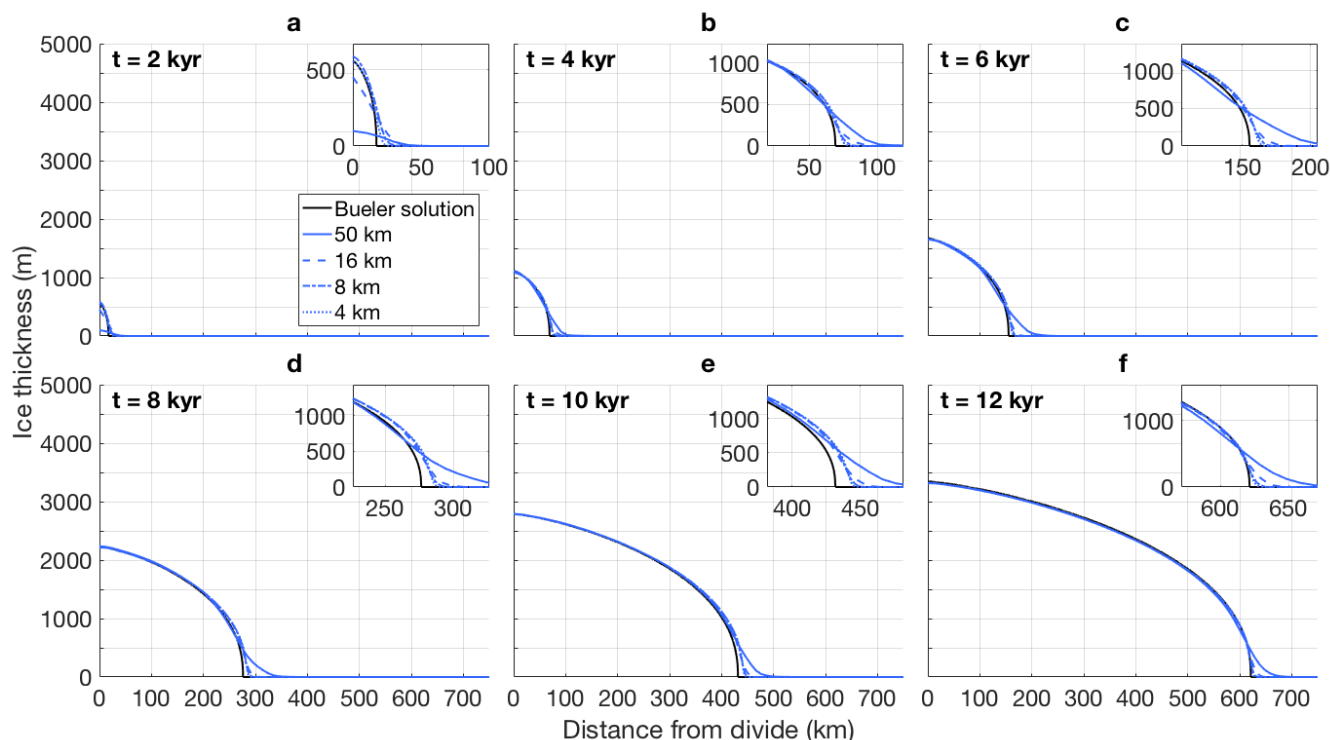
Geoscientific
Model Development
Discussions

Figure 4: the evolution through time of a schematic, radially symmetric, isothermal ice sheet, with positive accumulation rate, as simulated by UFEMISM at different ice-margin resolutions, compared to the Bueler solution.

3.2 EISMINT benchmark experiments

5   In order to further investigate the validity of our numerical schemes for ice dynamics and thermodynamics, we used UFEMISM to perform the first set of EISMINT benchmark experiments (Huybrechts et al., 1996). All of the 6 experiments consist of a radially symmetric ice sheet, lying atop a flat, non-deformable bedrock. While the temperature of the ice is calculated dynamically, the ice flow factor is kept fixed at a value of $A = 10^{-16}$ Pa^3 yr^{-1}, meaning that ice temperature is a purely diagnostic variable. The six experiments are divided into two groups of three: a "moving margin" and a "fixed margin" group; in the

10   "fixed margin" experiments, the mass balance is such that the expected theoretical ice margin lies outside the model domain, and ice thickness at the boundary is artificially kept at zero. In realistic model configurations, such a margin (i.e. where the ice thickness does not approach zero) can occur at a calving front. The three experiments within a group have different mass balances; a fixed, "steady-state" mass balance, one with an added 20 kyr sinusoid, and one with a 40 kyr sinusoid, which is useful for investigating the performance of the model in terms of temporal evolution. Simulations for each experiment were

15   performed with ice-margin resolutions of the original EISMINT resolution of 50 km, as well as finer resolutions of 16, 8 and 4 km.

Table 1: The six different EISMINT experiments.

Experiment	Margin	Mass Balance
I	moving	steady-state
II	moving	20 kyr
III	moving	40 kyr
IV	fixed	steady-state
V	fixed	20 kyr
VI	fixed	40 kyr

Shown in Fig. 5 are the simulated ice thickness and ice velocity of a radial transect of the ice sheet in experiment I, at the end of a 120 kyr simulation that was initialised with an ice thickness of zero. These results agree well with those presented by Huybrechts et al. (1996), showing an ice sheet that is ~2960 m thick at the divide, and has a maximum outward ice velocity of 5 ~55 m yr^{-1} at approximately 450 km away from the divide. The small sub-panel in the upper right corner of panel A zooms in on the ice margin, showing that the modelled ice margin converges to the analytical ice margin (the perimeter of the circle where the mass balance integrates to zero) as the resolution increases.
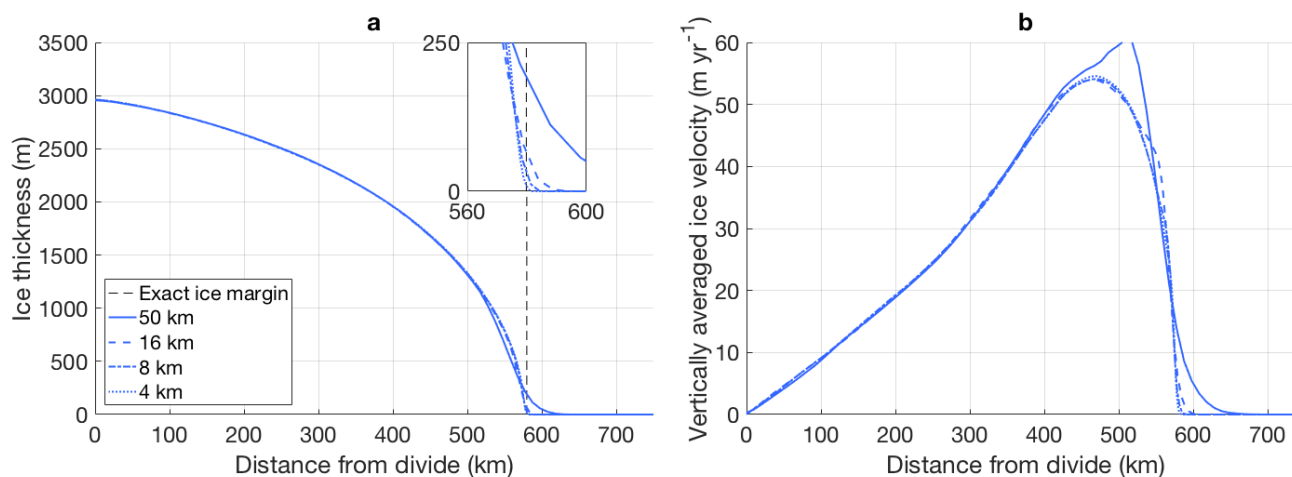
Figure 5: a) Ice thickness and b) vertically averaged ice velocity of a radial transect of the ice sheet in experiment I, simulated by 10 UFEMISM at 50, 16, 8, and 4 km ice-margin resolutions. The vertical dashed line in panel a denotes the analytical ice margin. The "bump" in the 50 km ice velocity transect around 500 km from the ice divide is a plotting artefact.

Shown in Fig. 6 are the same transects for experiment IV (steady-state, fixed margin), showing an ice sheet that is ~3400 m thick at the divide, compared to 3340 – 3420 m in Huybrechts et al. (1996).
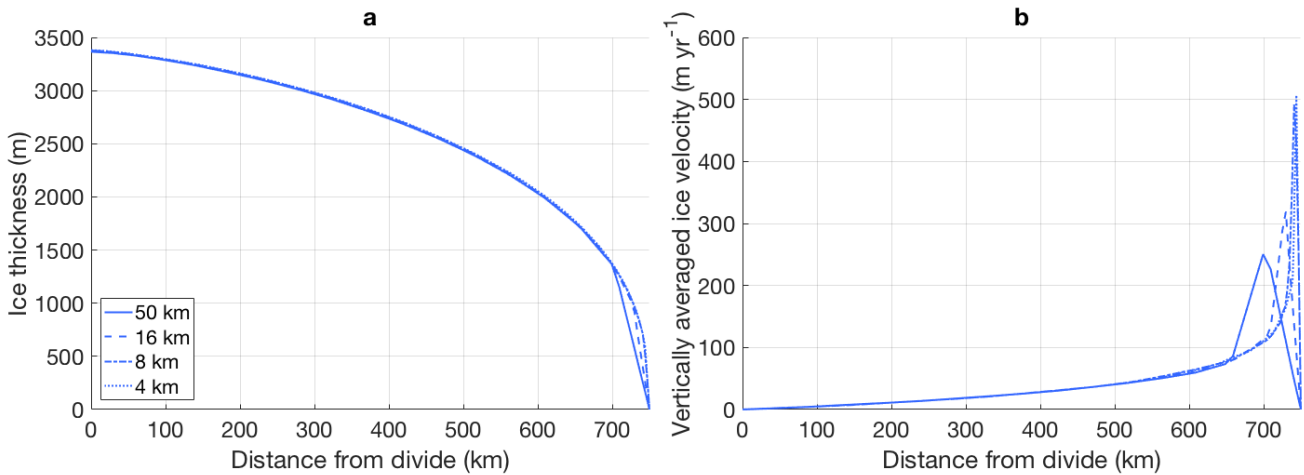
Geoscientific
Model Development
Discussions

Figure 6: a) Ice thickness and b) vertically averaged ice velocity of a radial transect of the ice sheet in experiment IV, simulated by UFEMISM at 50, 16, 8, and 4 km ice-margin resolutions. The sharp peaks in the velocity near the margin are a result of the singularity in these Nye-Vialov margins, the ice thickness approaches zero as one approaches the margin, but the ice flux remains finite, implying an infinite velocity.

Figs. 7 and 8 show the temperature transect of the ice sheet for experiments I and IV at 4 km resolution, and the basal temperature transects for all simulations in these experiments. Following the specifications from Huybrechts at al. (1996), the thermal conductivity and specific heat capacity of ice are kept fixed at uniform values of $k = 2.1$ J s^{-1} m^{-1} K^{-1} and $c_p = 2009$ J kg^{-1} K^{-1}, respectively. Here, too, results agree well those reported by Huybrechts et al. (1996). In experiment I, basal temperatures at the ice divide are $11 - 12$ K below the pressure melting point (PMP), increasing along the outward transect until they reach the PMP at $350 - 400$ km from the divide. In experiment IV we see similar results, with basal temperatures at the ice divide lying around 8 K below PMP, reaching PMP slightly further towards the margin at $400 - 450$ km from the divide. Preliminary experiments with a one-dimension set-up (vertical column only) show that these results are robust across different choices of vertical discretisations.

Geoscientific
Model Development
Discussions
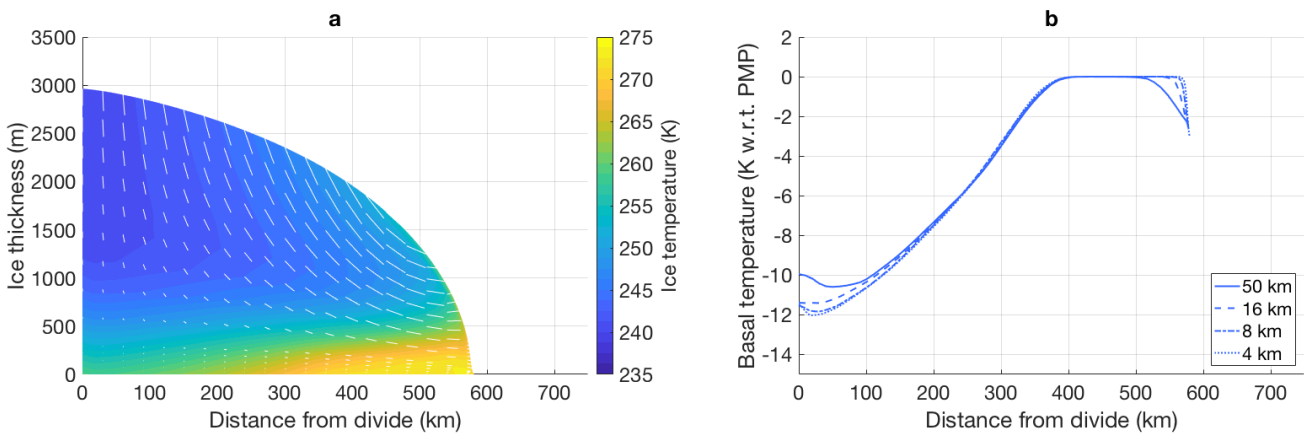
Figure 7: a) Ice temperature and velocity vectors for the steady-state ice sheet in experiment I, as produced by UFEMISM with a 4 km ice margin resolution. b) Basal temperature transects for the different simulations in the same experiment.
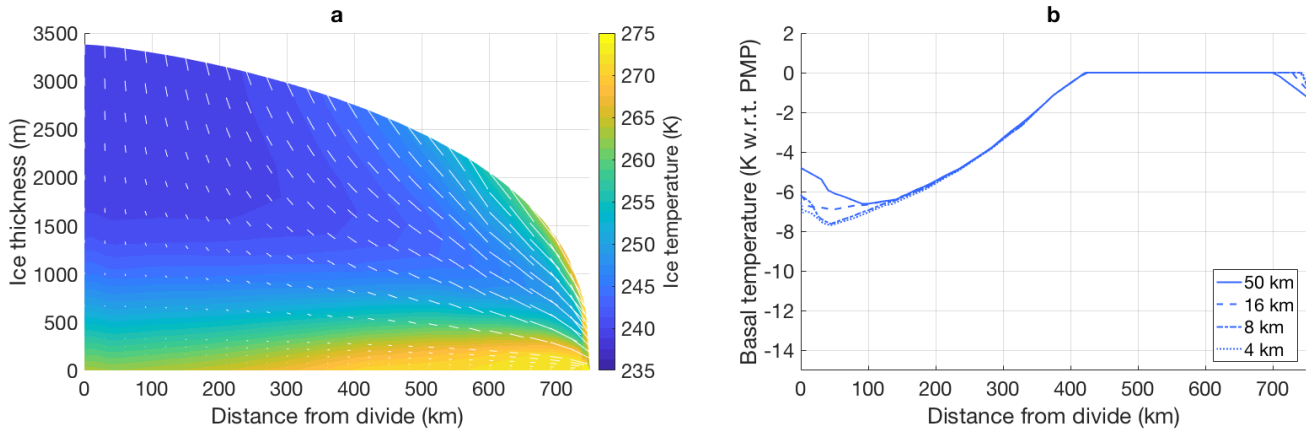
Figure 8: a) Ice temperature and velocity vectors for the steady-state ice sheet in experiment IV, as produced by UFEMISM with a 4 km ice margin resolution. b) Basal temperature transects for the different simulations in the same experiment.

Fig. 9 shows the ice thickness change at the ice divide over time for experiments II, III, V and VI. Here too, the results from simulations with UFEMISM at different resolutions agree with the results published by Huybrechts et al. (1996). The glacial-interglacial ice thickness (G-IG) changes for all four experiments lie within the ranges reported by Huybrechts et al. (1996), as listed in the top-right corners of both panels of Fig. 9. The simulated ice thickness timeseries at different resolutions are not distinguishable. Fig. 10 shows the basal temperature relative to the pressure melting point over time, for the same experiments. We find G-IG temperature changes that are slightly smaller than the values reported by Huybrechts et al. (1996), lying just outside their reported ranges. In agreement with the findings of Huybrechts et al. (1996), introducing glacial cycles in the moving margin experiments (Fig. 10a) results in G-IG mean temperature decrease of about 1 K, while the fixed margin experiments (Fig. 10b) see a temperature increase of about 2.5 K.
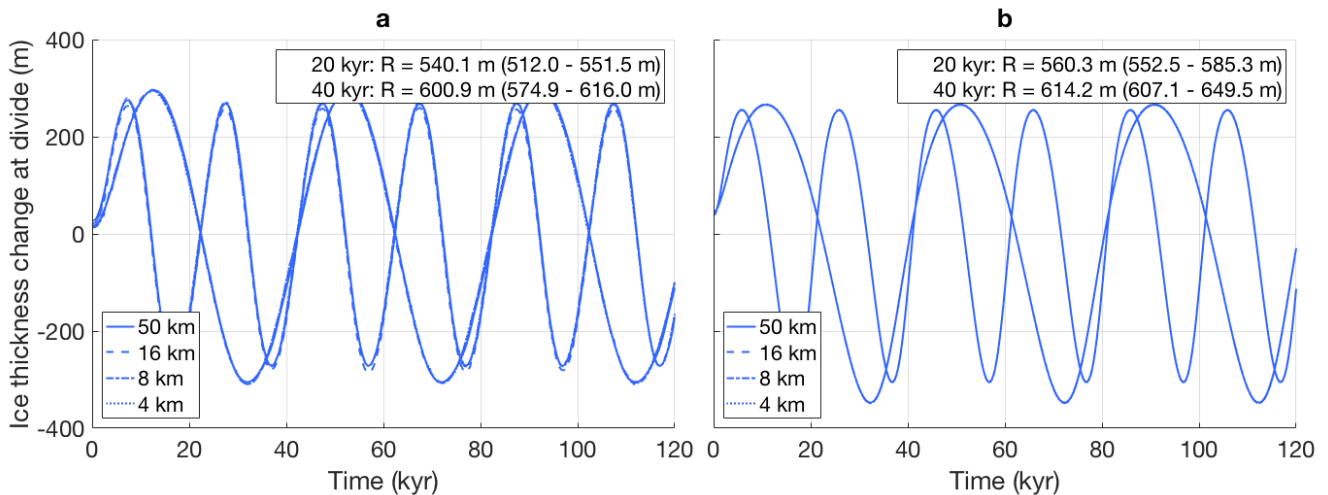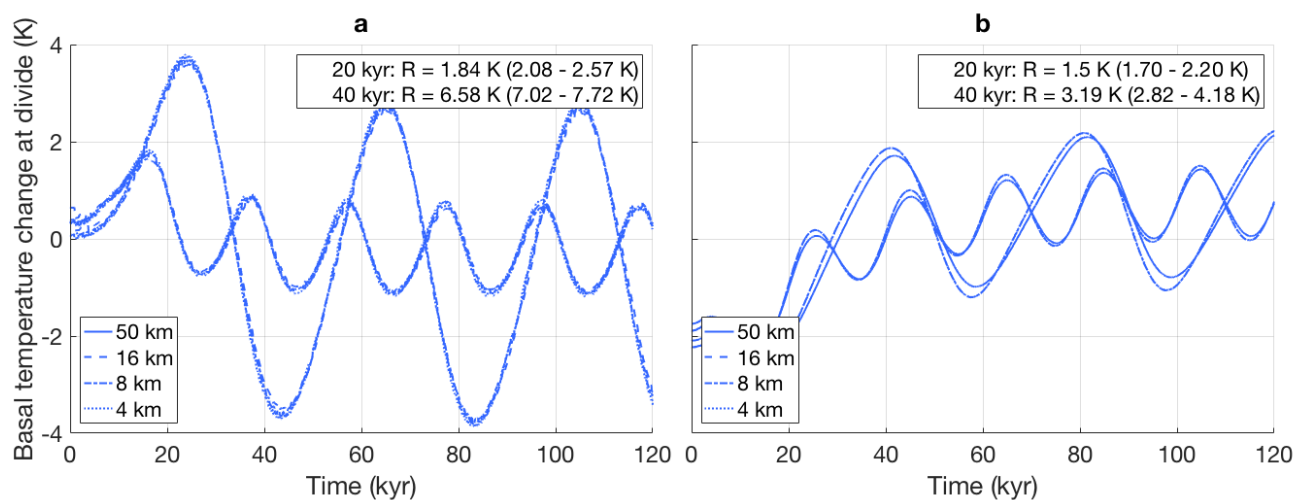
Figure 9: a) Ice thickness change at the divide for experiments II and III (moving margin, respectively 20 and 40 kyr sinusoid mass balance perturbation). b) The same for the fixed margin experiments (V and VI), simulated by UFEMISM at 50, 16, 8, and 4 km ice-margin resolutions. Listed in the top-right corners of both panels are the glacial-interglacial ice thickness changes simulated by UFEMISM, compared to the ranges reported by Huybrechts et al. (1996) between brackets.

Figure 10: a) Basal temperature change at the divide for experiments II and III (fixed margin, respectively 20 and 40 kyr sinusoid mass balance perturbation). b) The same for the fixed margin experiments (V and VI), simulated by UFEMISM at 50, 16, 8, and 4 km ice-margin resolutions. Listed in the top-right corners of both panels are the glacial-interglacial basal temperature changes simulated by UFEMISM, compared to the ranges reported by Huybrechts et al. (1996) between brackets.

4 Computational performance

The first version of UFEMISM presented here is parallelised using the MPI construct of "shared memory", allowing the program to run in parallel on a number of processor cores that are able to directly access the same physical memory (usually either 16, 24, or 32 cores on typical computer clusters or supercomputers). This is a temporary choice; the effort to extend the parallelisation to multiple nodes, using the full capability of MPI, is still ongoing, but is beyond the scope of this paper.

In order to investigate the computational performance of the different model components, we performed a series of simulations of Antarctic ice-sheet retreat. For these simulations, the climate was set to the present-day observed climate (ERA-40; Uppala et al., 2005) plus a spatially and temporally uniform 5 °C warming term to force ice-sheet retreat (and thus ensure frequent mesh updating) over a period of 10,000 years. The results of one of these simulations are shown in Fig. 11.
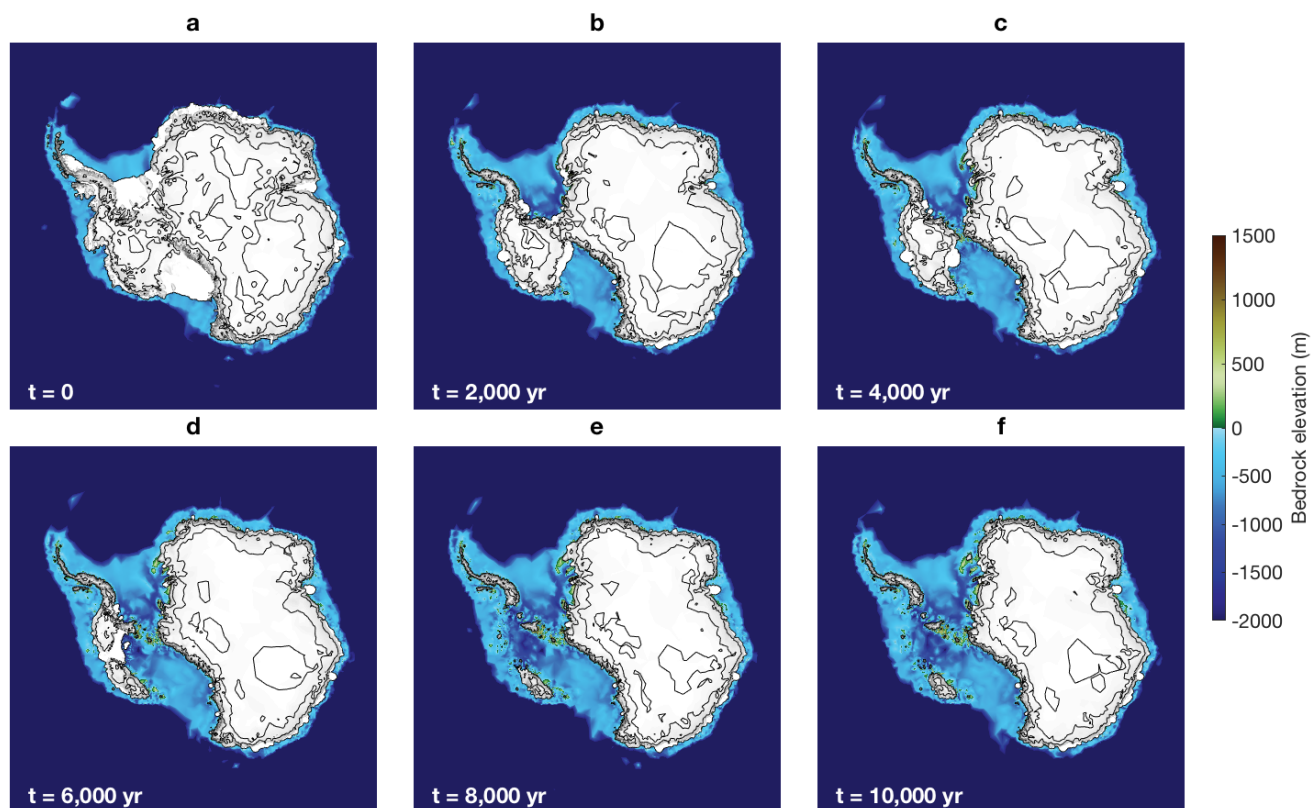
Figure 11: Results from the 10,000 yr Antarctic retreat simulation with a 4 km grounding line resolution. Panels a-f show the modelled ice sheet at 2,000 yr intervals. Bedrock elevation is indicated by colours, surface elevation contours are shown at 1,000 m intervals.

5   The computation times of the different model components as a function of number of cores and model resolution are shown in Fig. 12.
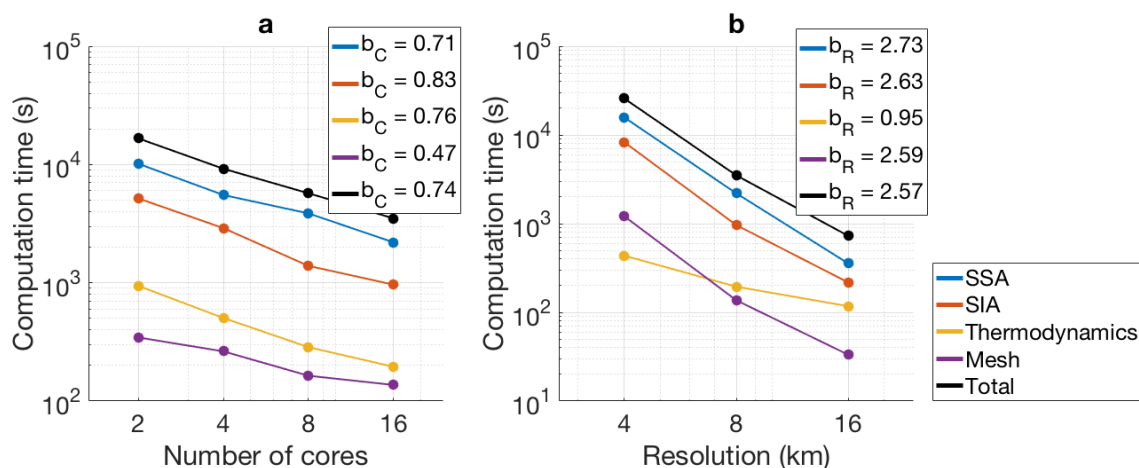
Figure 12: a) Computation times vs number of cores for the SSA (blue), SIA (red), thermodynamics (yellow) and mesh updating (purple) model components, as well as for the entire model (black), for a 10,000 yr Antarctic retreat simulation at 8 km resolution. These four components together account for ~99.8 % of the total computation time. b) Computation times vs model resolution for the same model components, run on 16 cores.

5 Fig. 12, panel A shows the degree of parallelisation of the model components. For each model component, logarithmic fits have been made between the computation time t, the number of cores C, and the horizontal resolution R. For the number of cores C, this fit is described by $\ln t = a + b_C \ln C$, with b_C describing the degree of parallelisation, such that $b_C = 1$ describes perfect parallelisation, i.e. doubling the number of cores reduces the computation time by half. For the resolution, this fit is described by $\ln t = a + b_R \ln R$. The three physics modules have a good degree of parallelisation, ranging from $b_C = 0.71$ for

10 the SSA to $b_C = 0.83$ for the SIA. Mesh updating scales less well, with $b_C = 0.47$. However, since mesh updating accounts for only $2 - 4$ % of total computation time, this does not noticeably affect the parallelisation of the complete model, which has $b_C = 0.74$. The SSA dominates the total computation time across all resolutions and numbers of processors, requiring as much or more computation time as all other model components combined.

15 Fig. 12, panel B shows computation versus model resolution for the same model components. In an idealised situation (such as the EISMINT experiments), the SIA and SSA should scale with the resolution to order $b_R = 3$ in a square-grid model (two orders from the number of grid cells, and one from the time-step dependence on resolution, according to the numerical stability criterion for the solution of the mass conservation equation). In UFEMISM, this can be theoretically reduced to order $b_R = 2$, since a high resolution is only applied over the one-dimensional ice-sheet margin, where the diffusivity is not necessarily the

20 largest of the model domain. The reason that the results shown in Fig. 12 deviate from this idealised case is because, for a realistic ice sheet, increasing the resolution resolves more topographical features, which increases the length of the one-dimensional domains of the ice margin, the grounding line, and the calving front. This is illustrated in Fig. 13, which shows the number of vertices of meshes for Greenland and Antarctica versus the model resolution at the ice margin.
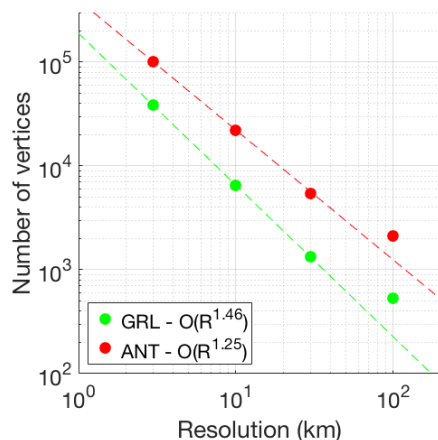
25 **Figure 13: Number of vertices vs ice margin resolution for Greenland and Antarctica.**

16

Similarly, resolving more topographical features also leads to an increased ice diffusivity in areas with steep gradients, decreasing the critical time step and increasing the computation time. Preliminary experiments with a simply square-grid model showed the same effect, and the computation time for those experiments scaled with resolution to order $b_R = 3.5 - 4.0$, rather than the $b_R = 3$ in the idealised case.

5

The computation time of the mesh updating component also scales well with model resolution, to the order $b_R = 2.59$. The thermodynamics component scales even better, with order $b_R = 0.95$. The reason that this is so much lower than the SIA and SSA components, is that the thermodynamics uses a constant rather than a dynamic time step, which is related to the choice of vertical discretisation rather than to the horizontal model resolution.

10

In these experiments, the entire ice margin, including the grounding line and calving front, was given the same resolution. When using UFEMISM for actual paleo-simulations, such a high resolution would only be used for the grounding line, where it directly increases the physical accuracy of the solution. In one last experiment, we performed the same Antarctic retreat simulation, with the resolution set to 4 km for the grounding line, 16 km for the rest of the ice margin, and 200 km over the

15 ice sheet interior. Run on 16 cores, the simulation took 5 h 30 m to complete, which predicts a requirement of about 70 h for an entire glacial cycle. Extrapolating these numbers to the Greenland, North American and Eurasian ice sheets is difficult, due to the differences in size, glacial-interglacial geometry changes, and relative area of floating ice. Previous work with the square-grid model ANICE (Berends et al., 2018, 2019) indicates that the Antarctic and North American ice sheets typically have roughly the same computational expense when they are at their maximum extent, while the Eurasian and Greenland ice sheets

20 respectively require about ½ and ¼ as much computation time. Based on these numbers, a full glacial cycle simulation of all four ice sheets with UFEMISM would take somewhere between 1.5 and 3 times as much as one for only Antarctica, implying a required computation time of about $1,600 - 3,200$ core hours ($100 - 200$ wall hours on a 16-core system). For comparison, a full glacial cycle simulation with ANICE at 40 km resolution takes about 8 core hours, which implies a simulation at 4 km would take about $25,000 - 80,000$ core hours (numbers based on extrapolation), meaning that UFEMISM is about $10 - 30$

25 times faster. If the grounding line resolution is decreased to 8 km, these numbers decrease to about $320 - 640$ core hours ($20 - 40$ wall hours) for UFEMISM, and $2,200 - 5000$ core hours for ANICE.

UFEMISM can be compiled with both the gfortran and ifort compilers, requiring only LAPACK, NetCDF and MPI as external packages, and can run on any number of processors (with a minimum of 2) that can access the same shared memory chip. This

30 means that it should be possible to compile and run the model on most consumer-grade systems. The model contains roughly 200 double precision data fields. For the 3 km resolution Antarctica mesh (~100,000 vertices) shown in Fig. 1, this implies a memory usage of about 1.6 Gb. Output is written to NetCDF files at about 10 kb per vertex, implying that a 120 kyr glacial cycle simulation of Antarctica at 3 km (100,000 vertices), where output is written every 1,000 model years, would generate about 150 Gb of data.

Conclusions and discussion

We have presented and evaluated a new thermomechanically coupled ice-sheet model, UFEMISM, which solves the SIA and SSA versions of the ice-dynamical equations on a fully adaptive mesh. The model is able to accurately reproduce the analytical solutions for the evolution of schematic ice sheets by Halfar (1981) and Bueler et al. (2005), as well as the EISMINT

5 benchmark experiments (Huybrechts et al., 1996), indicating that the numerical scheme developed for UFEMISM is valid. Analysis of the computational performance of the model indicates that it scales well with both number of processors and model resolution. Based on those results, UFEMISM should be able to simulate the evolution of the four large continental ice sheets over an entire glacial cycle, with a grounding line resolution of 4 km, in 1600 – 3200 core hours (100 – 200 wall hours on 16 processors), which is very feasible on typical facilities for scientific computation.

10

In the benchmark experiments we performed, only the SIA is solved, as floating ice and basal sliding are not considered. While a number of benchmark experiments exist for floating ice, these are less straightforward. MISMIP (Pattyn et al., 2012) describes a one-dimensional (flowline) case, which is shown in the subsequent MISMIP3d (Pattyn et al., 2013) experiment to be difficult to apply for a two-dimensional case. The two-dimensional experiments in MISMIP3d do not always have analytical

15 solutions, and the results produced by different models are shown to be strongly resolution-dependent. Different semi-analytical solutions for the ice flux across the grounding line in the absence of buttressing have been proposed (Schoof et al., 2007; Tsai et al., 2015). However, implementing those in numerical models has proven difficult (Pollard and DeConto, 2012; Pattyn et al., 2013), and recent work has demonstrated that these analytical solutions cannot be feasibly altered to account for buttressing, which is the case for the majority of Antarctic grounding lines (Reese et al., 2018). This means that the problem

20 of solving the stress balance near the grounding line in numerical models without resorting to very high (< 100 m) resolutions remains open, and is not solved by this new model configuration.

The current version of UFEMISM has been parallelised using MPI shared memory, meaning the number of processors that can run the model depends on the hardware system, limited by how many processors can access the same memory chip. Since

25 the three most computationally expensive model components (the SSA, SIA and thermodynamics modules, respectively) are shown to scale well, extending the parallelisation framework to allow the model to run on multiple nodes might substantially reduce the wall clock time for large simulations. The current version of UFEMISM uses domain subdivision to parallelise the mesh generation algorithm. This approach lends itself well to multi-node parallelisation, as each node can be assigned a region of the model domain.

30

The iterative scheme used in solving the SSA is currently the most computationally demanding model component, requires as much or more computation time as all the other model components combined. Preliminary experiments have identified several possible solutions that could substantially reduce this, including a spatially variable relaxation parameter (see Appendix B)

and/or a multigrid scheme. Efforts to develop these solutions into applications that are robust and stable enough for the large-scale, long-term simulations for which UFEMISM is intended are ongoing, but are beyond the scope of this study.

As UFEMISM is intended to be used for paleo-simulations, it has been developed to be able to include an elaborate climate /
5   mass balance forcing. Previous work by our group has focused on developing a computationally efficient climate forcing that explicitly includes important feedback processes in the ice-climate system, such as the altitude-temperature feedback, ice-albedo feedback and orographic precipitation feedback (Berends et al., 2018). While these solutions have not yet been implemented in UFEMISM, it has been designed with these solutions in mind, so that implementing them should be straightforward. We have also previously worked on developing a coupled ice-sheet – sea-level model (de Boer et al., 2014).
10 A key improvement in UFEMISM with respect to our previous ice-sheet model is the improved treatment of grounding line dynamics, which makes it important to accurately account for changes in the geoid and the resulting changes in water depth at the grounding line. While UFEMISM has not yet been coupled to a sea-level model, it has been designed with such a future coupling in mind.

Code and data availability

15 The Fortran90 source code of UFEMISM, as well as a collection of Matlab scripts for analysing and visualising output data, and some more elaborate documentation, are freely available at http://doi.org/10.5281/zenodo.4001592. The different benchmark experiments described in Sect. 3 can be run with the supplied config files. The Antarctic retreat simulations described in Sect. 4 require input files describing the present-day Antarctic ice sheet and climate. For the ice-sheet and bedrock geometry, we used the Bedmachine Antarctica v1 data by Morlighem et al. (2019) which is freely available as supplementary
20 material to their publication. For the present-day climate we used the ERA-40 reanalysis (Uppala et al., 2005), which can be downloaded from the ECMWF website (https://www.ecmwf.int/) after creating a free user account.

Geoscientific
Model Development
Discussions

5

Appendix A – Discretising derivatives on an unstructured triangular mesh

In order to derive a scheme for the unstructured triangular mesh, we first review the much more commonly used scheme for a regular rectangular grid, visualised in Fig. A1.
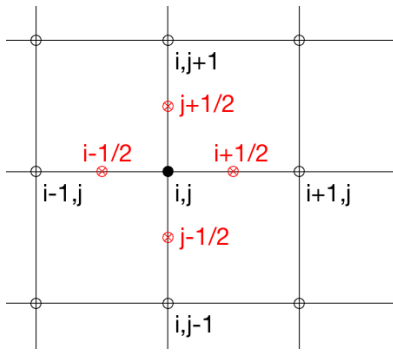
10

Figure A1: part of a regular rectangular grid, showing the "staggered" Arakawa C grid points in red.

The slopes (first order spatial derivatives) of a function f on grid point i,j are typically discretised using central differencing:

$$f_x^{i,j} = \frac{f^{i+1,j} - f^{i-1,j}}{2\Delta x},$$ (A1a)

$$f_y^{i,j} = \frac{f^{i,j+1} - f^{i,j-1}}{2\Delta y}.$$ (A1b)

These can be viewed as the averages of the slopes on the "staggered" Arakawa C grid points $i+\frac{1}{2}, i-\frac{1}{2}, j+\frac{1}{2}$ and $j-\frac{1}{2}$:

$$f_x^{i+\frac{1}{2},j} = \frac{f^{i+1,j} - f^{i,j}}{\Delta x}, f_x^{i-\frac{1}{2},j} = \frac{f^{i,j} - f^{i-1,j}}{\Delta x}, f_x^{i,j} = \frac{f_x^{i+\frac{1}{2},j} + f_x^{i-\frac{1}{2},j}}{2},$$ (A2a)

$$f_y^{i,j+\frac{1}{2}} = \frac{f^{i,j+1} - f^{i,j}}{\Delta y}, f_y^{i,j-\frac{1}{2}} = \frac{f^{i,j} - f^{i,j-1}}{\Delta y}, f_y^{i,j} = \frac{f_y^{i,j+\frac{1}{2}} + f_y^{i,j-\frac{1}{2}}}{2}.$$ (A2b)

The curvatures (second-order spatial derivatives) of f are then discretised as:

$$f_{xx}^{i,j} = \frac{f_x^{i+\frac{1}{2},j} - f_x^{i-\frac{1}{2},j}}{\Delta x} = \frac{f^{i+1,j} - 2f^{i,j} + f^{i-1,j}}{\Delta x^2},$$ (A3a)

20

$$f_{yy}^{i,j} = \frac{f_y^{i,j+\frac{1}{2}} - f_y^{i,j-\frac{1}{2}}}{\Delta y} = \frac{f^{i,j+1} - 2f^{i,j} + f^{i,j-1}}{\Delta y^2}, \tag{A3b}$$

$$f_{xy}^{i,j} = \frac{f_x^{i,j+1} - f_x^{i,j-1}}{2\Delta y} = \frac{f^{i+1,j+1} + f^{i-1,j-1} - f^{i+1,j-1} - f^{i-1,j+1}}{4\Delta x \Delta y}. \tag{A3c}$$

These can be viewed as the derivatives of the first-order derivatives on the staggered Arakawa C grid points. The important property of these discretisation, which we will use to generalise these definitions for use on the unstructured triangular mesh, is that the values of the first- and second-order spatial derivatives of f on a grid cell i,j are linear combinations of the values of f on i,j and its eight-fold neighbours.
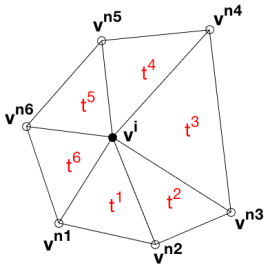
5

Figure A2: a very simple unstructured triangular mesh. The vertex under consideration, v^i, is surrounded by its neighbours v^{n1} to v^{n6}, which together span triangles t^1 to t^6.

Consider the unstructured triangular mesh in Fig. A2. We first define the partial derivatives f_x and f_y on the triangles surrounding vertex v^i. Since these triangles are plane sections, they have well-defined first-order derivatives, which can be

10 expressed using the normal vector to the plane. Defining $\boldsymbol{v}^i = [x^i, y^i, f^i]$, the upward normal vector $\boldsymbol{n}^1$ to triangle $t1$, spanned by $\boldsymbol{v}^i$, $\boldsymbol{v}^{n1}$ and $\boldsymbol{v}^{n2}$, is given by:

$$\boldsymbol{n}^1 = (\boldsymbol{v}^{n1} - \boldsymbol{v}^i) \times (\boldsymbol{v}^{n2} - \boldsymbol{v}^i). \tag{A4}$$

The first-order spatial derivatives of f on t^1, f_x^{t1} and f_y^{t1}, are then given by:

$$f_x^{t1} = \frac{-n_x^1}{n_z^1} = \frac{f^i(y^{n1} - y^{n2}) + f^{n1}(y^{n2} - y^i) + f^{n2}(y^i - y^{n1})}{x^i(y^{n1} - y^{n2}) + x^{n1}(y^{n2} - y^i) + x^{n2}(y^i - y^{n1})}, \tag{A5a}$$

$$f_y^{t1} = \frac{-n_y^1}{n_z^1} = \frac{f^i(x^{n2} - x^{n1}) + f^{n1}(x^i - x^{n2}) + f^{n2}(x^{n1} - x^i)}{x^i(y^{n1} - y^{n2}) + x^{n1}(y^{n2} - y^i) + x^{n2}(y^i - y^{n1})}. \tag{A5b}$$

As we see, these are linear combinations of the function values on vertex v^i and its direct neighbours v^{n1} and v^{n2}, where the linear coefficients are functions of the x and y coordinates of the three vertices. Similar to the approach used on the regular

15 rectangular grid, we calculate the spatial derivatives of f on i, f_x^i and f_y^i as the average of the values on the neighbouring piecewise smooth areas – in this case, the triangles:

$$f_x^i = \frac{f_x^{t1} + f_x^{t2} + \cdots}{N_{tri}^i}. \tag{A6}$$

21

where N_{tri}^i is the number of triangles containing vertex v^j. Preliminary experiments have shown that "weighting" the contributions from the surrounding triangles based on triangle size or internal angle, as is sometimes done, does not significantly affect the solution. Since $f_x^{t1}, f_x^{t2}, \dots$ are linear combinations of $f^i, f^{n1}, f^{n2},\dots$, we can write:

$$f_x^i = f^i N_x^i + \sum_{c=1}^{n} f^c N_x^{i,c}.$$

(A7)

where c sums over the vertices neighbouring v^i. The "neighbour functions" N_x^i and $N_x^{i,c}$ follow from the mesh geometry. This means that they can be calculated in advance, and used in the calculation of spatial derivatives of any function whose values are defined on the mesh vertices, greatly reducing the required computation time.

On the regular rectangular grid, the discretisation of the second derivative could be interpreted as the gradient of the first derivative on the staggered Arakawa C grid. This meant that the value of the second derivative on a grid cell could, like the first derivative, be expressed as a linear combination of the function values on that grid cell and its eight-fold neighbours. For the triangular mesh, we take a similar approach. We view the geometric centres of the triangles as "staggered" vertices, where the first spatial derivatives are calculated using the normal vector to the triangle. We construct a new set of "sub-triangles", spanned by the vertex v^i and these staggered vertices, as illustrated in Fig. A3. Since we know the values of f_x^i and f_y^i on these new vertices, we can use the same approach as before to calculate the second derivatives on the sub-triangles, and average them to get the value on v^i. Preliminary experiments showed that using the geometric centre instead of the circumcentre yields more stable solutions when using these to solve differential equations. A mathematical proof of why this is the case might be interesting, but lies beyond the scope of this study.
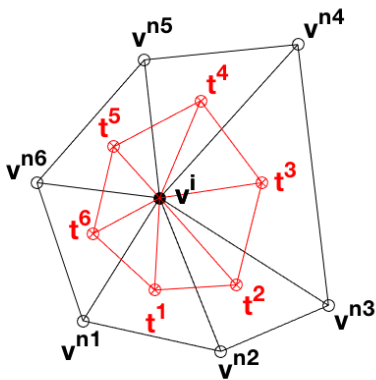
Figure A3: the geometric centres of the triangles surrounding vertex v^i make up the vertices of a new set of "sub-triangles", which can be used to define the second derivative of a function on vertex v^i.

For now, consider only the first derivative of f to x. The new, staggered vertex in triangle t^l is created by the horizontal coordinates of the geometric centre of t^l, and the derivative of f on t^l, f_x^{t1}, as the vertical coordinate:

$$t_x^1 = \left[\frac{x^i + x^{n1} + x^{n2}}{3}, \frac{y^i + y^{n1} + y^{n2}}{3}, f_x^{t1} \right]. \tag{A8}$$

Let t_s^1 be the sub-triangle spanned by $\boldsymbol{v}_x^i$, $\boldsymbol{t}_x^1$ and $\boldsymbol{t}_x^2$, where $\boldsymbol{v}_x^i = [x^i, y^i, f_x^i]$ (where f_x^i is given by Eq. A7). The normal vector $\boldsymbol{n}_x^1$ to this sub-triangle is given by:

$$\boldsymbol{n}_x^1 = (\boldsymbol{t}_x^1 - \boldsymbol{v}_x^i) \times (\boldsymbol{t}_x^2 - \boldsymbol{v}_x^i). \tag{A9}$$

The second-order derivatives of f on t_s^1 are then given by:

$$f_{xx}^{ts1} = \frac{-n_{x,x}^1}{n_{x,z}^1}, \tag{A10a}$$

$$f_{xy}^{ts1} = \frac{-n_{x,y}^1}{n_{x,z}^1}, \tag{A10b}$$

and the second-order derivatives f_{xx}^i and f_{xy}^i of f on v^i can be expressed as:

5

$$f_{xx}^i = \frac{f_{xx}^{ts1} + f_{xx}^{ts2} + \cdots}{N_{tri}^i} = f^i N_{xx}^i + \sum_{c=1}^n f^c N_{xx}^{i,c}, \tag{A11a}$$

$$f_{xy}^i = \frac{f_{xy}^{ts1} + f_{xy}^{ts2} + \cdots}{N_{tri}^i} = f^i N_{xy}^i + \sum_{c=1}^n f^c N_{xy}^{i,c}. \tag{A11b}$$

The second-order derivatives, like the first-order ones, can be expressed on the vertex v^i as a linear combination of the function value f^i on the vertex v^i, and the function values f^c on the neighbours v^c of v^i. The second-order neighbour functions N_{xx}^i and $N_{xx}^{i,c}$, like their first-order counterparts, follow from mesh geometry.

10

In order to describe fluxes between adjacent vertices, we need to define a discretisation of the derivatives on a point lying in between those two vertices. While it is possible to just take the average value of the derivatives on the vertices themselves, as defined by Eq. A7, the resulting value would be a linear combination of the function values on all the neighbours of these two vertices, resulting in an undesirable degree of numerical diffusion.
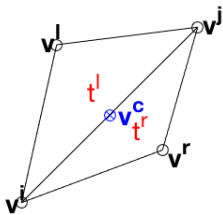
15

Figure A4: a very simple mesh, showing the staggered vertex v^c connecting vertices v^i and v^j.

Consider the simple mesh in Fig. A4, consisting of four vertices spanning two triangles. We defined the staggered vertex v^c as the midpoint between vertices v^i and v^j. On the two adjacent triangles, t^l and t^r, the derivatives f_x^l, f_y^l, f_x^r and f_y^r are well-defined:

$$f_x^l = \frac{-n_x^l}{n_z^l} = \frac{f^i(y^j - y^l) + f^j(y^l - y^i) + f^l(y^i - y^j)}{x^i(y^j - y^l) + x^j(y^l - y^i) + x^l(y^i - y^j)}, \tag{A12a}$$

$$f_y^l = \frac{-n_x^l}{n_z^l} = \frac{f^i(x^l - x^j) + f^j(x^i - x^l) + f^l(x^j - x^i)}{x^i(y^j - y^l) + x^j(y^l - y^i) + x^l(y^i - y^j)}, \tag{A12b}$$

$$f_x^r = \frac{-n_x^r}{n_z^l} = \frac{f^i(y^r - y^j) + f^j(y^i - y^r) + f^r(y^j - y^i)}{x^i(y^r - y^j) + x^j(y^i - y^r) + x^r(y^j - y^i)}, \tag{A12c}$$

$$f_x^r = \frac{-n_y^r}{n_z^l} = \frac{f^i(x^j - x^r) + f^j(x^r - x^i) + f^r(x^i - x^j)}{x^i(y^r - y^j) + x^j(y^i - y^r) + x^r(y^j - y^i)}. \tag{A12d}$$

Similar to the approach on the regular rectangular Arakawa C grid, we define the derivatives f_x^c and f_y^c of f on v^c as the average of the values on the two adjacent triangles:

$$f_x^c = \frac{1}{2}(f_x^l + f_x^r), \tag{A13a}$$

$$f_y^c = \frac{1}{2}(f_y^l + f_y^r). \tag{A13b}$$

The purpose of the Arakawa C grid is to separate velocities on cell boundaries into components parallel and orthogonal to those boundaries. The component orthogonal to the boundary carries a flux from one cell into another, while the parallel component does not. On the mesh, this is complicated by the fact that the orientation of boundaries is not fixed. Let **u** be the vector pointing from v^i to v^j:

$$\boldsymbol{u} = \begin{bmatrix} x_j - x_i \\ y_j - y_i \end{bmatrix}. \tag{A14}$$

The parallel and orthogonal derivatives f_{par}^c and f_{ort}^c of f on v^c are then given by:

$$f_{par}^c = f_x^c \frac{u_x}{|u|} + f_y^c \frac{u_y}{|u|}, \tag{A15a}$$

$$f_{ort}^c = f_x^c \frac{-u_y}{|u|} + f_y^c \frac{u_x}{|u|}. \tag{A15b}$$

It can be shown that this results in:

$$f_{par}^c = \frac{f^j - f^i}{|u|}. \tag{A16}$$

This is, of course, simply the slope of the line between v^i and v^j.

Appendix B – deriving a SOR iteration for solving the SSA

A concrete form of the stress balance resulting from the SSA is given by Bueler and Brown (2009), based on the work by
MacAyeal (1989) and Weis et al. (1999):

$$\frac{\partial}{\partial x}\left[2\bar{v}H\left(2u_x + v_y\right)\right] + \frac{\partial}{\partial y}\left[\bar{v}H\left(u_y + v_x\right)\right] - \frac{\tau_c u}{|\boldsymbol{u}|} = \rho g H h_x, \tag{B1a}$$

$$\frac{\partial}{\partial x}\left[\bar{v}H\left(u_y + v_x\right)\right] + \frac{\partial}{\partial y}\left[2\bar{v}H\left(u_x + 2v_y\right)\right] - \frac{\tau_c v}{|\boldsymbol{u}|} = \rho g H h_y. \tag{B1b}$$

The first two terms on the left-hand sides of these equations describe the membrane stresses, in terms of the horizontal ice

5   velocities u and v, the vertically averaged ice viscosity $\bar{v}$ and the ice thickness H. The last terms on the left-hand side describe
the basal stress resulting from a Coulomb sliding law, in terms of the till yield stress τ_c (which is spatially variable, and is zero
for floating ice) and the horizontal ice velocities u and v. These stresses are balanced out by the driving stress, which is
described on the right-hand sides in terms of the ice thickness H and the surface slopes h_x and h_y.

10   The vertically averaged ice viscosity is described by MacAyeal (1989) as a function of ice velocity:

$$\bar{v} = \frac{1}{2}\int_b^h A(T^*)^{\frac{-1}{n}}\,dz\left[u_x^2 + v_y^2 + u_x v_y + \frac{1}{4}\left(u_y + v_x\right)^2\right]^{\frac{1-n}{2n}}. \tag{B2}$$

Using the fact that $\frac{\partial}{\partial x}\bar{v}H \ll \bar{v}H$, and setting the basal shear stress $\tau_b = \frac{\tau_c}{|\boldsymbol{u}|}$, Eq. 1 simplifies to:

$$4u_{xx} + u_{yy} + 3v_{xy} - \frac{\tau_b u}{\bar{v}H} = \frac{\rho g h_x}{\bar{v}}, \tag{B3a}$$

$$4v_{yy} + v_{xx} + 3u_{xy} - \frac{\tau_b v}{\bar{v}H} = \frac{\rho g h_y}{\bar{v}}. \tag{B3b}$$

Although both $\bar{v}$ and τ_b are functions of u and v, this dependence is relatively weak. Bueler and Brown (2009) show that a
good approach to solving these equations, is to use two nested iterative loops: an inner loop that solves u and v for a given $\bar{v}$
and τ_b, and an outer loop that updates $\bar{v}$ and τ_b with the new values of u and v. They show that all four variables converge to

15   a unique, stable solution. Since $\bar{v}$ and τ_b are not updated in the inner loop, the derivation for the relaxation iteration is greatly
simplified. Replacing the derivatives of u and v in Eq. 3 with their discretised approximations (as derived in Appendix A)
yields:

$$4\left(u^i N_{xx}^i + \sum_{c=1}^n u^c N_{xx}^c\right) + u^i N_{yy}^i + \sum_{c=1}^n u^c N_{yy}^c + 3v_{xy} - \frac{\tau_b u^i}{\bar{v}H} = \frac{\rho g h_x}{\bar{v}}, \tag{B4a}$$

$$4\left(v^i N_{yy}^i + \sum_{c=1}^n v^c N_{yy}^c\right) + v^i N_{xx}^i + \sum_{c=1}^n v^c N_{xx}^c + 3u_{xy} - \frac{\tau_b v^i}{\bar{v}H} = \frac{\rho g h_y}{\bar{v}}. \tag{B4b}$$

Isolating u^i and v^i yields:

$$u^i\left(4N_{xx}^i + N_{yy}^i - \frac{\tau_b}{\bar{\nu}H}\right) + \sum_{c=1}^{n} u^c\left(4N_{xx}^c + N_{yy}^c\right) + 3v_{xy} = \frac{\rho g h_x}{\bar{\nu}}, \tag{B5a}$$

$$v^i\left(4N_{yy}^i + N_{xx}^i - \frac{\tau_b}{\bar{\nu}H}\right) + \sum_{c=1}^{n} v^c\left(4N_{yy}^c + N_{xx}^c\right) + 3u_{xy} = \frac{\rho g h_y}{\bar{\nu}}. \tag{B5b}$$

Defining the "centre coefficients" $e_u^i = 4N_{xx}^i + N_{yy}^i - \frac{\tau_b}{\bar{\nu}H}$ and $e_v^i = 4N_{yy}^i + N_{xx}^i - \frac{\tau_b}{\bar{\nu}H}$, and rearranging the terms in Eq. 5 yields:

$$u^i = \frac{1}{e_u^i}\left[\frac{\rho g h_x}{\bar{\nu}} - 3v_{xy} - \sum_{c=1}^{n} u^c\left(4N_{xx}^c + N_{yy}^c\right)\right], \tag{B6a}$$

$$v^i = \frac{1}{e_v^i}\left[\frac{\rho g h_y}{\bar{\nu}} - 3u_{xy} - \sum_{c=1}^{n} v^c\left(4N_{yy}^c + N_{xx}^c\right)\right]. \tag{B6b}$$

These equations now express the function values u^i and v^i in terms of the values of their direct neighbours, in the form $u^i = f(u^{n1}, u^{n2}, ..., u^n)$ this can be adapted into an SOR iteration of the form $u_i^{t+1} = (1-\omega)u_i^t + \omega f(u^{n1}, u^{n2}, ..., u^n)$. This yields the following equations:

$$u_i^{t+1} = (1-\omega)u_i^t + \frac{\omega}{e_u^i}\left[\frac{\rho g h_x}{\bar{\nu}} - 3v_{xy} - \sum_{c=1}^{n} u^c\left(4N_{xx}^c + N_{yy}^c\right)\right], \tag{B7a}$$

$$v_i^{t+1} = (1-\omega)v_i^t + \frac{\omega}{e_v^i}\left[\frac{\rho g h_y}{\bar{\nu}} - 3u_{xy} - \sum_{c=1}^{n} v^c\left(4N_{yy}^c + N_{xx}^c\right)\right]. \tag{B7b}$$

Rearranging these terms gives the equations that can be found in the UFEMISM source code:

$$u_i^{t+1} = u_i^t - \frac{\omega}{e_u^i}\left[\sum_{c=1}^{n} u^c\left(4N_{xx}^c + N_{yy}^c\right) + e_u^i u_i^t + 3v_{xy} - \frac{\rho g h_x}{\bar{\nu}}\right], \tag{B8a}$$

$$v_i^{t+1} = v_i^t - \frac{\omega}{e_v^i}\left[\sum_{c=1}^{n} v^c\left(4N_{yy}^c + N_{xx}^c\right) + e_v^i v_i^t + 3u_{xy} - \frac{\rho g h_y}{\bar{\nu}}\right]. \tag{B8b}$$

Finding the optimal value for the relaxation parameter ω, to achieve the highest possible convergence without creating numerical instability, is one of the most-studied subjects in numerical mathematics. Preliminary experiments with UFEMISM have suggested that, when ω is too high, numerical instability primarily occurs at vertices that are surrounded by sharp triangle. This suggests that, in order to optimise computational performance, ω could be made into a spatially variable field ω_i, with values depending on mesh geometry. Further preliminary experiments showed that, while this does indeed significantly improve performance, it is difficult to create a robust setup that is guaranteed to either always avoid, or otherwise cope with, numerical instability. Solving this problem is deemed to be beyond the scope of the current study.

Appendix C – deriving the iterative scheme for solving the heat equation

In order to calculate a depth-dependent temperature distribution, a vertical spatial discretisation is required. Following the approach used by many ice-sheet models, we adopt the scaled vertical coordinate:

$$\zeta = \frac{h - z}{H}. \tag{C1}$$

This guarantees that the top and bottom of the vertical ice column always coincide with the first and last vertical grid point, respectively. This coordinate transformation results in the appearance of a few extra terms in the heat equation (Eq. 5):

$$\frac{\partial T}{\partial t} + \frac{\partial T}{\partial \zeta}\frac{\partial \zeta}{\partial t} = \frac{k}{\rho c_p}\frac{\partial^2 T}{\partial \zeta^2}\left(\frac{\partial \zeta}{\partial z}\right)^2 - u\left(\frac{\partial T}{\partial x} + \frac{\partial T}{\partial \zeta}\frac{\partial \zeta}{\partial x}\right) - v\left(\frac{\partial T}{\partial y} + \frac{\partial T}{\partial \zeta}\frac{\partial \zeta}{\partial y}\right) - w\left(\frac{\partial T}{\partial \zeta}\frac{\partial \zeta}{\partial z}\right) + \frac{\Phi}{\rho c_p}. \tag{C2}$$

The different spatial derivatives of ζ follow from Eq. C1:

$$\frac{\partial \zeta}{\partial t} = \frac{1}{H}\left(\frac{\partial h}{\partial t} - \zeta\frac{\partial H}{\partial t}\right), \tag{C3a}$$

$$\frac{\partial \zeta}{\partial x} = \frac{1}{H}\left(\frac{\partial h}{\partial x} - \zeta\frac{\partial H}{\partial x}\right), \tag{C3b}$$

$$\frac{\partial \zeta}{\partial y} = \frac{1}{H}\left(\frac{\partial h}{\partial y} - \zeta\frac{\partial H}{\partial y}\right), \tag{C3c}$$

$$\frac{\partial \zeta}{\partial z} = \frac{-1}{H}. \tag{C3d}$$

The scaled vertical derivatives of a function, e.g. $\frac{\partial T}{\partial \zeta}$ and $\frac{\partial^2 T}{\partial \zeta^2}$, can be discretised to yield expressions of the form:

$$\frac{\partial T_k}{\partial \zeta} = a_\zeta T_{k-1} + b_\zeta T_k + c_\zeta T_{k+1}, \tag{C4a}$$

$$\frac{\partial^2 T_k}{\partial \zeta^2} = a_{\zeta\zeta} T_{k-1} + b_{\zeta\zeta} T_k + c_{\zeta\zeta} T_{k+1}. \tag{C4b}$$

These expressions are very similar to the neighbour functions that were used in Appendix A. We will not include their derivation here. Using these expressions to discretise the vertical derivatives in Eq. C2, and using an implicit time-discretisation, yields:

$$\frac{T_k^{t+\Delta t} - T_k^t}{\Delta t} + \frac{\partial \zeta}{\partial t}\left(a_\zeta T_{k-1}^{t+\Delta t} + b_\zeta T_k^{t+\Delta t} + c_\zeta T_{k+1}^{t+\Delta t}\right) =$$

$$\frac{k}{\rho c_p H^2}\left(a_{\zeta\zeta} T_{k-1}^{t+\Delta t} + b_{\zeta\zeta} T_k^{t+\Delta t} + c_{\zeta\zeta} T_{k+1}^{t+\Delta t}\right)$$

$$-u\left[\frac{\partial \zeta}{\partial x}\left(a_\zeta T_{k-1}^{t+\Delta t} + b_\zeta T_k^{t+\Delta t} + c_\zeta T_{k+1}^{t+\Delta t}\right) + \frac{\partial T}{\partial x}\right]$$

$$-v\left[\frac{\partial \zeta}{\partial y}\left(a_\zeta T_{k-1}^{t+\Delta t} + b_\zeta T_k^{t+\Delta t} + c_\zeta T_{k+1}^{t+\Delta t}\right) + \frac{\partial T}{\partial y}\right]$$

$$-w\left[\frac{-1}{H}\left(a_\zeta T_{k-1}^{t+\Delta t} + b_\zeta T_k^{t+\Delta t} + c_\zeta T_{k+1}^{t+\Delta t}\right)\right] + \frac{\Phi}{\rho c_p}. \tag{C5}$$

This can be rearranged to read:

$$T_{k-1}^{t+\Delta t}\left[a_\zeta\left(\frac{\partial\zeta}{\partial t}+u\frac{\partial\zeta}{\partial x}+v\frac{\partial\zeta}{\partial y}-\frac{w}{H}\right)-\frac{a_{\zeta\zeta}k}{\rho c_p H^2}\right]$$

$$+T_k^{t+\Delta t}\left[b_\zeta\left(\frac{\partial\zeta}{\partial t}+u\frac{\partial\zeta}{\partial x}+v\frac{\partial\zeta}{\partial y}-\frac{w}{H}\right)-\frac{b_{\zeta\zeta}k}{\rho c_p H^2}-\frac{1}{\Delta t}\right]$$

$$+T_{k+1}^{t+\Delta t}\left[c_\zeta\left(\frac{\partial\zeta}{\partial t}+u\frac{\partial\zeta}{\partial x}+v\frac{\partial\zeta}{\partial y}-\frac{w}{H}\right)-\frac{c_{\zeta\zeta}k}{\rho c_p H^2}\right] \quad\text{(C6)}$$

$$=\frac{T_k^t}{\Delta t}-u\frac{\partial T}{\partial x}-v\frac{\partial T}{\partial y}+\frac{\Phi}{\rho c_p}.$$

This system of equations can be represented by the matrix equation $AT^{t+\Delta t}=\delta$, where the lower diagonal α, central diagonal β and upper diagonal γ of the tridiagonal matrix A, and the solution δ, are given by:

$$\alpha=a_\zeta\left(\frac{\partial\zeta}{\partial t}+u\frac{\partial\zeta}{\partial x}+v\frac{\partial\zeta}{\partial y}-\frac{w}{H}\right)-\frac{a_{\zeta\zeta}k}{\rho c_p H^2}, \quad\text{(C7a)}$$

$$\beta=b_\zeta\left(\frac{\partial\zeta}{\partial t}+u\frac{\partial\zeta}{\partial x}+v\frac{\partial\zeta}{\partial y}-\frac{w}{H}\right)-\frac{b_{\zeta\zeta}k}{\rho c_p H^2}-\frac{1}{\Delta t}, \quad\text{(C7b)}$$

$$\gamma=c_\zeta\left(\frac{\partial\zeta}{\partial t}+u\frac{\partial\zeta}{\partial x}+v\frac{\partial\zeta}{\partial y}-\frac{w}{H}\right)-\frac{c_{\zeta\zeta}k}{\rho c_p H^2}, \quad\text{(C7c)}$$

$$\delta=\frac{T_k^t}{\Delta t}-u\frac{\partial T}{\partial x}-v\frac{\partial T}{\partial y}+\frac{\Phi}{\rho c_p}. \quad\text{(C7d)}$$

This matrix equation has to be solved for every individual grid cell. In UFEMISM, this is done with the Fortran package LAPACK; in Matlab, it can be done with the "backslash method": `T = A\delta`. We apply a Dirichlet boundary condition at the top of the column, keeping ice temperature equal to surface air temperature. At the base, a Neumann boundary condition is applied, keeping the vertical temperature gradient fixed to a value dictated by the geothermal heat flux and the frictional heating from sliding. Ice temperature throughout the vertical column is limited to the depth-dependent pressure melting point.

References

Abe-Ouchi, A., Saito, F., Kawamura, K., Raymo, M. E., Okuno, J., Takahashi, K., and Blatter, H.: Insolation-driven 100,000-year glacial cycles and hysteresis of ice-sheet volume, Nature Letters 500, 190-194, 2013

Barletta, V. R., Bevis, M., Smith, B. E., Wilson, T., Brown, A., Bordoni, A., Willis, M., Abbas Khan, S., Rovira-Navarro, M., Dalziel, I., Smalley, R. J., Kendrick, E., Konfal, S., Caccamise, D. J. I., Aster, R. C., Nyblade, A., and Wiens, D. A.: Observed rapid bedrock uplift in Amundsen Sea Embayment promotes ice-sheet stability, Science 360, 1335-1339, 2018

Berends, C. J., de Boer, B., and van de Wal, R. S. W.: Application of HadCM3@Bristolv1.0 simulations of paleoclimate as forcing for an ice-sheet model, ANICE2.1: set-up and benchmark experiments, Geoscientific Model Development 11, 4657-4675, 2018

Berends, C. J., de Boer, B., Dolan, A. M., Hill, D. J., and van de Wal, R. S. W.: Modelling ice sheet evolution and atmospheric CO2 during the Late Pliocene, Climate of the Past 15, 1603-1619, 2019

Bevan, S. L., Luckman, A., Hubbard, B., Kulessa, B., Ashmore, D., Kuipers Munneke, P., O'Leary, M., Booth, A., Sevestre, H., and McGrath, D.: Centuries of intense surface melt on Larsen C Ice Shelf, The Cryosphere 11, 2743-2753, 2017

5 de Boer, B., Stocchi, P., and van de Wal, R.: A fully coupled 3-D ice-sheet-sea-level model: algorithm and applications, Geoscientific Model Development 7, 2141-2156, 2014

Bueler, E., Lingle, C. S., Kallen-Brown, J. A., Covey, D. N., and Bowman, L. N.: Exact solutions and verification of numerical models for isothermal ice sheets, Journal of Glaciology 51, 291-306, 2005

Bueler, E., Brown, J., and Lingle, C. S.: Exact solutions to the thermomechanically coupled shallow-ice approximation:
10 effective tools for verification, Journal of Glaciology 53, 499-516, 2007

Bueler, E. and Brown, J.: Shallow shelf approximation as a "sliding law" in a thermomechanically coupled ice sheet model, Journal of Geophysical Research 114, 2009

DeConto, R. M. and Pollard, D.: Contribution of Antarctica to past and future sea-level rise, Nature 531, 591-597, 2016

Cuzzone, J. K., Morlighem, M., Larour, E., Schlegel, N.-J., and Seroussi, H.: Implementation of higher-order vertical finite
15 elements in ISSM v4.13 for improved ice sheet flow modeling over paleoclimate timescales, Geoscientific Model Development 11, 1683-1694, 2018

Cuzzone, J. K., Schlegel, N.-J., Morlighem, M., Larour, E., Briner, J. P., Seroussi, H., and Caron, L.: The impact of model resolution on the simulated Holocene retreat of the southwestern Greenland ice sheet using the Ice Sheet System Model (ISSM), The Cryosphere 13, 879-893, 2019

20 Depoorter, M. A., Bamber, J. L., Griggs, J. A., Lenaerts, J. T. M., Ligtenberg, S. R. M., van den Broeke, M. R., and Moholdt, G.: Calving fluxes and basal melt rates of Antarctic ice shelves, Nature Letters 502, 89-93, 2013

Gagliardini, O., Zwinger, T., Gillet-Chaulet, F., Durand, G., Favier, L., de Fleurian, B., Greve, R., Malinen, M., Martín, C., Råback, P., Ruokolainen, J., Sacchettini, M., Schäfer, M., Seddik, H., and Thies, J.: Capabilities and performance of Elmer/Ice, a new-generation ice sheet model, Geoscientific Model Development 6, 1299-1318, 2013

25 Gladstone, R. M., Payne, A. J., and Cornford, S. L.: Resolution requirements for grounding-line modelling: sensitivity to basal drag and ice-shelf buttressing, Annals of Glaciology 53, 97-105, 2012

Gomez, N., Pollard, D., and Mitrovica, J. X.: A 3-D coupled ice sheet - sea level model applied to Antarctica through the last 40 ky, Earth and Planetary Science Letters 384, 88-99, 2013

Greve, R.: Application of a Polythermal Three-Dimensional Ice Sheet Model to the Greenland Ice Sheet: Response to Steady-
30 State and Transient Climate Scenarios, Journal of Climate 10, 901-918, 1997

Greve, R., Saito, F., and Abe-Ouchi, A.: Initial results of the SeaRISE numerical experiments with the models SICOPOLIS and IcIES for the Greenland ice sheet, Annals of Glaciology 52, 23-30, 2011

Halfar, P.: On the Dynamics of Ice Sheets, Journal of Geophysical Research 86, 11065-11072, 1981

Hoffman, M. J., Perego, M., Price, S. F., Lipscomb, W. H., Zhang, T., Jacobsen, D., Tezaur, I., Salinger, A. G., Tuminaro, R., and Bertagna, L.: MPAS-Albany Land Ice (MALI): a variable-resolution ice sheet model for Earth system modeling using Voronoi grids, Geoscientific Model Development 11, 3747-3780, 2018

Huybrechts, P., Payne, T., Abe-Ouchi, A., Calov, R., Fabre, A., Fastook, J. L., Greve, R., Hindmarsh, R. C. A., Hoydal, O., Johannesson, T., MacAyeal, D. R., Marsiat, I., Ritz, C., Verbitsky, M. Y., Waddington, E. D., and Warner, R.: The EISMINT benchmarks for testing ice-sheet models, Annals of Glaciology 23, 1-12, 1996

Jones, P. W.: First- and Second-Order Conservative Remapping Schemes for Grids in Spherical Coordinates, Monthly Weather Review 127, 2204-2210, 1999

Kuipers Munneke, P., Luckman, A., Bevan, S. L., Smeets, C. J. P. P., Gilbert, E., van den Broeke, M. R., Wang, W., Zender, C., Hubbard, B., Ashmore, D., Orr, A., King, J. C., and Kulessa, B.: Intense Winter Surface Melt on an Antarctic Ice Shelf, Geophysical Research Letters 45, 7615-7623, 2018

Larour, E., Seroussi, H., Morlighem, M., and Rignot, E.: Continental scale, high order, high spatial resolution, ice sheet modeling using the Ice Sheet System Model (ISSM), Journal of Geophysical Research 117, 2012

Lazeroms, W. M. J., Jenkins, A., Gudmundsson, G. H., and van de Wal, R. S. W.: Modelling present-day basal melt rates for Antarctic ice shelves using a parametrization of buoyant meltwater plumes, The Cryosphere 12, 49-70, 2018

Lipscomb, W. H., Price, S. F., Hoffman, M. J., Leguy, G. R., Bennett, A. R., Bradley, S. L., Evans, K. J., Fyke, J. G., Kennedy, J. H., Perego, M., Ranken, D. M., Sacks, W. J., Salinger, A. G., Vargo, L. J., and Worley, P. H.: Description and evaluation of the Community Ice Sheet Model (CISM) v2.1, Geoscientific Model Development 12, 387-424, 2019

MacAyeal, D. R.: Large-Scale Ice Flow Over a Viscous Basal Sediment: Theory and Application to Ice Stream B, Antarctica, Journal of Geophysical Research 94, 4071-4087, 1989

Morland, L. W. and Johnson, I. R.: Steady motion of ice sheets, Journal of Glaciology 25, 229-246, 1980

Morland, L. W.: Unconfined ice-shelf flow, in Dynamics of the West Antarctic Ice Sheet, edited by C. J. van der Veen, and J. Oerlemans, pp. 99–116, Kluwer Acad., Dordrecht, Netherlands, 1987

Morlighem, M., Williams, C. N., Rignot, E., An, L., Arndt, J. E., Bamber, J. L., Catania, G., Chauché, N., Dowdeswell, J. A., Dorschel, B., Fenty, I., Hogan, K., Howat, I. M., Hubbard, A., Jakobsson, M., Jordan, T. M., Kjeldsen, K. K., Millan, R., Mayer, L., Mouginot, J., Noël, B. P. Y., O'Cofaigh, C., Palmer, S., Rysgaard, S., Seroussi, H., Siegert, M. J., Slabon, P., Straneo, F., van den Broeke, M. R., Weinrebe, W., Wood, M., and Zinglersen, K. B.: BedMachine v3: Complete Bed Topography and Ocean Bathymetry Mapping of Greenland From Multibeam Echo Sounding Combined With Mass Conservation, Geophysical Research Letters 44, 11,051-11,061, 2017

Morlighem, M., Rignot, E., Binder, T., Blankenship, D. D., Drews, R., Eagles, G., Eisen, O., Ferraccioli, F., Forsberg, R., Fretwell, P., Goel, V., Greenbaum, J. S., Gudmundsson, H., Guo, J., Helm, V., Hofstede, C., Howat, I., Humbert, A., Jokat, W., Karlsson, N. B., Lee, W. S., Matsuoka, K., Millan, R., Mouginot, J., Paden, J., Pattyn, F., Roberts, J., Rosier, S., Ruppel, A., Seroussi, H., Smith, E. C., Steinhage, D., Sun, B., van den Broeke, M. R., Van Ommen, T. D., van Wessem, M., and

Young, D. A.: Deep glacial troughs and stabilizing ridges unveiled beneath the margins of the Antarctic ice sheet, Nature Geoscience 13, 132-138, 2019

Oppenheimer, M., Glavovic, B., Hinkel, J., van de Wal, R. S. W., Magnan, A.K., Abd-Elgawad, A., Cai, R., Cifuentes-Jar, R., DeConto, R. M., Ghosh, T., Hay, J., Isla, F., Marzeion, B., Meyssignac, B., Sebesvari, Z.: Sea Level Rise and Implications for
5   Low Lying Islands, Coasts, and Communities. In: IPCC Special Report on the Ocean and Cryosphere in a Changing Climate [H.-O. Pörtner, D.C. Roberts, V. Masson-Delmotte, P. Zhai, M. Tignor, E. Poloczanska, K. Mintenbeck, M. Nicolai, A. Okem, J. Petzold, B. Rama, N. Weyer (eds.)], 2019

Pattyn, F.: A new three-dimensional higher-order thermomechanical ice sheet model: Basic sensitivity, ice stream development, and ice flow across subglacial lakes, Journal of Geophysical Research 108, 2003
10   Pattyn, F., Schoof, C., Perichon, L., Hindmarsh, R. C. A., Bueler, E., de Fleurian, B., Durand, G., Gagliardini, O., Gladstone, R. M., Goldberg, D., Gudmunsson, G. H., Huybrechts, P., Lee, V., Nick, F. M., Payne, A. J., Pollard, D., Rybak, O., Saito, F., and Vieli, A.: Results of the Marine Ice Sheet Model Intercomparison Project, MISMIP, The Cryosphere 6, 573-588, 2012

Pattyn, F., Perichon, L., Durand, G., Favier, L., Gagliardini, O., Hindmarsh, R. C. A., Zwinger, T., Albrecht, T., Cornford, S. L., Docquier, D., Fürst, J. J., Goldberg, D., Gudmunsson, G. H., Humbert, A., Hütten, M., Huybrechts, P., Jouvet, G., Kleiner,
15   T., Larour, E., Martin, D., Morlighem, M., Payne, A. J., Pollard, D., Rückamp, M., Rybak, O., Seroussi, H., Thoma, M., and Wilkens, N.: Grounding-line migration in plan-view marine ice-sheet models: results of the ice2sea MISMEP3d intercomparison, Journal of Glaciology 59, 410-422, 2013

Pattyn, F.: Sea-level response to melting of Antarctic ice shelves on multi-centennial timescales with the fast Elementary Thermomechanical Ice Sheet model (f.ETISh v1.0), The Cryosphere 11, 1851-1878, 2017
20   Pattyn, F.: The paradigm shift in Antarctic ice sheet modelling, Nature Communications 9, 2018

Pollard, D. and DeConto, R. M.: Description of a hybrid ice sheet-shelf model, and application to Antarctica, Geoscientific Model Development 5, 1273-1295, 2012

Pollard, D., Kump, L. R., and Zachos, J. C.: Interactions between carbon dioxide, climate, weathering, and the Antarctic ice sheet in the earliest Oligocene, Global and Planetary Change 111, 258-267, 2013
25   Pollard, D., DeConto, R. M., and Alley, R. B.: Potential Antarctic Ice Sheet retreat driven by hydrofracturing and ice cliff failure, Earth and Planetary Science Letters 412, 112-121, 2015

Quiquet, A., Dumas, C., Ritz, C., Peyaud, V., and Roche, D. M.: The GRISLI ice sheet model (version 2.0): calibration and validation for multi-millennial changes of the Antarctic ice sheet, Geoscientific Model Development 11, 5003-5025, 2018

Reese, R., Winkelmann, R., and Gudmunsson, G. H.: Grounding-line flux formula applied as a flux condition in numerical
30   simulations fails for buttressed Antarctic ice streams, The Cryosphere 12, 3229-3242, 2018

Reese, R., Albrecht, T., Mengel, M., Asay-Davis, X. S., and Winkelmann, R.: Antarctic sub-shelf melt rates via PICO, The Cryosphere 12, 1969-1985, 2018b

Robinson, A., Alvarez-Solas, J., Montoya, M., Goelzer, H., Greve, R., and Ritz, C.: Description and validation of the ice-sheet model Yelmo (version 1.0), Geoscientific Model Development 13, 2805-2823, 2020

Ruppert, J.: A Delaunay Refinement Algorithm for Quality 2-Dimensional Mesh Generation, Journal of Algorithms 18, 548-585, 1995

Schoof, C.: Ice sheet grounding line dynamics: Steady states, stability, and hysteresis, Journal of Geophysical Research 112, 2007

5 Stap, L. B., van de Wal, R. S. W., de Boer, B., Bintanja, R., and Lourens, L. J.: The influence of ice sheets on temperature during the past 38 million years inferred from a one-dimensional ice sheet-climate model, Climate of the Past 13, 1243-1257, 2017

Tsai, V. C., Stewart, A. L., and Thompson, A. F.: Marine ice-sheet profiles and stability under Coulomb basal conditions, Journal of Glaciology 61, 205-215, 2015

10 Uppala, S. M., Kallberg, P. W., Simmons, A. J., Andrae, U., da Costa Bechtold, V., Fiorino, M., Gibson, J. K., Haseler, J., Hernandez, A., Kelly, G. A., Li, X., Onogi, K., Saarinen, S., Sokka, N., Allan, R. P., Andersson, E., Arpe, K., Balmaseda, M. A., Beljaars, A. C. M., van de Berg, L., Bidlot, J., Bormann, N., Caires, S., Chevallier, F., Dethof, A., Dragosavac, M., Fisher, M., Fuentes, M., Hagemann, S., Holm, E., Hoskins, B. J., Isaksen, L., Janssen, P. A. E. M., Jenne, R., McNally, A. P., Mahfouf, J.-F., Morcrette, J.-J., Rayner, N. A., Saunders, R. W., Simon, P., Sterl, A., Trenberth, K. E., Untch, A., Vasiljevic, D., Viterbo,

15 P., and Woollen, J.: The ERA-40 re-analysis, Quarterly Journal of the Royal Meteorological Society 131, 2961-3012, 2005

van de Wal, R. S. W., Zhang, X., Minobe, S., Jevrejeva, S., Riva, R. E. M., Little, C., Richter, K., and Palmer, M. D.: Uncertainties in Long-Term Twenty-First Century Process-Based Coastal Sea-Level Projections, Surveys in Geophysics 40, 1655-1671, 2019

Weis, M., Greve, R., and Hutter, K.: Theory of shallow ice shelves, Continuum Mechanics and Thermodynamics 11, 15-50,

20 1999

Willeit, M., Ganopolski, A., Calov, R., and Brovkin, V.: Mid-Pleistocene transition in glacial cycles explained by declining CO and regolith removal, Science Advances 5, 2019

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