Reviewer 2

Excellent paper. Great improvement from previous studies where the ML approach learns only the misfits whilst the authors have implemented an on-the-fly system.

Response: We thank the Referee for their support of our work.

Figure 3 and 4 need to be together for a better comparison.

Response: We have made this change.

Architectures of networks used?

Response: We have added extensive explanation related to the LSTM architecture used in the revision.

Was layer normalisation used? Why not?

Response: Layer normalisation was not explored in this work. We note that the main focus of our study was to show that an inaccurate surrogate can be improved by observations during deployment in contrast with explicitly finding the best possible surrogate. We believe several methodological tricks can be used to potentially improve our LSTM implementation.

What other temporal NN could be used? Transformers?

Response: Methods that are in the family of recurrent neural networks would be appropriate for this forecasting. Some examples are transformers, neural ordinary differential equations, gated recurrent units. We note that while several different surrogate modeling strategies are viable for the forward model, the purpose of this article was to demonstrate data assimilation with a fixed model with improved results.

What is the effort of using a CNN instead of a PCA reduction to feed the LSTM AE?

Response: We decided to utilise PCA instead of CNN for maximising the interpretability of the latent space although this would lead to reduced compression efficiency. Several recent articles have looked at autoencoders for dimensionality reduction and our data-assimilation strategy is equally applicable for that scenario. We have added a statement to clarify this in the main article.

I would like to see a brief discussion of how an adversarial approach might help the rollout of the forecast here.

Response: We thank the reviewer for this suggestion. In this work, we have not explored adversarial training of the LSTM, but suspect it would be helpful in making the forecasts more robust to noisy inputs. A prominent failure mode of RNN architectures is the growth of errors due to the autoregressive nature of forecasting (small errors compounding to grow uncontrollably) and this approach may assist with that. We shall be exploring this for our future studies.