Review of egusphere-2026-5523: “Cheap and accurate higher-order ice flow emulator for mountain glaciers”

Summary

The authors present DahuNet, a convolutional neural network (CNN) that emulates higher-order ice flow over grounded ice and, unlike earlier IGM emulators, needs no online retraining. Inference is therefore much cheaper, and accuracy improves slightly. In two 500-year experiments, the emulator tracks the reference solution closely, which offers the prospect of simulating many mountain glaciers with higher-order dynamics at low cost.

The data-generation pipeline is the paper's main strength: it balances efficient geometry generation with accurate reference velocities, at a scale far beyond earlier catalogues. I also enjoyed the analysis, which shows (1) that the physics-enriched input explains most of the accuracy gain, with the variational term playing a second-order role, and (2) that velocity errors feed back on geometry in a way that keeps them bounded instead of letting them grow.

The potential impact is high, but the manuscript currently claims more than it demonstrates. The training distribution (spatially uniform rheology and basal resistance, grid spacing of 50–250 m, land-terminating glaciers only) make this a convincing proof of concept rather than a general-purpose solver, and the paper should say so explicitly. I also have concerns about one governing equation, the attribution of the accuracy gains, and data availability.

These concerns need to be addressed before a possible publication, which I am keen to see. I wish the authors the best of luck for the rest of the publication process!

General comments

1. Ice viscosity and basal resistance are spatially uniform, there is no thermomechanical coupling, and subglacial hydrology is absent (l. 496–505). These are strong assumptions, far from real-world conditions, and they deserve more discussion than the few lines they currently receive. They also bear directly on the claims of applicability to regional and global ensembles (l. 20–21, 520–523). This makes the work a proof of concept. I suggest stating this in the title, the abstract, the aims in the introduction, and the conclusions.

2. I understand that the 180 × 180 domain is only a training convenience. Is that correct? If so, the main geometric constraint of DahuNet is the grid spacing: Δx enters as an input channel and was only sampled between 50 and 250 m. Outside that range, users cannot choose their resolution without extrapolating.

3. As written, $\tau_b \propto (|u_b|/u_{ref})^m$ with $m = 3$ makes basal stress grow with the cube of sliding speed. A Weertman law gives $\tau_b \propto u_b^{(1/m)}$, which is also what the sliding proxy $|\nabla s|^3 h^3$ (l. 247, 541) assumes. Please check whether this is a typo or affects the training physics.

4. The targets use MOLHO (MOno-Layer Higher-Order) with $N_z = 2$, a vertically integrated approximation of Blatter-Pattyn (BP) with a vertical-shear profile derived from the shallow-ice

approximation (SIA) (l. 128–131). The title, abstract and conclusions should make clear that this, not the full three-dimensional BP problem (l. 35), is what is emulated.

5. DahuNet differs from the CNN baseline in its five feature fields, but also in its activation function (GELU instead of LeakyReLU) and its dense lift layer (l. 549–550). Lines 299–301 attribute the improvement entirely to the features Ψ_5. An ablation (for instance, the CNN with Ψ_5 added) would make this attribution convincing.

6. The text states that the variational penalty improves both architectures, mostly in the upper tail (l. 296–297, 302, 306). Table 1 shows the opposite for the CNN: the 90th-percentile errors increase (surface speed 6.5 → 7.2 m yr^{-1}, basal speed 5.6 → 6.1 m yr^{-1}). Please reconcile the text with the table.

7. The speed-up relative to retrained pinnbp is the main practical achievement, yet it receives only a few lines (l. 371–376). A figure comparing run time and memory use of Dahu_λ, pinnbp with and without retraining, and the reference solver would make this result visible.

8. The decomposition in Sect. 3.3 is a really valuable analysis. However, the mechanism described at l. 459–466, where a positive speed bias thins the glacier and reduces the driving stress, follows from the diffusive character of mass conservation and would apply to any reasonable flux error.

9. Since the CNN you present cannot be used for realistic simulations of mountain glaciers, I argue that, besides the neat proof of concept, the value of your work for the community is mostly the data you generated. This is a very time-consuming task, much more than the network design, as rightfully reflected by the manuscript. In this light, I can't stress enough what a pity it is to review an article that focuses on data-based methods without having access to the data. A size above 100 GB is not a strong argument against archiving: precision can be reduced (for instance to Float16 or rounding options) with little loss and repositories such as Zenodo accept large deposits or several uploads. Finally, I sometimes have requested data that was supposedly available upon request and got negative answers, which is a huge hurdle for research and shouldn't apply to the very useful data you have generated.
The repository contains the training module but, as far as I can tell, not the data-generation pipeline (terrain sampling, climate forcing, snapshot solves), which should also be released. The code is cited as a live GitHub repository. Please archive the exact version used, together with the trained weights, under a DOI.

10. The discussion largely repeats the results. Some suggestions to make it more useful:

1. Role of the synthetic training set: emphasise that the dataset represents most of the effort and could serve as a basis for follow-up work, and discuss that data quality is often the main bottleneck in machine-learning emulation.

2. Architecture: the physics-enriched input made a large difference. Real-world applications with spatially varying friction and rheology will likely need similar features, which raises the question of how to extend this approach.

3. Computational implications: this subsection is already informative.

4. Limitations: expand along the lines of the comments above. Eight 5 × 5 layers give a footprint of 33 × 33 cells, which spans 1.65 km at 50 m but 8.25 km at 250 m, so the physical context available to the network changes with resolution.

Specific comments

1. L. 1: "Cheap" reads as informal for a title. "Efficient", "accelerated" or "fast" would be more appropriate, for instance: "Accurate and fast emulator of higher-order ice flow for mountain glaciers: a proof of concept".

2. L. 13: "Preferred" sounds arbitrary. Please tie it to the scores in Table 1.

3. L. 22: You are missing an opportunity to introduce the high impact of mountain glaciers retreat in a warming climate.

4. L. 33: Please add a reference to support this statement.

5. L. 58–60: This is hard to follow. A suggestion: "The geometry is evolved with the cheaper online neural solver. For a subset of snapshots, we solve the full dynamics problem offline and use the solution as a training target."

6. L. 64–66: I would consider removing this sentence. It is very specific and does not add much at this point.

7. L. 68–70: Consider moving the research question one paragraph earlier, so that the question comes before the approach. Otherwise, l. 54–67 come somewhat unexpectedly.

8. L. 105–111, Eqs. (3)–(4): Eq. (4) uses Z_H, which is only defined at Eq. (9). Please state the relation between I_H and Z_H when the mapping is introduced, including what Z_H is for M^id.

9. Notation in general: I would recommend naming the mappings after the models, for example M^molho, M^pinnbp and M^dahu (with the corresponding θ), and U^molho instead of U^ref. "Identity mapping" is confusing, because the identity is $x \mapsto x$, which is not what is meant here.

10. L. 126–127: The number of trainable variables is indeed fixed, but the cost of a forward pass still scales with the number of grid cells. Please clarify that "cheap" refers to the optimisation, not the evaluation.

11. L. 161: Please state that the sampled Δx range defines where the emulator is valid. Since Δx is an input channel, applying the emulator outside 50–250 m means extrapolating.

12. Fig. 2: The panels are not labelled, although the caption refers to a–d, and the fonts in the right-hand panels are too small.

13. L. 180–181: With uniform friction and viscosity, I wouldn't call these glaciers dynamically plausible.

14. L. 225–226: I am unsure how the boundary is treated. Can cells with i = 180 be covered by ice? If so, cropping 10 cells (only 500 m at 50 m resolution) may be too little to remove boundary effects.

15. L. 240 onwards: Augmenting the input with these quantities makes physical sense, and the results are convincing. I was very pleased with this part of the manuscript.

16. L. 249–250: Adding input channels increases the size of the first (dense lift) layer, so a few additional trainable parameters are introduced (5 × 48 weights in the default configuration). The effect is negligible, but the statement should be corrected.

17. L. 260 onwards: The additional loss term is physically well motivated, and its effect on the loss is a nice result!

18. Table 1 and following: Please use a different name than "CNN" for the baseline, since DahuNet is also a CNN.

19. Fig. 3: Please label the panels. Since the whiskers extend to 1.5 × IQR (interquartile range), they add no information, if I understand this correctly. Showing the 5th and 95th percentiles instead would be more informative.

20. L. 316–317: I am glad the authors investigated this!

21. L. 348, 365: "a few m yr^{-1}": please give the actual values.

22. Fig. 5 and l. 376: Please align the name pinnbp with the notation used for the mappings, and specify the retraining schedule used for pinnbp, since the cost comparison depends on it.

23. L. 523–524: Modellers usually want to have non-uniform friction and ice rheology. The work is a valuable first step towards such general use and should be presented as such.

24. L. 549–550: Please add references for the activation functions (GELU, LeakyReLU).

25. Appendices: The appendices are rarely referenced in the main text. Please point to them where relevant, so that readers know additional information is available.

26. Appendix A4: The adaptive weighting relies on several choices (ρ, r_max, α, K_λ) that were not tuned. This is acceptable, but it should be acknowledged and briefly discussed.

Technical corrections

- L. 67: "coupled back to mass conservation" → "coupled with the mass-conservation equation".
- L. 109: "a differentiable mapping of one of two types" → "a differentiable mapping".
- L. 204: unresolved citation "(?)".
- L. 214–215: "have had a chance to grow" → "have grown".
- L. 216: "frozen" → "fixed".
- L. 349–350: Suggestion: "reaches about 10 m, equivalent to 5–10 % of the mean ice thickness of 100–200 m".
- L. 353: "physics informed" → "physics-informed".
- L. 393: the opening quotation mark around 'replay' is inverted.

- Spelling: British and American spellings are mixed, for example realisation/realization (l. 176, Fig. 2), discretisation/discretization (l. 359), initialization (l. 181), optimizer (Fig. A1). Please harmonise.
- Both "Dahunet" (l. 529) and "DahuNet" are used.
- References: several entries contain a doubled DOI prefix ("https://doi.org/https://doi.org/…"), for example Cook et al. (2023), Gregov et al. (2026) and Zekollari et al. (2022).