Line 2: "This paper aims at keeping classical numerics competitive in this field." This is merely a comment conveying my divergent opinion: I do not think that the competition (if there is, as I rather see them as complementary) between machine learning (ML) algorithms and process based models (classical in terms used here) is about speed. In my opinion, it is about accuracy and physical completeness. ML reproduces what it has been trained with, and it does this very fast and efficient. The particular reason why these techniques (or at least in my opinion should be) utilized is exactly to speed things up. On the other hand, ML is restricted to a given setup (fixed set of geometry and/or processes) defined by the training set. These surrogate models need new training (which tends to be computationally intensive, so no free lunch also here) to introduce new physics, often even if moving to another topography. That – in my opinion – is where the strength of process models lies – they easier can introduce improved physics and they are more generally applicable. I, though, think it is futile to try to compete in terms of speed with a ML implementation using the same setup in the training set as in a process-based model (of which complexity it ever may be), in particular, if the training phase of the ML is left out of the comparison.

I share your point of view, but my knowledge about AI-based approaches is not sufficient. In particular, I cannot assess the reliability of the "physics-informed deep learning" version published in 2023. But as a spectator, I see that the AI-based approach has motivated several researchers to consider numerical modeling of ice flow. To me, it seems that the huge numerical effort (I accept that you may have a different opinion what huge means here) of classical numerical models kept researchers off from this field for several years.

Line 5: "... a lightweight implementation of the new scheme in MATLAB." This relates to the second major point before. In my opinion, it would have been better to wait for the Python implementation to be ready before publishing. At this point in time, the reader needs a licensed software (MATLAB) to reproduce results and is left without any insight on the performance of the Python implementation to come. To palliate the first problem, perhaps the author can test with the open source software Octave and report to what extend this can be used in replacement to MATLAB to run your current version of the code?

In response to the editor's comment on MATLAB, I promised to provide the Python implementation before the final paper is published. In short:

- Everything except for the linear solver: smooth and fast
- Linear solver based on SciPy functions: poor performance
- SciPy CG solver with re-engineered MATLAB preconditioner (incomplete Cholesky with specific treatment of dropped entries: poor performance, even with Numba just-in-time compiling
- Linear solver written in C (same as in MATLAB, but simplified for this specific equation): quite good performance

In the preliminary tests, the Python version with the compiled C solver was even faster than the MATLAB version.

Line 16: "Three-dimensional simulations of the Stokes equations with a free surface (as implemented, e.g., in the model Elmer/Ice) and the shallow ice approximation (SIA) are end-members in the hierarchy of ice-flow models." May I in this connection point out that there are existing Stokes simulations of the Western part of the Alps (in fact, including Black Forest, which was not part of the investigation but the domain) spanning several thousand years around the LGM (Cohen et al, 2018 and 2023).

I remember talking with Denis Cohen about his simulations of the Rhine glacier long ago. I also remember that he told that Elmer/Ice full Stokes simulations are not much more expensive than PISM simulations. However, I think that they were still costly in relation to the spatial resolution. But now I am a bit lost what to do with your comment.

Line 17: "The SIA considers vertically averaged velocities, assumes hydrostatic pressure, and neglects all stresses arising from horizontal shearing." To my knowledge, SIA defines horizontal velocities as a function of the vertical coordinate (Greve and Blatter, 2009), but only its force- balance considers vertically integrated stresses (expressed in terms of integrals of the vertically varying horizontal velocity field along columns). So, the fluxes, but not the velocities are vertically integrated variables. Could it be that in the presented application the fact that one would have to perform a quadrature of the force balance in each column is circumvented by reducing the solution to the mass-balance and the introduction of the pre-factors f_d and f_s therein? If so, I would ask to explain what assumptions the replacement of the vertical integrals by – what occurs to me – fixed pre-factors are implicitly introduced by this procedure (e.g. on vertical distributions of enhancement factors, temperatures).

The "vertically averaged velocities" are remnant of shallow-water or Savage–Hutter equations, which I am more familiar with. Of course, the velocity is not an independent variable in the SIA, but defined by thickness and gradient of the ice surface. The vertical profile of the horizontal velocity is the integral given in Eq. A3. The fluxes are then obtained by integrating once more (Eq. A4) and the depth-averaged velocity is just flux divided by thickness. As soon as the mechanical properties of the ice become depth-dependent, the proportionalities of Eqs. A2 and A3 get depth-dependent factors. Getting f_d from the vertical profile of the deformation parameter would, of course, become more complicated then. However, I am not sure what "replacement of the vertical integrals" means. The integrals are just solved for constant (not depth-dependent) parameter values.

Line 18: "In between, there are two-dimensional approaches that account for these stresses, ..." Even Stokes can be solved in two dimensions (flow line). I would suggest to write depth-integrated instead.

Yes, of course! I even did not think about flow-line approaches because the mass balance perpendicular to the flow line is crucial in the examples I have seen so far.

Line 26: "The still limited performance of two-dimensional models arises from a combination of assuming hydrostatic pressure and the explicit time-stepping scheme implemented in almost all contemporary models (Bueler, 2023)." Like before, I would suggest using depth-averaged or integrated models, instead. Further I would ask to explain what performance is referred to? Pure computational performance or beyond that also in terms of accuracy and stability? In view of the latter (which links to my primary point of criticism), I would drop the "still", as to me it implies that the instabilities/inaccuracies described in this manuscript are something that could be overcome. In my opinion, they are built-in, as models based on hydrostatic approximation are not capable to resolve horizontal gradients in stresses that act on scales smaller than a few multiples of the ice thickness and assume small slopes (Hindmarsh, 2004).

Performance is numerical performance here for me, but this performance is dominated by the instability of the usual schemes. So not accuracy in the context of narrow valleys, steep slopes or strong sliding etc.

The other point reflects our main disagreement. I agree that the hydrostatic approximation cannot resolve horizontal gradients in stresses that act on scales smaller than a few multiples of the ice thickness. However, as explained in the general part, I strongly disagree to the opinion that the numerical issues are an inherent property of the SIA and thus cannot be fixed by an improved numerical scheme.

Line 62, eq. (3) and line 68 and eq. (5): "The factor involving the cosine of the slope angle β of the ice surface with ... is a simple correction for steep slopes." I will come back to this later also in the appendix, but to me it appears that introducing this slope correction violates a principal assumption of the SIA that the surface normal in lowest order points exactly into the vertical direction which by shallowness also would demand that $\cos\beta \approx 1 \to \beta \approx 0$. In the other extreme, for $\beta \to \pi/2$ the 5th and 8th power of $\cos\beta$ in (9) will result in a vastly vanishing pseudo diffusivity and (8) basically reduces to the geodetic mass balance. In other words, mass transport shuts down on steep slopes. That in my view is artificial and covers the fact that the hydrostatic pressure approximation is not valid on slopes significantly deviating from zero (Hindmarsh, 2004) – see point one in major critics.

Your argument with $\beta \to \pi/2$ is technically correct for a given vertical thickness. Here it happens because the thickness perpendicular to the bed goes to zero. This points towards a major disagreement (see general points and below): You say that the vertical thickness is responsible for the flow velocity, while I say that it is the thickness perpendicular to the surface. For discussing the effect of the slope correction, however, given vertical thickness is not the right point of view. It would rather be a given flux since the ice from the upper parts has to be transported. At given flux, the vertical thickness is then proportional to $(\cos\beta)^{-8/5}$, where $(\cos\beta)^{-3/5}$ comes from $\sin\beta$ instead of $\tan\beta$ and $(\cos\beta)^{-1}$ from converting vertical to surface-normal thickness. So there is no technical problem with the slope correction, and the question whether the SIA can be written in a rotated coordinate system is discussed below.

Line 67: "The factor f_d mainly depends on temperature in reality." This factor should also contain some contributions from the vertical integral of the force balance, including the temperature dependence and enhancement factors. As demanded before, I think for completeness, the exact composition of f_d as well as f_s leading to the constant values reported in the text should be written out and explained – if not in the main part, at least in the Appendix. Further, as this seems to be the only mentioning of temperature, which, by the Arrhenius factor (Greve and Blatter, 2009) has a strong influence on the ice viscosity and hence the flow: I would kindly ask how the model – if at all – treats temperature variations in the ice, in particular as I would think that during a glaciation of the Alps the thermal ice conditions significantly vary, both, spatially and in time? Would the presented model be in principle capable of including advective temperature transport?

Of course, I can replace the proportionality in Eq. A2 by the full expression

$$\frac{\partial}{\partial z} v_d(z) = A\tau(z)^n$$

with the ice deformation parameter A ($\text{Pa}^{-n}\text{s}^{-1}$). Then we will arrive at the expression for f_d as a function of A and ρ under the condition that A and ρ are vertically constant. In this case, the exponential Arrhenius factor would just occur as a factor in f_d. This would perhaps also help to become clear how to compute f_d if A depends on z. For f_s, however, this would not make so much sense because the sliding stuff is so uncertain.
Concerning temperature etc., I think that our interests differ fundamentally. From your previous work, I guess that you are interested in realistic, physically consistent, and comprehensive models. I am more interested in innovative solutions for individual model components.
Advective heat transport could be included easily as long as we are not interested in the vertical temperature profile because we could directly derive the energy flux from the ice flux. Depending on the scheme, we would just have to reduce δt not to exceed the CFL criterion. And I would add a dispersion term to take into account the vertical velocity profile. If we include the vertical temperature profile, it would become technically more complicated and I am not sure whether I would do this with such a simple model.

Line 70: "The correction is explained in the derivation of Eq. (3) in Appendix A. It should, however, be mentioned that it is less elaborate than the correction for rapid mass movements developed by Savage and Hutter (1989) and only exact if the ice surface is parallel to the bed. In particular, it does not capture the effect of steep walls in a valley." I discuss this in detail later where the method is explained in the Appendix. As mentioned before, I think the cosine in (3) is in violation of the basic assumptions behind SIA. I also do not see the parallel slab problem of a creeping shear thinning fluid in a direct connection to the Savage-Hutter theory, which is an expansion of the depth-averaged Navier–Stokes equations (i.e., including acceleration terms) for yielding granular flows, neglecting terms of order $O(\epsilon^{3/2})$ of the aspect ratio ϵ, and in contrary to the SIA, usually already defined in a locally rotated coordinate system (e.g., Zwinger et al. , 2003).

I explained briefly in the general part why I think that key point of the Savage–Hutter theory can be transferred to the SIA and is consistent with the parallel slab solution. I try to provide some more details below.

Line 80: "It should, however, be emphasized that modeling sliding along the bed is still one of the major challenges in this field. It is even questionable whether this process can be described well by the SIA or whether lateral stress components have to be taken into account." Please, refer to literature where these topics are addressed. For the latter: It is established (see literature cited in the major points section) that SIA is unsuited for fast sliding conditions.

Of course, I do not mind including the references.

Line 89: "Formally, it looks like a nonlinear diffusion equation, although it is mathematically not a diffusion equation due to the occurrence of ∇s in D. Can you please elaborate: If not a diffusion equation, what is it then?

Not for the paper: It is still a parabolic PDE in the part of the domain where $\nabla s \neq 0$ because its linearized form is parabolic. Mathematical studies on nonlinear diffusion, however, typically do not allow D depending on ∇s. The reason is that the linearized form has a different diffusivity of $\tilde{D} = \frac{\partial}{\partial |\nabla s|}(D|\nabla s|)$, which would be $\tilde{D} = 3D$ here. Then all properties (diffusion of small disturbances, Fourier criterion, ...) depend on $\tilde{D}$ instead of D.

Line 120: "The upstream scheme ensures $D = 0$ in this situation and thus avoids a permanent systematic increase in ice volume." What about the considerations on the second term of the r.h.s. of (8) at the contact line in order to avoid negative heights? There is often negative net mass balance, r, at the front of the glacier. Is it masked out? Generally speaking, to stay consistent in mass balance one should solve the mass balance or in higher order models the kinematic boundary condition together with the inequality for positive or zero flowheights (see, e.g., Gagliardini et al. 2013, section 6.5), which in my opinion should be applicable to the scheme applied in this manuscript.

There is no minimum thickness $h_{\min}$ implemented here, but it could be. And the workflow is:

1. Compute D
2. Compute r and update the ice surface
3. Solve the ice flow equation
4. Remove negative thickness values

Steps 2 and 3 could be interchanged because the gradient in r is typically small. If the thickness is negative and lower than $r\delta t$ (so more ice lost than melting), it is considered an error in the mass balance. At the front, however, this does not happen.

Line 128: "Therefore, a dynamic smoothing of the slope of the ice surface in the diffusivity $D(h, |\nabla s|)$ is the central novel idea in MinSIA. The idea was originally motivated by Airy's linear theory of ocean waves (gravity waves in an inviscid fluid with a free surface). According to this theory, oscillations in pressure and particle velocity decrease exponentially with depth below the surface. The respective depth of penetration is proportional to the wavelength. Based on this result, MinSIA uses a smoothed version of $|\nabla s|$ with the length scale of smoothing being proportional to the ice thickness h." I am not able to see a connection from a theory based on gravity waves in Newtonian fluids to the mechanical behaviour of a (strongly non-linear and highly viscous) shear thinning free-surface thin-film Stokes problem. My theory: this smoothing algorithm works because it re-introduces the basic principle of thin-film approximations to resolve horizontal gradients with respect to a large aspect ratio (i.e., over several ice-thicknesses), even if run on mesh resolutions that apparently violate this assumption. To probe that theory, I suggest to run the simulation on a to the SIA appropriate mesh resolution (> 1000 m range) without smoothing and report on instabilities of such a run. I would also ask to include a clear motivation to opt for such small horizontal resolutions of 25 m–100 m in connection with SIA. Thinking of the main motivation presented in the text, a fast solution, I would even say that increasing the resolution should work exactly in this direction as the problem size gets smaller. This links to the first in the main point of critics.

Since both reviewers do not like the motivation of the approach from Airy's theory, I can rewrite this part.

Technically, I agree to your theory. If we use a sufficiently large grid spacing, smoothing vanishes and even explicit schemes become stable. So there is no need to probe your theory technically. As discussed in the general part, our disagreement is whether the numerical instability is an inherent property of the SIA, whether the SIA becomes increasingly wrong just because we use finer grids and whether a coarse resolution that does not capture the valleys is better.

Line 163: "The preconditioned conjugate gradients (PCG) method is used for solving the linear equation system. An incomplete Cholesky factorization is used as a preconditioner, whereby the version that compensates dropped nondiagonal elements at the respective diagonal elements turned out to be particularly suitable. A direct solver is also implemented for testing." As mentioned in the second point of major critics. I would ask for a more detailed explanation of this particular choice. What method was used for the direct solver? Were these methods utilizing specific features of the hardware (shared-memory or distributed memory parallelism, vector units, accelerators, if present)? What can the reader expect if this model is run on other, more modern platforms than the CPU that was reported in the text? Are there special license conditions to utilize the mentioned methods in MATLAB?

The linear solver is not an essential part of the scheme and it should be easy to replace it by any other solver if desired. I am even not sure which direct solver MATLAB uses by default, but the direct solver is only interesting for small test examples since it requires much memory. The preconditioner was found by performing experiments with several different preconditioners and settings, but already knowing that incomplete Cholesky factorizations are not too bad for problems of this type. The C code for the entire iterative solver in the Python implementation will, of course, be open and consists of less than 100 lines.

But to be honest, I am not interested so much in the hardware-related aspects. There were no parallel-computing features used explicitly, which means that MATLAB just uses multiple cores for builtin functions. Practically, however, it was not much more than 1.5 cores on average (read from the Linux top). In principle, everything except for the steps related to preconditioning could be parallelized. So there might be a scale at which the preconditioner becomes the bottleneck, so that another preconditioner might become better.

Line 187: "As a first test, the accuracy of the solutions for different values of the smoothing factor f with respect to the reference solution ($f = 0.25$, $\delta t = \frac{1}{64}$ yr) is measured." In relation to the major point of critics, I do not think that comparison to a result obtained with the method is able to show the accuracy of the method itself. To me, it merely shows the sensitivity of the method to a change in numerical parameters and resolutions.

Since the first reviewer also asked for a simulation without smoothing, I started one. As stated in the manuscript (and also in your next comment), it is in principle too expensive. So I started it from $t = 2900$ yr and it made 60 yr so far. For $\delta t = 1/2048$ yr at $\delta x = 25$ m, there are still considerable staircase oscillations, which vanish for $\delta t = 1/4096$ yr. The maximum oscillation (over all nodes of the grid) rapidly falls below 10^{-4} m and stays there.

A bit more analysis: Revisiting Fig. 2(b), we can see that the deviation between $f = 1$ and $f = 0.5$ is about twice as big as the deviation between $f = 0.5$ and $f = 0.25$. This finding tentatively suggests that the error resulting from smoothing is linear in f. In this case, the deviation between $f = 0.25$ and $f = 0$ should be as big as the deviation between $f = 0.5$ and $f = 0.25$ (from a geometric series $0.5 + 0.25 + 0.125 + \ldots = 1$). The numerically obtained deviation between $f = 0.25$ and $f = 0$ at $t = 2960$ yr (so after the 60 yr) is even lower than this, but this may be due to the short time span. Nevertheless, the deviations between $f = 0.25$ and $f = 0$ seem to be in the order of magnitude of a few centimeters, which seems to be sufficiently small to me to consider $f = 0.25$ as a reference.

Line 209: "Since even explicit schemes are stable for sufficiently small time increments, the occurrence of the staircase oscillation must be related to both the smoothing factor f and the time increment δt." I would like to see a proof on the statement of stability for small timesteps. Either by citing literature or running the model (at least for some time) with no smoothing on the 25 m resolution. What I think, is that the oscillations are a reaction of the SIA to significant undulations with a length-scale way below the ice-thickness. Consequently, I would conclude that the oscillations are natural to the SIA (if run on resolutions that violate the shallowness assumption) and damped out by the smoothing, similar to as they would by running the SIA without smoothing on coarser meshes (which I already asked you to test) that resolve horizontal scales above the typical ice-thicknesses.

The point that the oscillations arise from undulations with a length-scale way below the ice-thickness is correct (the oscillations with the worst impact are on the scale of δx), but the rest of your statement reflects our main disagreement. As stated in the general response, the SIA is practically a diffusion equation (the fact that the linearized diffusivity is $3D$ instead of D does not matter here), which means that these undulations do not result in growing oscillations for an exact solution. Exact solutions of diffusion-type equations smooth such undulations without overshooting. The exact solution dampens such undulations even faster if the wavelength is shorter. The conversion of surface undulations into oscillations is fully a matter of the numerical scheme. This is confirmed by the simulation without smoothing and $\delta t = 1/4096$ yr reported in the previous point.

Line 267: "The simulations over the entire time span with $\delta t = 0.25$ yr took 35400 s for $\delta x = 25$ m, 6740 s for $\delta x = 50$ m, and 1290 s for $\delta x = 100$ m on an Intel Core i5-7600 CPU (3.50 GHz) from 2018." This, to me, seems to be the only information on how some sort of performance evaluation has been done. In my view, this is not enough such that the reader can get a clear picture what to expect if running the model themselves. I would ask to specify the utilization of the CPU (single core or multiple cores, as there seem to be 4 included in the reported model). If multiple cores have been utilized, please report on what parallel paradigm was deployed (shared or distributed memory). How much memory was available in the test platform and how was the saturation of the available memory at the runtime? This links to the second point of my major critics.

Yes, this is the only information, and I feel unable to provide much more information. It should be immediately clear that it is much faster than Elmer/Ice and PISM, but not a comprehensive model including heat transport etc. and with the limitations of the SIA. From my experience, it is also much faster than the implementation of the iSOSIA in the respective erosion model. And a comparison with the AI-based IGM seems to be extremely difficult to me at the moment (usage of GPU computing, time for training, ...). I can finally add the times for the Python version.

Line 288: "Figure 8 illustrates the weakness of the scheme at steep slopes for $\delta x = 25$ m and $\delta t = 0.25$ yr." I would rather say it reveals the inapplicability of the approximation than a weakness in the numerical scheme (see major point of critics).

Definitely not at this occasion. As written in the text, this applies to steep slopes with rather small ice thickness. Here the shallow-ice or slab approximation itself is ok. It is just that advection dominates over diffusion at these points and that the scheme is not better than any other scheme then.

Line 305: "At present, a MATLAB implementation of MinSIA is available under the GNU General Public License. A Python implementation is under development." As mentioned under the major points and before in this section – despite the setup being shared with an open license – the proprietary nature of the software needed to run it imposes a big hurdle for people to reproduce these findings. This all would have been no problem if the publication would have been postponed to the point in time when the announced Python version would have been available.

I hope that is point was addressed sufficiently in the response to an earlier comment. In this specific case, I could imagine that you would indeed have tested it without smoothing and a small δt. But I honestly do not know many reviewers who would have spent this work. And would if have had any effect on your opinion that everything that I wrote about partial differential equations and about the SIA is wrong?

Line 334: "A 2-element array corresponds to $r_{\max} = \infty$ *and a scalar value additionally to* $g_- = g_+$*." I am not able to understand the meaning behind this sentence. Can you please elaborate? Do you allow for infinitely large accumulation rates?*

It just means that you can omit he third entry of the array if you want unlimited ice production and that you can omit the second entry if you want the ablation gradient to be equal to the accumulation gradient. May be useful or not, but it is implemented like this.

Line 390: "Let us, for simplicity, assume that the ice surface is parallel to the bed with a slope angle β and that the z-coordinate is perpendicular to the surface with $z = 0$ at the bed." I think this is the culprit of the discrepancy about the bed-slope correction I raised in the main issues. The parallel slab example (see Greve and Blatter, 2009), which seems to be adapted here, in my view, cannot be intermixed with the derivation of the SIA. The basic assumption behind the SIA is a directly into the vertical direction pointing surface normal (Greve and Blatter, 2009). The resulting full alignment with the negative direction of the gravity is a necessary condition for the zero Cauchy stress vector at the surface leading to identically vanishing stress-components and pressure, which constitutes another necessary condition for the vertical integration of the shear stresses (Greve and Blatter, 2009) in SIA. If the integration in the vertical direction is done from bedrock, b, to the free surface h, one obtains the shear stress at the bedrock as $\tau = \rho\rho g h \nabla s$, where $h = s - b$, i.e., the total ice-depth in vertical direction. Thus, in SIA the shear stress is imposed by the hydrostatic pressure gradient induced by the horizontal change in surface elevation (with no influence of the bedrock slope at all), whereas in the parallel slab, i.e., equation (A1), the shear stress is the result of the downslope weight component of a in the rotated reference frame bed- parallel surface with in this rotated reference frame identically vanishing gradients of the free surface. As I see it, the author is setting (A1) identical to the bed-shear stress as defined in SIA, which I claim violates the basic assumptions of the SIA. If the author disagrees, I would ask to include a detailed derivation of the SIA (starting from the Stokes equations) in a rotated reference frame and show that the derived equations still comply with the lowest order approximation, even for large bedrock angles, as they occur in the studied topographies of the Alps and even the Black Forest.

For simplicity, linear Stokes equation with constant viscosity,

$$-\nabla p + \rho \vec{g} + \eta \Delta \vec{v} \;=\; 0,$$

and the same direction of the velocity everywhere,

$$\vec{v} \;=\; f(x, y, z)\vec{e},$$

with a (constant) unit vector $\vec{e}$. Then

$$\Delta v \;=\; (\Delta f)\vec{e}.$$

Define a normal vector $\vec{n}$ perpendicular to the velocity, so $\vec{e} \cdot \vec{n} = 0$ and obtain

$$\nabla p \cdot \vec{n} = \rho(\vec{g} \cdot \vec{n}),$$

which means that the pressure gradient perpendicular to the velocity is the component of gravity perpendicular to the velocity. I leave the rest of the derivation, including the generalization to the nonlinear flow law, to you. But for sure: hydrostatic perpendicular to the velocity and not necessarily in the vertical direction.

For completeness, the hydrostatic pressure at the bed (valid for Stokes and Navier–Stokes) in Cartesian coordinates (see, e.g., Hergarten 2024, doi 10.5194/gmd-17-781-2024) is

$$p_b = \frac{\rho g h}{1 + \nabla s \cdot \nabla b}.$$

Your SIA version ($p_b = \rho g h$) is obtained for $\nabla b = 0$ and the slab version (my correction terms) for $\nabla b = \nabla s$, which leads to

$$p_b = \rho g h \cos^2 \beta.$$

Just to mention: Hutter (1983, 0.1007/978-94-015-1167-4) defined $\nabla b \approx \nabla s$ as a condition.