Review: "Computationally efficient subglacial drainage modelling using Gaussian Process emulators: GlaDS-GP v1.0" by Hill et al.

Reviewer: Jacob Downs

This work outlines a Gaussian Process based emulator of the GlaDS subglacial hydrology model. Inputs to the emulator consist of 8 scalar model parameters, while outputs include either spatio-temporal flotation fraction fields or scalar metrics describing bulk properties of the subglacial drainage system. Performance of the emulator is thoroughly examined by comparing emulators using different training sets as well as using different numbers of principal components to represent flotation fraction fields.

Overall, this manuscript is well written, the methods are clear, and I appreciate the rigorous evaluation of the emulator's accuracy. In light of this, I believe this work represents a good contribution to the field. However, I believe that this work would benefit from a more detailed discussion of how the GP emulator compares to existing emulators. In particular, I think the discussion should outline more of the key differences between this work and Verjans and Robel [2024], including differences in their potential use cases.

The authors should outline in specific terms where their emulator could be applied versus other existing methods. For instance, is this purely a tool for calibrating subglacial hydrology parameters or assessing sensitivity? Is there a path for using this as a substitute for GlaDS to predict effective pressure when coupled with an ice sheet model? What do the authors intend to do with this emulator, or what might future work on the emulator entail? As someone interested in this space, I was hoping the authors might delve more deeply into some of these topics in the discussion.

Line 52

Might benefit from rewording to"The Gaussian Process emulators we develop take subglacial drainage model parameters as inputs and predict spatially and seasonally resolved flotation fraction ... "

Line 63

Maybe this could be reworded as as "where the radius of channels is modeled as a balance between the creep closure of ice and opening by melt. "

Line 65

"The continuum distributed (sheet) drainage system is defined on the mesh nodes, with possible channel locations defined by element edges."

This could be clarified. Maybe you could say that variables describing the distributed system are represented by linear finite elements with degrees of freedom on mesh nodes? This would help clarify that the distributed drainage system isn't "confined" to the mesh nodes, but that's where the degrees of freedom are defined.

Line 83

"For details on Gaussian Processes"

Line 130

Please mention why there are d+1 hyperparameters. Also, depending on the kernel, this number could vary.

Line 170

That seems reasonable. If I understand, would it be fair to say that you hope that much of the spatial / temporal complexity is captured in the principal, components while the response of the coefficients of the principal components to variations in parameters is expected to be smooth?

Line 285

Interesting, I like the commentary on the first couple principal components, and I think interpretability is a really nice advantage of this method.

Line 321

"Using 8 PCs reduces the height..."

Figure 4

I'm not sure what is meant by 95% prediction interval in c and f. Is this the prediction uncertainty of the emulator integrated through time and space? Also, when considering the prediction error, is this accounting for uncertainty in θ (sampled via MCMC) as well as the GP prediction uncertainty? Or is it just the latter, and you are using the most probable estimate for the hyperparameters? Presumably the advantage of using MCMC on θ as opposed to maximum likelihood estimation is to characterize its effect on uncertainty as well?

Line 340

Before discussing results for the test cases with different levels of error, please introduce what the "high-error" and "median-error" simulations are. I think I found this information in the figure 6 and 7 captions, but it would be helpful to include in in the text.

Figure 9

It's difficult to see the differences in the median error across different numbers of simulations due to the scale of the outliers.

Section 6.4

I think some of this discussion could be expanded to highlight more of the nuances of the different approaches. For example, the emulator in Verjans and Robel [2024] aims to be fairly general purpose, and its more of a one-to-one substitute for GLaDS. Hence, I see their neural network based approach as fundamentally different from your yours in its intent. This also makes the comparison of the number of parameters difficult as the input / output spaces in Verjans and Robel [2024] is very broad, meaning more parameters are likely needed.

There are certainly a number of appealing elements to your GP approach, including interpretability, the speed at which it can be trained, and how you also obtain uncertainty estimates without additional work. But I feel like its difficult to directly compare your emulator to Verjans and Robel [2024], and I have a hard time seeing the two emulators being used in the same way. In this sense, I see your emulator as being far more comparable to Brinkerhoff et al. [2021].

Line 481

To me, saying that IGM enforces conservation of momentum sounds as if it is enforced strictly. Although IGM uses a physics-based loss function, conservation of momentum is only upheld approximately. Contrast this with something like Horie and Mitsume [2024], in which a value is strictly conserved in a neural network.

Section 6.6

I think this manuscript would significantly benefit from elaborating on specific use cases for this emulator. For instance, do you see the emulator being used more or less as is, or do you think the value of this work is in the general approach that you present? You mention uncertainty in future sea level rise, but not a clear application of the emulator for this purpose.

There is a pretty clear use case for using the emulator to do Bayesian calibration of subglacial hydrology model parameters, but what other use cases might it have? Do you see a path forward for coupling effective pressure fields from the emulator to an ice sheet model? Clarifying the intended use of this emulator or more concrete pathways for other applications would really strengthen the discussion.

References

Douglas Brinkerhoff, Andy Aschwanden, and Mark Fahnestock. Constraining subglacial processes from surface velocity observations using surrogate-based bayesian inference. 67 (263):385–403, 2021. ISSN 0022-1430, 1727-5652. doi: 10.1017/jog.2020.112. URL https://www.cambridge.org/core/journals/journal-of-glaciology/article/constraining-subglacial-processe
Publisher: Cambridge University Press.

Masanobu Horie and Naoto Mitsume. Graph neural PDE solvers with conservation and similarity-equivariance. 2024. URL https://openreview.net/forum?id=WajJf47TUi.

Vincent Verjans and Alexander Robel. Accelerating subglacial hydrology for ice sheet models with deep learning methods. 51(2):e2023GL105281, 2024. ISSN 1944-8007. doi: 10.1029/2023GL105281. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2023GL105281. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2023GL105281.