Response to Referee 1

for "Deep learning of subgrid-scale parametrisations for short-term forecasting of sea-ice dynamics with a Maxwell-Elasto-Brittle rheology"

Finn, T.S., Durand, C., Farchi, A., Bocquet, M.,
Chen, Y., Carrassi, A., Dansereau, V.

19th April 2023

RC: Reviewer Comment, AR: Author Response

RC: The authors present an idealised study of sea-ice fracture in a channel due to wind forcing, demonstrating that a neural network (NN) is able to significantly reduce errors of a lower-resolution version of the physical model with respect to a higher-resolution version of the same model for 10-minute forecasts. They conclude that the NN has learned the tendencies from the unresolved scales in the lower-resolution model, and can therefore be used to parameterise these unresolved scales.

I appreciate the originality of the work and the level of detailed analysis it provides. It fits well with current efforts in the community to use machine learning for parameterisation of unresolved scales in geophysical models. However, given the very idealised setup, I have some concerns about the wider applicability of the results. Below, I spell that out in comments which I would like the authors to address before publication:

AR: We thank referee 1 for the constructive feedback on our manuscript, especially with respect to the bias correction and a possible overfitting. In the following, we discuss the raised concerns and what we plan to change in our revised manuscript.

RC: There are a number of very strong idealisations and restrictions in the setup of this study: a) it is a so-called "perfect model" study, i.e. the performance of the lower-resolution model with/without NN corrections is assessed against a "truth" which is a simulation of that same model at higher resolution, without involving any observations or simulations from a different model; b) The forecast lead times considered are extremely short for most real-life weather and climate applications (only up to 1h); c) spatial domain is a simple rectangular channel; d) no treatment of sea-ice thermodynamics. Given these very strong idealisations and

restrictions, one would hope for results that are a bit more convincing than the ones presented. I have concerns about whether the methods presented will be useful in a more realistic context, where each of the above assumptions will need to be relaxed. Can the authors please add some in-depth discussion (or even preliminary analysis) about what they think will happen if their methods are applied in a more realistic context?

AR: Our study is designed to be a proof-of-concept. As sea ice imposes novel challenges for neural networks, it was previously unknown if model error corrections in this form are possible at all for sea-ice modelling. We think this study shows that there is indeed a huge potential for hybrid modelling of sea ice. Given the limited scope of this study, we have decided to apply such simplified settings, far from settings in operational forecasting or projections. For example, we have used the twin experiment setting to prove our points and to cheaply generate a known truth. If realistic model error corrections would be trained with twin experiments, the neural network would learn to emulate the fields from the higher resolution, so, instantiations of already known processes. Consequently, we believe that the true potential lies in the possible learning of model error corrections from observations, which is beyond the scope of our proof-of-concept. Furthermore, the model error correction is designed to correct model errors as soon as possible, before they have a too large impact on the forecast. This is why we concentrate on such short forecast lead times of up to one hour, although they might be far from operational settings. In further studies, with more realistic setups, we will investigate the impact of the model error corrections in longer forecast lead times.
To take this concern into account, we will strengthen the proof-of-concept character of the study in the introduction. Additionally, we will discuss more steps what might follow towards more generalised and realistic settings.

RC: Figure 7 and the corresponding text makes me wonder how much of the error reduction achieved by the NN is actually due to correcting the bias (i.e. mean error) of the low-resolution simulation w.r.t. the high-resolution simulation. Can the authors please provide some analysis to quantify the contribution of bias to the overall errors, with and without the NN corrections? For instance, one could just decompose the mean squared errors shown in the manuscript into squared bias and variance of the errors. I am asking this because there is a range of other methods to treat biases (e.g. a-priori by tuning the model, and a-posteriori by subtracting them from the forecast before further analysis). These methods are often simpler than the machine-learning approach and are in wide use in the weather and climate community. Utilising a complex and costly machine-learning approach only pays off it is clearly superior to other available methods.

AR: As shown in Appendix B, Table B1, we made tests where we simply correct the constant model bias. The performance of this bias corrected model is almost the same as for the raw model without any correction. Most of the model error is hence temporally and spatially variable and cannot be corrected by a simple bias correction. Additionally, we made the tests with a rather small neural network, where only one multi-scale convolutional layer is used. We expect that even simpler methods, e.g., a linear regression, perform worse than this small neural network. Additionally, there are many possibilities with neural networks that we have not taken into account, e.g., generating stochastic parametrisations with correlations learned from data. Therefore, we believe that this study indicates the potential for hybrid modelling, which would be otherwise unachievable.

In the discussion part, we have tried to clarify the possibilities with neural networks. In the revised manuscript, we may further clarify some of these possibilities.

RC: Following up on the previous comment, I would like the authors to comment on potential overfitting of the NN in their methods. If I did the maths correctly, there are about 4500 degrees of freedom in the lower-resolution physical model (9 variables times 500 grid points). As stated on line 197, the NN has 1.2 million trainable parameters. So one could argue the NN has orders of magnitude more degrees of freedom than features it is learning from or results it is predicting. I am not an expert on machine learning, but that strikes me as odd - could the authors please comment on that? I would also like to see some quantitative analysis on the risk of overfitting.

AR: We agree that a single low-resolution field has only 2558 degrees-of-freedom (DOFs). Compared to this number, the number of parameters in the neural network (1.2×10^6) seems to be very high. However, the training dataset has 2558 × number of samples DOFs. In the end, this sums up to around 12.3×10^6 million DOFs, an order of magnitude larger than the parameters in the neural network. During training, the scores in the validation dataset have been smoothly improving without a sign of overfitting. Furthermore, we have made new experiments (Fig. 1), where we have only used a fraction of the data for training. Even in our most extreme case with only 10 % of the data (480 samples), the model is not overfitting with smoothly decreasing MSE and MAE, and we achieve a nice scaling of the performance with the number of samples. We attribute this behaviour to the projection into Cartesian space and to the strategy of learning a correction for all variables at the same time. Caused by the projection from Cartesian to triangular space, the information content of the extracted features is reduced. Additionally, learning for multiple outputs at the same time acts as a type of regularisation for a single output, the network has to find features that suits all outputs.

We will introduce a new subsection in the additional results part of the Appendix with the discussion of Fig. 1 that indicates no overfitting during the training of the neural networks.

RC: Please revise the presentation of the methods, this is not sufficient in

some places, and difficult to follow in others. See technical comments.

AR: Based on this raised concern and the other review, we have decided to rearrange the presentation of the methods. The sea-ice model and the neural network will be discussed in a hopefully easier language, explaining rather the reasoning behind the different components, instead of the technical details. The more technical explanations will be moved into an Appendix. We will replace Section 2 with a Section where we first introduce the model error correction problem from a mathematical standpoint (previously Section 3.1), then a shortened explanation of the sea-ice model and, finally, an introduction into the used twin experiments. Section 3 about the neural network will be more concise and will focus more on the reasoning behind the different components.

RC: I am afraid I do not quite understand the motivation why a projection to a Cartesian grid is needed (Section 3.2). It seems to complicate the methods unnecessarily. Can the authors please clarify the motivation for doing this, and what the feasibility/implications would be of doing the analysis on the original triangular grid? Is this just a reflection of the fact that the standard machine-learning libraries for spatial analysis cannot deal with non-Cartesian grids?

AR: Compared to more “classical” neural networks, so-called multilayer perceptrons, by construction, convolutional neural networks (CNNs) are biased towards localised features, motif extraction across all grid points, and a directional dependency. Additionally, the backend libraries e.g., TensorFlow and PyTorch, are optimised for image processing. CNNs are hence especially efficiently implemented for Cartesian spaces. Although there are different convolutional architectures better suited for unstructured grids, e.g., graph neural networks, they are usually more computationally heavy and more difficult to implement. Given the limited scope of our proof-of-concept, we have thus decided to make use of “normal” convolutional neural networks. Additionally, the variables are different at different positions on the triangles. Consequently, we interpolate from triangular space to a common Cartesian space where the features are extracted. Combined with projecting the features back into triangular space and linearly combining them therein, this architecture turns out to be very efficient and can act as a baseline approach for further studies.
In our new Section 3, we will explain our reasoning behind the feature extraction in Cartesian spaces more than in the former neural network Section.

RC: Figure 1: Please specify which physical variable that is displayed (damage?). Could the authors find a more convincing ”showcase” example? By visual inspection, it looks to me like there is still substantial errors in the ”hybrid” field, which seems at odds with the claim of an 75 % error reduction. Please quantify the error reduction for the case shown.

AR: There was a technical issue to correctly render the file on Copernicus' side. This should be fixed now, and the missing labels etc. should be there now. The shown damage fields are for a lead time of 60 minutes. Given your feedback, we have decided to change the snapshot of the sea-ice damage to a more representative case (Fig. 2). There, the improvement is 62 %, and the hybrid model is able to correctly represent the damaging processes such that not too much damage is produced as in the low-resolution forecast without correction.

RC: l. 35f. (and elsewhere): I am not sure what "wave-like" and "channel-like" - please be more precise.

AR: In the revised manuscript, we will be more specific and will avoid 'wave-like' and 'channel-like'.

RC: Line 76 & 89: a 10 minute (or even 1 hour) forecast is extremely short both for main-stream earth system models and real-world applications. Can you please comment on that and justify looking at these very short time scales?

AR: On the one hand, we want to correct the model error after each integration step, which would be in our case 16 seconds. On the other hand, the neural network is not perfect, and the more signal during training, the better. Furthermore, the neural network is trained without taking interactions with the sea-ice model into account. The missing interactions lead to a distribution shift during the application of the model error correction. Consequently, using a correction time of 10 minutes is already a compromise. Given the limited scope of this proof-of-concept and an already visible distribution shift (Table 8), we have restricted the forecast time to one hour.

RC: The introduction in ll. 80-92 already gives too much technical detail about the methods. This belongs elsewhere.

AR: We will reduce the amount of technical information in the introduction.

**RC: In Figure 2 and the corresponding text, the authors need to help the reader to get a physical understanding of the situation that causes the ice to fracture. Please add arrows indicating the wind field, and refer to Equation (1). Please specify which direction is x and which is y.
Also Figure 2: Please use other colours than black and red to indicate the two grids, otherwise it is difficult to see for color-blind people.**

AR: We will add the forcing field with arrows for this specific case in Figure 2, as shown in Fig. 3. Additionally we will change the colour of the coarse grid in Figure 2 to a lighter blue tone, which should make the figure easier to read.

RC: Line 134: I do not know what "wave-like" means. Please be more precise, and provide the equation with the wind forcing at the earliest possible place in the text.

AR: We will introduce the equation for the wind forcing in the new shortened description of the regional sea-ice model and avoid the term ‘wave-like’.

RC: Figure 6: I much appreciate the sensitivity testing in Section 5.2, very good! However, I am puzzled by the very weak cross-variable coupling in the permutation feature importance. It seems contradictory to your claim that the NN has "learned the dynamics" of the physical model. For instance, for damage as an output variable, it seems that the NN only extracts information from the damage itself, all other input variables are unimportant! Could you please provide some more explanation/clarification/analysis on this?

AR: The input variables are naturally correlated to each other, e.g., for the forcing and the velocities or the area and thickness. Therefore, by destroying the information of a single variable, almost the same information is still available to the neural network by another variable. With compound effects, the picture changes (Tab. 1), e.g., the dynamics of all stress components have combined a large impact on the sea-ice area and thickness, or the absolute values of the dynamical variables combined strongly influence the stress components. However, our result shows in fact that the dynamical behaviour of a variable is the single most important predictor, if no compound effects are taken into account. We will add the Table 1 to the additional results in the Appendix, showing what happens if the information of a complete group is destroyed, which includes compound effects.

RC: Figure 7: It is striking that the low-resolution model is much worse than simple persistence. This makes me wonder whether the NN is just correcting biases (see general comment #2). Please provide some discussion on this.

AR: Dynamical processes of below 8 km are in the subgrid-scale of the low-resolution model setup and parametrised with the damaging process. These processes are nevertheless included in the truth fields on which basis the low-resolution forecasts are initialised. The mismatch between resolved and parameterised processes results into a strong drift for the dynamical variables (velocities, stresses, damage) within the first minutes of forecast. This drift is not a simple bias but a dynamical process, because otherwise the bias-corrected forecast would perform better. Contrary to the low-resolution model, the persistence forecast has no drift, and for the dynamical variables a better score for the shown lead time of up to one hour. We will add some sentences that explain the absence of connection to a bias and will refer for this to the Appendix with the results for additional architectures.

RC: Lines 516 - 519: This is a good start, but a much more in-depth discussion is needed here of the implications and wider applicability of the work presented (see general comment #1).

AR: As written for comment #1, we will make the proof-of-concept character of the study stronger in the introduction, and we will add some discussion about steps towards the generalization.

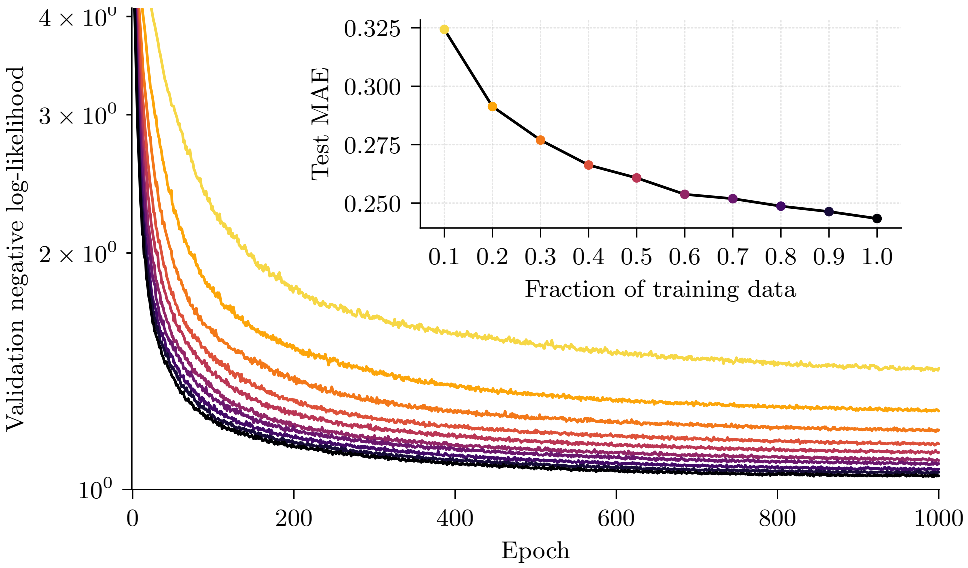


Figure 1: The negative log-likelihood for a Laplace approximation, proportional to the mean absolute error (MAE), with a fixed weighting in the validation dataset as function of epochs for different fractions of training data, the brighter the color, the less training data is used. The smaller Figure shows the averaged MAE in the test dataset as function of the fraction of training data. The performance of the neural networks is averaged across ten different random seeds.

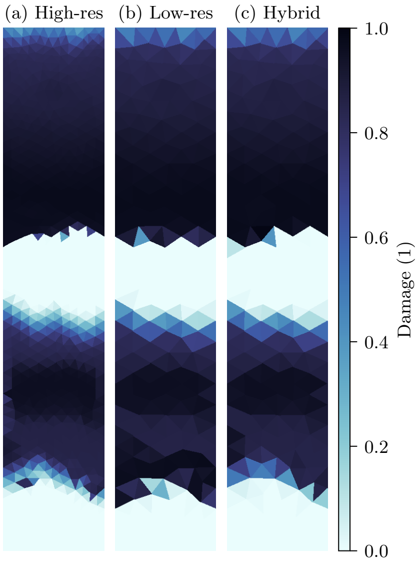


Figure 2: Snapshot of sea-ice damage after a one-hour forecast with the here-used sea-ice dynamics only-model. Shown are the high-resolution truth (a, 4 km resolution) and low-resolution forecasts (b, c). To initialise the low-resolution forecasts, the initial conditions of the high-resolution are projected into a low-resolution space with 8 km resolution. Started from these projected initial conditions, the low-resolution forecast (b) generates too much damage compared to the high-resolution field. Running the low-resolution model together with the learned model error correction (c) leads to a better representation of the damaging process, which improves the forecast by 62 % in this example.

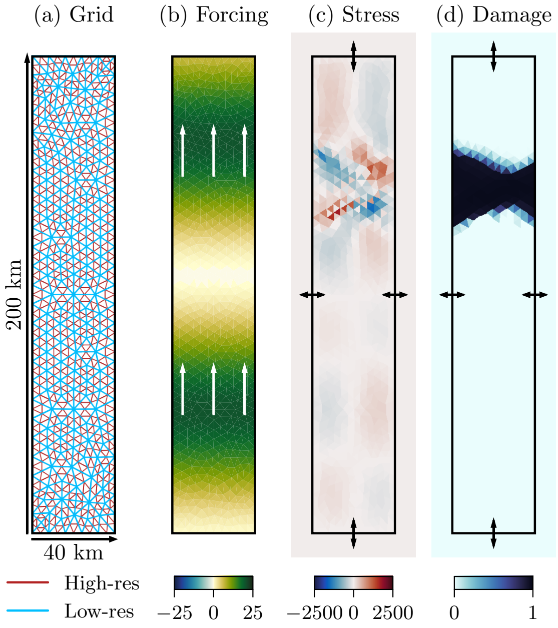
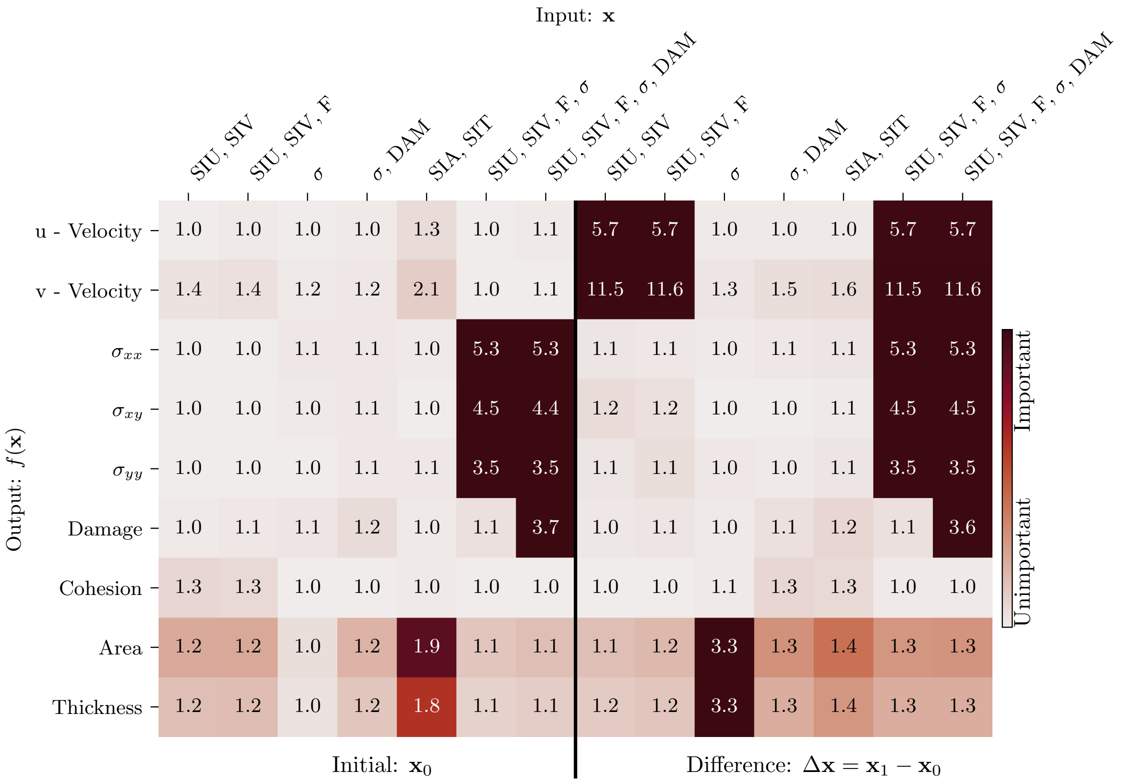


Figure 3: (a) The model domain with the high- (red) and low-resolution (blue) grid; (b) snapshot of the forcing wind velocity in y-direction in $\mathrm{m\,s^{-1}}$, the arrows indicate the movement direction; (c) snapshot of the stress, σ_{xy} in Pa, where the arrows correspond to von Neumann boundary conditions on all four sides; (d) snapshot of the damage, where the arrows correspond to an inflow of undamaged sea ice on all four sides. The three snapshots are taken at an arbitrary time and represent a commonly encountered case in our dataset.



Input: $\mathbf{x}$
Output: $f(\mathbf{x})$
Important
Unimportant

	Initial: $\mathbf{x}_0$							Difference: $\Delta\mathbf{x} = \mathbf{x}_1 - \mathbf{x}_0$						
	SIU, SIV	SIU, SIV, F	σ	σ, DAM	SIA, SIT	SIU, SIV, F, σ	SIU, SIV, F, σ, DAM	SIU, SIV	SIU, SIV, F	σ	σ, DAM	SIA, SIT	SIU, SIV, F, σ	SIU, SIV, F, σ, DAM
u - Velocity	1.0	1.0	1.0	1.0	1.3	1.0	1.1	5.7	5.7	1.0	1.0	1.0	5.7	5.7
v - Velocity	1.4	1.4	1.2	1.2	2.1	1.0	1.1	11.5	11.6	1.3	1.5	1.6	11.5	11.6
σ_{xx}	1.0	1.0	1.1	1.1	1.0	5.3	5.3	1.1	1.1	1.0	1.1	1.1	5.3	5.3
σ_{xy}	1.0	1.0	1.0	1.1	1.0	4.5	4.4	1.2	1.2	1.0	1.0	1.1	4.5	4.5
σ_{yy}	1.0	1.0	1.0	1.1	1.1	3.5	3.5	1.1	1.1	1.0	1.0	1.1	3.5	3.5
Damage	1.0	1.1	1.1	1.2	1.0	1.1	3.7	1.0	1.1	1.0	1.1	1.2	1.1	3.6
Cohesion	1.3	1.3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.1	1.3	1.3	1.0	1.0
Area	1.2	1.2	1.0	1.2	1.9	1.1	1.1	1.1	1.2	3.3	1.3	1.4	1.3	1.3
Thickness	1.2	1.2	1.0	1.2	1.8	1.1	1.1	1.2	1.2	3.3	1.3	1.4	1.3	1.3

Table 1: Permutation feature importance of different variable groups. The colouring is the same as in Table 6 of the original manuscript. *SIU* stands for velocity in x-direction, *SIV* for velocity in y-direction, F for wind forcing, σ for all stress variables, *DAM* for damage, *SIA* for sea-ice area, and *SIT* for sea-ice thickness.