Response to Referee #1
for "Elucidating the performance of data assimilation neural networks for chaotic dynamics"

Egusphere-2026-245

Marc Bocquet, Tobias S. Finn, Sibo Cheng, Alban Farchi

12 June 2026

This paper tries to elucidate the reasons of the impressive results obtained by Data Assimilation Networks (DANs) in Bocquet et al. 2024 (Boc24 in the manuscript), by leveraging explainability techniques relying on the sensitivity of their Jacobian matrices. In particular, they provide a satisfactory proof that the scalability of the method to "unseen" higher dimensional model versions is due to its focus on local patterns.

The manuscript is well-written and interesting to read, and achieve the stated goal, providing a much needed explainability framework, a feature usually absent from most works using machine learning (ML). I recommend therefore its publication once the following comments have been addressed.

We sincerely thank the Reviewer for the positive evaluation of the manuscript and for the constructive comments. In the following, we discuss the raised concerns and what we have changed or revised in the manuscript. The revised manuscript and a pdf document highlighting the differences between the original and the revised manuscript will be provided.

(1) My main feeling is that the article go a bridge too far when it comes to discarding the usefulness of ensembles. For example lines 30-31: 'This result challenges the long-standing assumption that explicit ensemble representations are indispensable to estimate flow-dependent uncertainties in chaotic systems.' This may be true for Data Assimilation (DA), but it is not sure that this holds with respect to other analysis, where ensemble representations (or probabilities) might still present some usefulness.

We agree with this important qualification. Our intention was not to argue that ensembles are generally unnecessary, nor that probabilistic representations can be dispensed with in forecasting, uncertainty quantification, etc. Our result is narrower: in the perfect-model filtering experiments considered here, the point-estimation accuracy of a learned analysis operator can match ensemble-based baselines even when the operator is driven by a single forecast state. Actually, in parallel to using DAN, we keep using ensemble DA methods for DA studies, either as main DA methods or as baseline against ML-enhanced methods. Note however that the sentence that you quote does not imply that the ensembles have become useless, but that alternatives to ensembles for UQ may exist; and it is true that we have only shown this in the context of DA.

Added subsection in the main discussion section:

Ensembles, uncertainty quantification, and multiplicative ergodic theorem

Our results show that, in the perfect-model filtering setting considered here, an explicit forecast ensemble is not strictly required for a learned analysis operator to extract the flow-dependent information needed to achieve EnKF-like point-estimation accuracy. This conclusion should not be interpreted as a dismissal of ensemble representations in general. Ensembles remain essential for probabilistic forecasting, uncertainty quantification, the diagnosis of model and observation errors, smoothing, risk-sensitive applications, and regimes in which the posterior distribution is strongly non-Gaussian or multi-modal.

Moreover, the deterministic analysis operator $\mathbf{a}_{\boldsymbol{\theta}}$ could itself be used as a building block for ensemble generation. For instance, applying $\mathbf{a}_{\boldsymbol{\theta}}$ to perturbed innovations would produce a set of analysis states, in a way that is consistent with the local, innovation-dependent interpretation used in, e.g., Appendix I to obtain the linear regression approximation of $\mathbf{a}_{\boldsymbol{\theta}}$ with respect to $\boldsymbol{\zeta}$. Such an

analysis ensemble could then be propagated by the forecast model to estimate forecast uncertainty. We have not pursued this probabilistic use of DAN in the present study, since our focus was on point-estimation accuracy, but it represents a natural direction for future work.

(2) The authors should also comment on the fact that besides DA, the determination of flow-dependent uncertainties using ML has already been studied, with variable success. This raises the question of why it works so well here. One could conjecture that is due to the information on uncertainties (and instabilities) needed by the DA processes are actually suitable for its inference using ML, while determining actual precises quantities such as the Covariant Lyapunov Vectors (CLVs) and Lyapunov exponents is a more challenging task. The fact that a Multiplicative Ergodic Theorem (MET) exists for the underlying systems is clear, but this provides a mapping between the states of a system and its CLVs, it doesn't mean that this mapping between the CLVs at a given time can be determined alone from the state at the same time. For example, the Ginelli algorithm combines a forward and a backward pass, which take some "time" to converge (see F. Noethen studies to have an idea on this). Therefore it may explain why ML sometimes struggle to learn this mapping. Here, for DA, it works very well, and the DAN seems to be able to learn what is useful, even with just one member, but somehow this is an easier task than learning the MET mapping. In the end, it is probably connected to the fact that CLVs are non-local (and therefore dimensional scalability of algorithms computing them is not clear), contrary to the mapping between the forecast state and the analysis error covariance, as shown by your work.

We believe that our manuscript did not go in any way against your comment. We take your comment as an invitation to discuss this very interesting point.

By *The fact that a MET exists for the underlying systems is clear, but this provides a mapping between the states of a system and its CLVs, it doesn't mean that this mapping between the CLVs at a given time can be determined alone from the state at the same time.*, we guess that what you have in mind is: *MET ensures that, for almost every state on the attractor, the Oseledets subspaces/CLVs are state-dependent measurable objects. However, MET alone does not imply that this state-to-CLV map is regular, local, or practically recoverable from instantaneous state samples without information about the tangent dynamics along trajectories.*

In that sense, we agree with you that the existence of the Oseledets splitting does not by itself establish that the map from state to CLVs is smooth, continuous, or easily learnable from finite data. However, for the deterministic systems considered here, MET does imply that the CLV/Oseledets subspaces are measurable functions of the full state, for almost every state on the attractor. We have revised the text to make clear that our claim concerns this almost-everywhere state dependence, not the stronger assertion that the CLVs can be obtained by a simple local diagnostic from the instantaneous state alone.

One should also keep in mind that the DA process as a dynamical system is asymptotically stable as opposed to the forecast dynamics it is based upon. That may be critical in explaining why learning such mapping is easier, as opposed to, e.g., learning a mapping from the state to the CLVs.

Last part of the added subsection the main discussion section: At a more theoretical level, the success of DAN may be facilitated by the existence, in the present setting, of a sufficiently regular and local dependence of the relevant flow-dependent analysis correction on the forecast state. This should not be taken for granted in general. For example, the multiplicative ergodic theorem ensures that covariant Lyapunov subspaces are defined as measurable functions of the state, for almost every state on the attractor, but it does not guarantee that this state-to-Oseledets-splitting map is continuous, local, or easily learnable from finite data. Other applications beyond DA, such as learning covariant Lyapunov vectors directly from the state, may therefore involve maps with poorer regularity or locality, making them substantially more difficult to learn and possibly less scalable. This question is left for future work.

(3) Lines 67-68: The analysis is an estimator of the conditional probability density function, or an estimator of its first moment ? Please clarify.

Thank you for pointing out this imprecision. The analysis state is a point estimator of the hidden state, obtained from the conditional distribution. Depending on the filtering formulation

and loss, it can be interpreted as a posterior mean, a MAP estimate, or another summary of the conditional distribution. In the present deterministic least-squares training, it is best described as a point estimator approximating the conditional mean under the adopted loss. We have revised the sentence accordingly.

Changed text: A filtering DA scheme estimates the hidden state $\mathbf{x}_k^{\mathrm{t}}$ at t_k from the observations available up to that time. The analysis $\mathbf{x}_k^{\mathrm{a}}$ is a point estimator, such as the posterior mean or maximum a posteriori estimate, of the conditional probability density function $p(\mathbf{x}_k^{\mathrm{t}}|\mathbf{y}_k, \mathbf{y}_{k-1}, \ldots, \mathbf{y}_1)$. A *sequential* filtering DA scheme infers $\mathbf{x}_k^{\mathrm{a}}$ from $\mathbf{y}_k$ and from background information about the state at t_{k-1} (possibly brought forward to t_k using $\mathcal{M}$).

(4) Does the argument on translational invariance holds also because CNNs are know to be shift invariant, meaning the method used here would not be applicable to DNN without this invariance ?

To be precise, convolutional neural networks are translation equivariant, not strictly translation invariant, until an invariant reduction such as time-averaging is applied. The translational symmetry is first a property of the L96 model and of the homogeneous observation configuration; the convolutional architecture then encodes this equivariance by weight sharing and makes the learning problem much more data-efficient. A fully connected DNN could, in principle, learn the same map if supplied with enough data and suitable augmentation, but it would not encode the symmetry, would have many more parameters, and would not naturally transfer to a different value of N_{x}. Thus the method is not mathematically inapplicable without a CNN, but the demonstrated scalability relies on using an architecture compatible with the symmetry and locality of the problem.

(5) Basically the authors show that the CNNs DANs can be generalized to "unseen" cases (i.e. unseen higher dimensional version of the model at hand), because actually they focus on a subset of features (i.e. the local features) common to most of the model versions. Does that means that in the case of models encountering a dramatic change in its local properties (such as a change in the logarithmic slope of the energy spectrum for example) when the dimensionality is increased, DAN generalization would not work ?

Yes, this is an important limitation. The generalisation is expected to work only when the local dynamics and local forecast-error structures seen during training remain representative of those encountered at the new resolution or model size. If increasing the dimension changes the local physics, the effective energy spectrum, the balance relations, or the observation-induced error structures, then a DAN trained at the original dimension may fail and would likely require retraining, conditioning on the new regime, or a multiscale architecture. We had tested this on a (continuous) Kuramoto-Sivashinsky model with two distinct spectral resolutions over the same domain. The DAN trained on one resolution would, out of the box, fail when tested on the other resolution, as expected. We have added this caveat to the manuscript.

Changed text: This neat scalability has limitations. If increasing the dimension changed the local statistics or local dynamical balances of the model, for example through a substantial change of spectral slope or local instability mechanisms, a DAN trained at the smaller dimension should not be expected to generalise without fine-tuning, additional training, or architectural adaptation.

(6) The manuscript is not self-contained enough, lots of details are simply mentioned as being from Boc24 and the reader has to go there to understand DANs details. Therefore, if in the future it becomes (more) difficult to find Boc24 for any reason, this manuscript would cease to be understandable. Could the authors incorporate a reasonable amount of the Boc24 setup description in this paper as well? Like for example a figure of the CNNs DAN schematics.

We agree and we have created a new appendix (now Appendix A), referred from the methodological section, to provide the details on the architecture, and we have supplemented the appendix on the sensitivity to the training parameters (now Appendix B) with a figure describing how the

data are organised and fed to the training scheme.

(7) There is a typo in the x label of fig 7.

The transpose does appear when using the original pdf and using any pdf viewer we have access to. Figure 6 is also impacted (background grid). We have checked that all the required fonts are embedded in both the standalone figure and in the original manuscript pdf. Hence, the missing transpose seems to be due to Copernicus' processing of the figures.

(8) Line 131: 'memorises' is maybe a bit too much anthropomorphic here.

As much as we would like to agree with you, *to memorise* is also commonly used for machines. And it is probably easier to pinpoint the memory footprint in a CNN than in a human brain, notably in the filters of the convolutional layers. Some neural networks, such as echo state networks are explicitly based on the ability to memorise patterns.