Review of ‘Universal Differential Equations for glacier ice flow modelling’ by Bolibar et al.

Doug Brinkerhoff

July 7, 2023

1 Summary

In this paper, the authors describe embedding a neural-network-based parameterization of the viscous rate factor within a shallow ice model and training it on a semi-synthetic dataset with the aid of reverse mode automatic differentiation. I think that this is generally an important topic and that this paper is a useful step in the direction of modelling/ML hybrids. I don’t have many objections with respect to the technical content of this work, however I do think it suffers from a few significant misunderstandings of its own methods, of referenced works, and of the broader context into which these results fit. These issues, along with some more minor technical corrections, are outlined below.

2 Comments

L40 It should be noted here that adjoints have been known about and used in glaciology since at least Doug Macayeal’s 1993 paper on optimal control methods (MacAyeal, Douglas R. ”A tutorial on the use of control methods in ice-sheet modeling.” Journal of Glaciology 39.131 (1993): 91-98.). While utilizing neural networks to do things in glaciology is a bit new, the general notion of ‘reverse mode AD’ is not. (It would also be worth citing Tarasov (2012, ”A data-calibrated distribution of deglacial chronologies for the North American ice complex from glaciological modeling.” Earth and Planetary Science Letters 315 (2012): 30-40.) as an early example of NN surrogates in glaciology.

L56 – 64 I find this paragraph to be quite confusing on account of the use of ‘scalar parameters’. Many inversion techniques (including the Macayeal paper listed above) are built around finding spatially distributed parameter fields, which are not scalar. Indeed, different forms of regularization in such fields often correspond to a Gaussian process functional prior over spatial (and in some cases temporal) parameter fields. Furthermore, what does ‘reduced to the current structure of the mechanical model mean’? As a side note the reference to Brinkerhoff (2016) is questionable: that work uses no gradients but rather very basic MCMC to find distributions over parameter values. However, there are many other good examples of differentiable ice flow models.

L67 I think it’s perhaps a bit of a stretch to say that the neural networks in this work learn ‘spatiotemporal variability’. What they are learning is a parameterization of ice softness as a function of surface temperature based on a number of training examples that happen (or are designed) to span a practical domain and range of said function.

L84 I think it’s too early to talk about assuming that $C = 0$ here: the equation remains a “diffusion” equation regardless of this choice and it is not necessary in order to be able to define a D.

L99 ‘propriety’ → ‘property’

Eq. 4 I suggest using **u** or some other symbolic choice to make clear that velocity is a vector (rather than V).

Sec. 2.2 I think that this section needs to be moved after 2.3, since it is not yet clear what the embedded neural net is supposed to be representing in this case. Furthermore, the function SIASolver is never defined. Does this yield a thickness or a velocity? Since ice velocities are prognostic, why should this function take V_0 as an argument, when there cannot be any dependency on this argument? Should this be an H_0?

Sec. 2.2 I really struggle with casting this problem as one that is time dependent and I am not sure I see the value in doing so in this work. The reason for this is that the inversion targets are velocities, which are diagnostic of thickness. Since thickness at some time is being considered as known, then one can simply compute the velocity at this time and use it in computing a loss, updating the parameters of the embedded neural network parameterization of hardness, etc. There is simply no need for a time dependent solver in the procedure as written (although, to be sure there is utility to time-dependent adjoints of glacier models). I think that the consequences of this independence are readily apparent in the authors results in the sense that they find no trouble in recovering their chosen ice hardness parameterization even with mis-specified surface mass balance rates: the reason for this is that the mass balance doesn't really affect the velocity and thus doesn't affect the recovered parameterization. I think that the authors allude to this themselves around L324, but I would like to see some more robust justification for why all of the fancy stuff is necessary here. Finally, as a suggestion for an augmentation that might make all of this a little bit more compelling, is it possible to do all of this with thickness as the predicted value rather than velocity?

L140 I think that the phrase 'spatial' should be dropped here. This is trying to recover $A(T_s)$. One could imagine a circumstance in which T_s is a function of space internally to each training example, but it does not seem to be the case here (based on Fig. 3, although I could misunderstand this).

L144 I think that this part is confusing: the authors write that their function for A ignores ice temperature, but isn't that what the surface temperature is supposed to proxy? This section is generally a bit unclear and could benefit from improved notation. For example, I suggest using the symbol T_s for surface temperature, to make it clear that this is what is being fed into the parameterization.

L162 Why 'timestamps' and not just 'times'?

Sec 3.1 Typically model runs require some type of spinup because the physics and various data products are not all self-consistent, leading to unrealistically large transient behavior as all of these things equilibrate. Was that the case here? Does it influence the results?

L176 I know that the CFL is mentioned in Appendix A, but it would be worth summarizing the methods used to ensure time-stepping stability conditions are satisfied here.

177–180 Why is noise applied to A rather than to the observed velocity? It's not A that is being observed, and thus it makes little sense to simulate noise in A.

L181 Most readers will not be able to infer what (1,3,10,3,1) means when describing neural networks. This either needs to be expanded here, or a discussion of neural network architecture choice should be added as an appendix.

L182 Once again, I think that some clarification of what is actually happening here is in order: the authors are using a very common formulation of a numerical ice flow model in a standard way. The novelty here is the replacement of a static rate factor A with a function $A_\omega(T_s)$ that is parameterized via a function that happens to be quite flexible (a neural network). I don't think that this is semantically equivalent to saying the problem is highly constrained by the PDE nature of the ice flow model.

L186 Why is the final sigmoid necessary? What happens without it? I can understand the desire to impose positivity (which could perhaps more easily be effected by log-transforming A, as is commonly done), but why should there be an upper bound?

L195–198 I don't understand this section. Why does adding a source term in the ODE produce instability? What is the H matrix? Are you saying that you're adding the mass balance after the integration of the flow equations?

L199–205 How long are these runs for? Is there sufficient geometric change over this time period to warrant such a detailed approach (and are the changes large enough to significantly affect A)? Shouldn't A respond over fairly long time scales, since it's ultimately ice temperature rather than surface temperature that controls flow?

Sec 3.4 I think that this section is too deep with respect to AD to be of use to glaciologists reading this, while being too shallow to be of use to AD practitioners. I suggest either delving deeply into a software-engineering type study of the influence of different AD approaches or to cut this section and simply describe the approach that was actually used. This goes for Appendix B as well, which is mostly a textbook definition of finite differences and the adjoint method.

L254–255 Referencing the potential broader impacts of cloud computing is fine, but a statement like this needs to be backed up with some evidence: it is not immediately obvious that the cloud improves scientific equity.

Fig. 3 This figure begs the question: why use a neural network for parameterizing $A(T)$. It is clear that a quadratic or exponential would have worked just as well. Indeed, if the authors had predicted the log of A, then it is likely that a linear model would have fit these data just as well as the NN. I appreciate the potential for generalization of the NN approach, but an ablation study with simpler models might be helpful here.

L4.1 Again, because velocity is diagnostic, this is not a surprising outcome.

L316 There is a substantial literature on the joint inference of traction and rheological parameters (typically in a Bayesian framework which allows for a quantification of induced covariance between parameters). In summary, this inversion is not necessarily ill-posed 'by nature' because there is scale separation between different processes.

L324 This description of the seasonal cycle of glacier velocities is perhaps oversimplified: many glaciers exhibit minimal velocities at the end of summer and speed up during the winter. Maybe reword to express a bit more nuance?

L329 I don't understand the use of 'initial conditions' here. If we're not doing time evolution, then the conditions aren't really 'initial', they're just the geometry.

L335 It would be nice to see some references that illustrate the so-called 'equifinality problem'.

Sec. 5 Throughout this work, the uncertainty in inferred parameters is not addressed. I think that there should at least be a discussion of the potential implications of such and avenues for providing a more rigorous uncertainty quantification.

Sec. 5.2.1 This editorial on AD approaches is not relevant to the current work.

L374 The authors seem to have a misunderstanding of PINNs, which involve positing a neural network as the PDE solution and then using a point collocation method to adjust the solution so as to minimize the solution residual in some norm. As such, a PINN is *not* a surrogate, but rather is a numerical method for solving PDEs in the same vein as FEM or FD and does not require 'training data' in the way that it is usually understood in the ML literature. In contrast, the work of Jouvet (which is indeed a surrogate, but which does not employ PINNs in the described way) trains a CNN to operate as the approximate solution operator to the ice flow equations from many examples of solutions generated by a ‘normal’ ice flow model (or perhaps by using a PINN).

Sec. 5.2.3 I do not understand this section.

Eq. A9,A10 I don’t understand the how these conditions related to step size choices. Is there another reference that might illustrate this point more clearly?