Building a performant hybrid model framework for the IFS

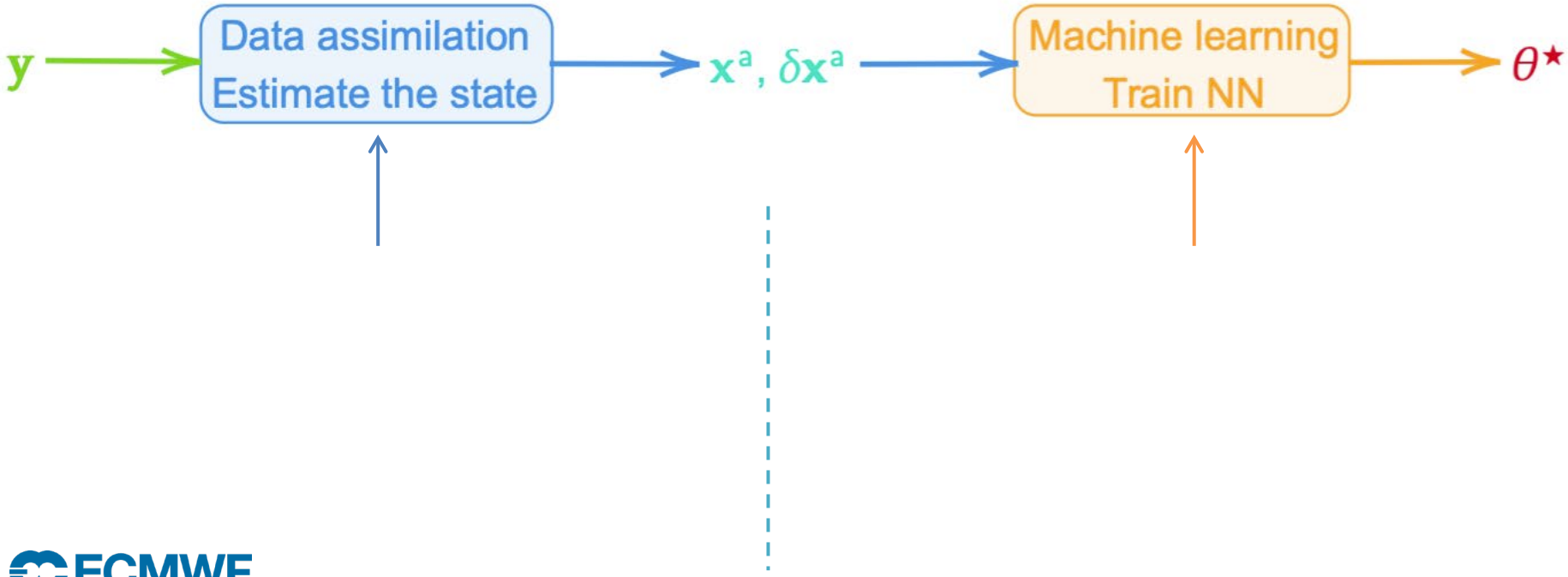
Alban Farchi, Marcin Chrust
Marc Bocquet (ENPC), Massimo Bonavita, Patrick Laloyaux, Wei Pan

Outline

- A brief history of NN 4D-Var
- From sequential to batched learning
- Ongoing and future developments

The early days of hybrid modelling

- The predictable part of analysis increments can be attributed to model errors
- First version of the hybrid model, a.k.a. offline learning (Farchi et al, 2021a):
  - Train a NN to predict the analysis increments.
  - Include the NN in the forecasting system and use the NN output as a forcing throughout the forecast.



A NN correction for the IFS

- 4 atmospheric variables in the same NN: t, lnsp, vo, d
- Vertical / column architecture: atmospheric columns are processed independently (Bonavita and Laloyaux, 2020, Farchi et al. 2025)
- 137 model levels in both input and output: 412 channels in total

Offline validation at T15

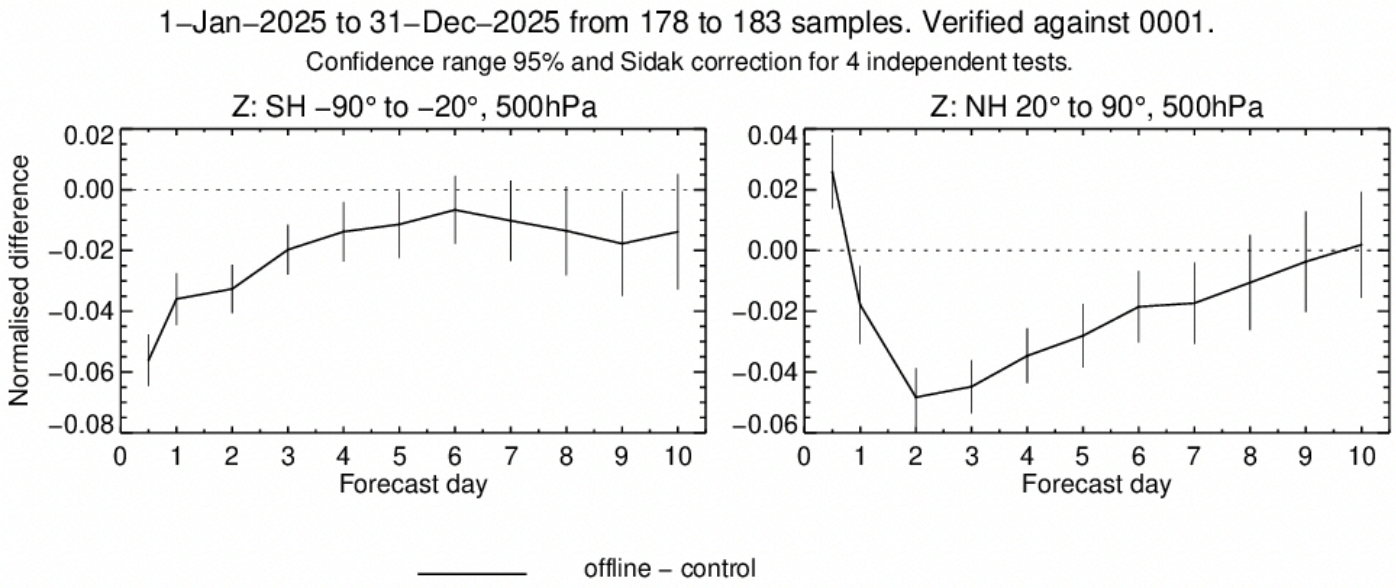
Relative MSE on the validation set

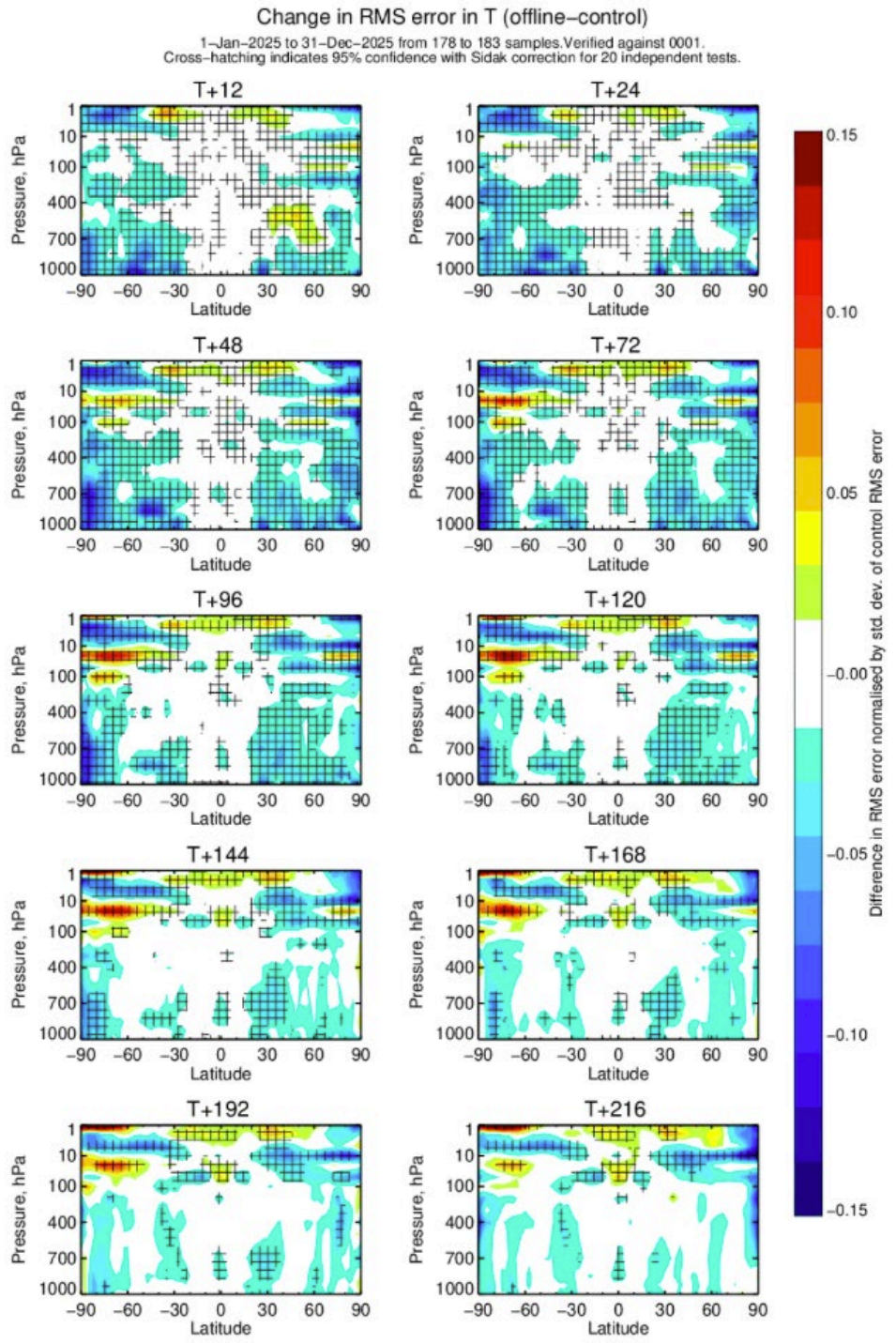
Name	Lnsp	T	Vo	D
No correction	1.000	1.000	1.000	1.000
Mean increment	1.000	0.969	1.000	1.000
Climatological increment	0.812	0.665	0.933	0.943
Trained NN	**0.794**	**0.610**	**0.862**	**0.918**

- NN trained using 3 years of operational increments at large scale (T15)
- Overall, the NN can predict 10 to 30% of the increments
- The increments for t, lnsp, and d are more predictable than for vo
- The predictions are in general most accurate close to the surface

Accuracy of the offline hybrid model

- Forecast-only experiments in 2025 (unseen data) at TCo399
- Correction masked in the tropics

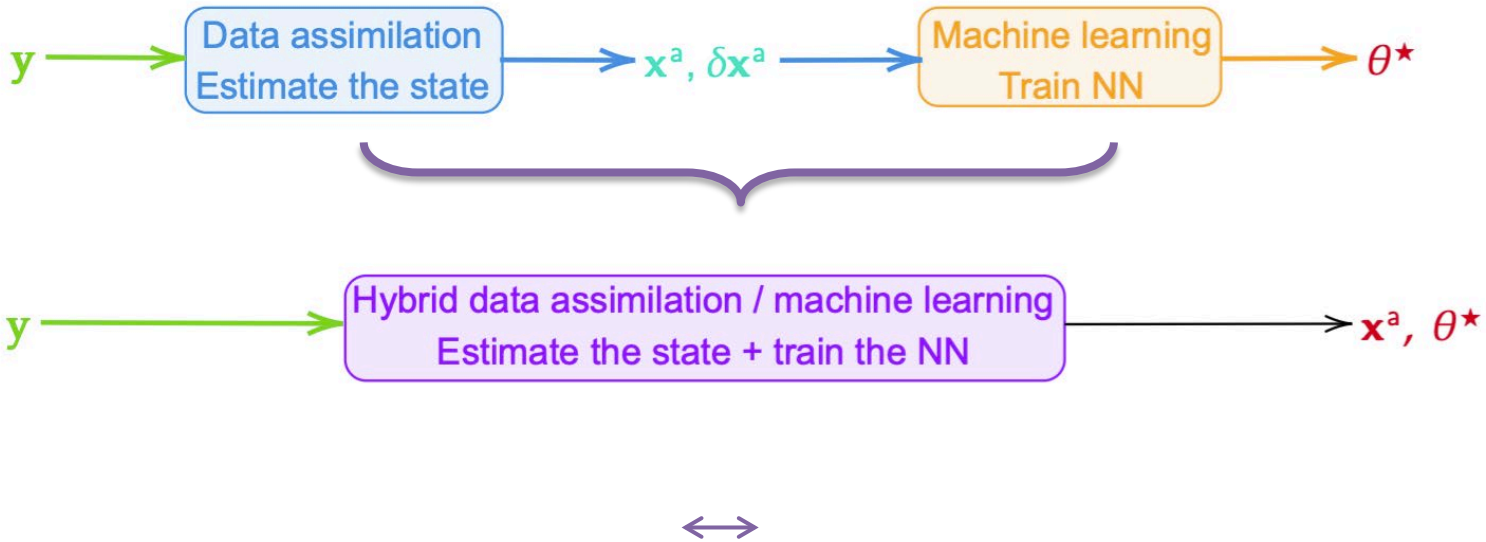




ECMWF

Limitations of offline learning

- Major limitations of offline learning:
  - Analysis increments are taken as proxy for forecast errors that develop over one DAW.
  - Can significantly differ from the instantaneous model errors.
- Move towards online learning (Farchi et al. 2021, 2023):
  - Interactions between the forecast model and the NN can be taken into account

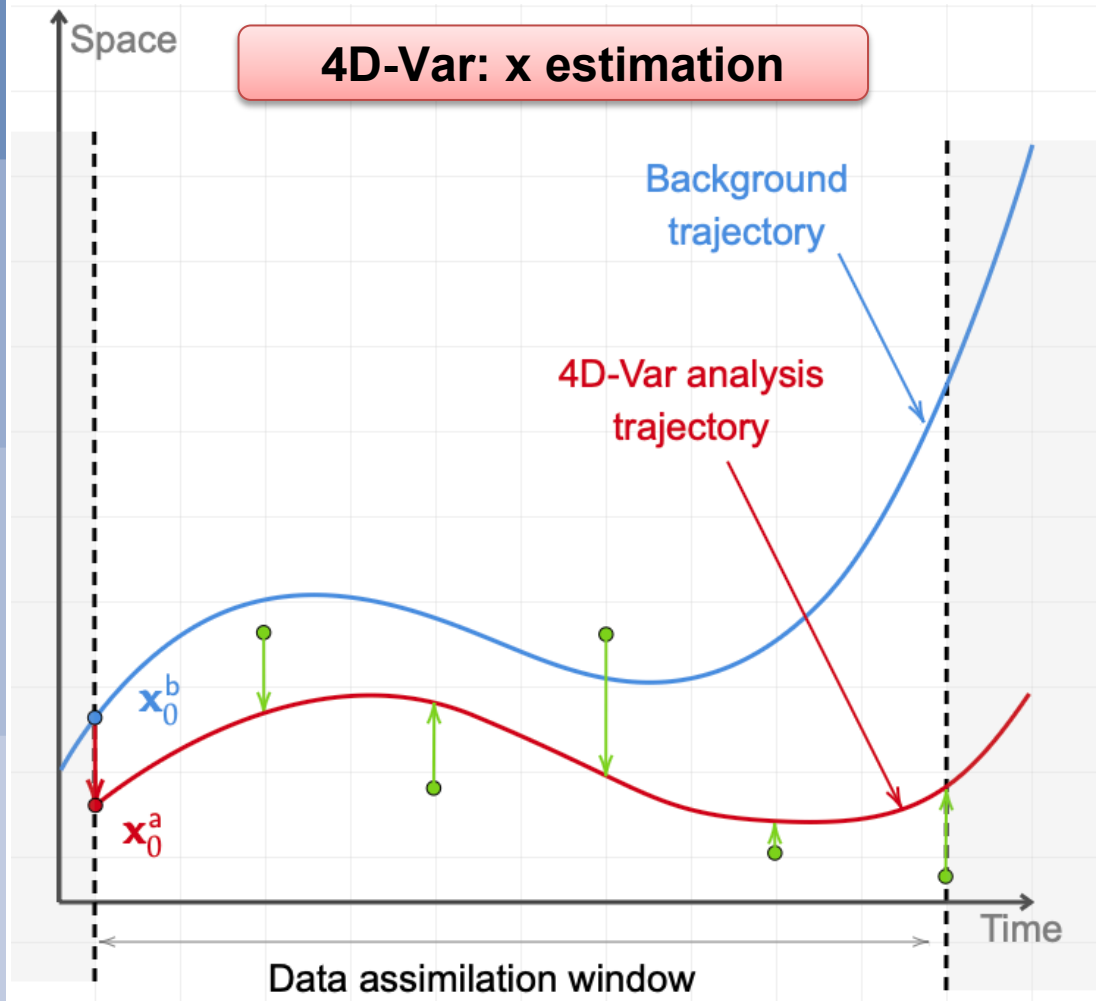


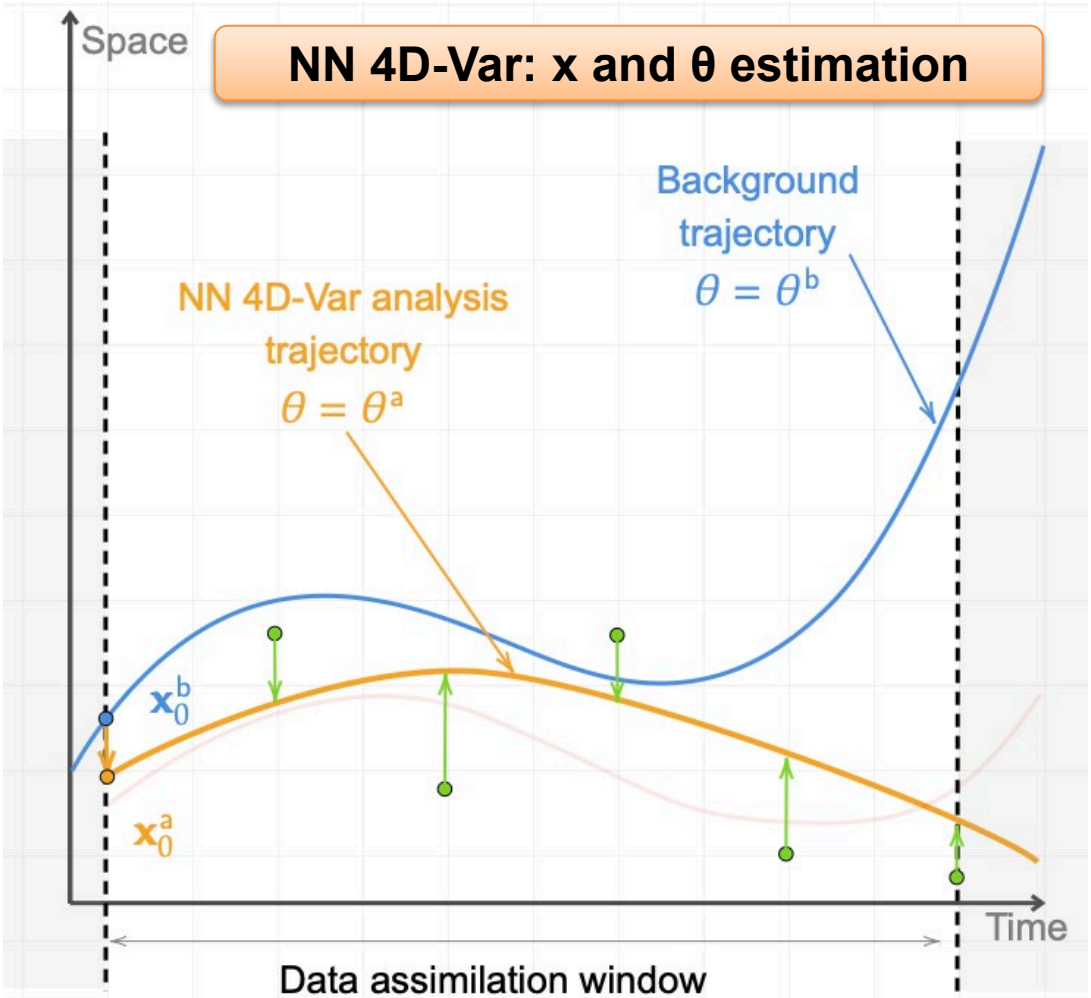
NN 4D-Var – a rigorous formalism to train NN online

- A new variant of weak-constraint 4D-Var:

(Farchi et al, 2021, 2023)

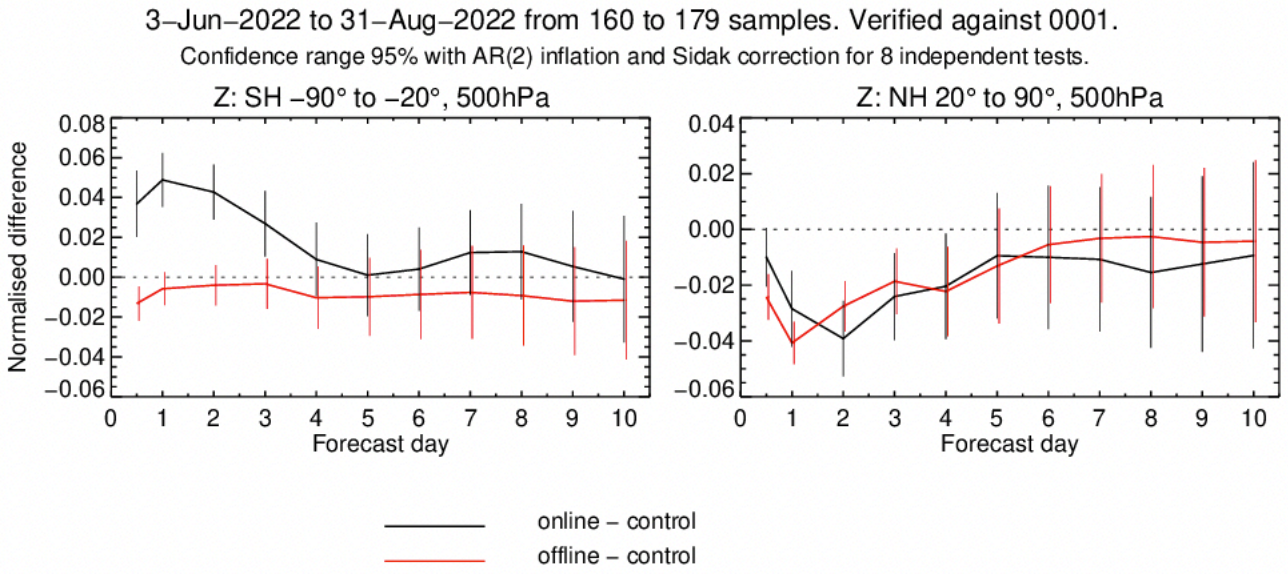
$$\mathcal{J}(\theta,\ \mathbf{x}_0) = \frac{1}{2}\left\|\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right\|^2_{\mathbf{B}^{-1}} + \frac{1}{2}\left\|\theta - \theta^{\mathrm{b}}\right\|^2_{\mathbf{P}^{-1}} + \frac{1}{2}\sum_{k=0}^{L}\left\|\mathbf{y}_k - \mathcal{H}_k \circ \mathcal{M}_{k:0}\left(\theta,\ \mathbf{x}_0\right)\right\|^2_{\mathbf{R}^{-1}}$$



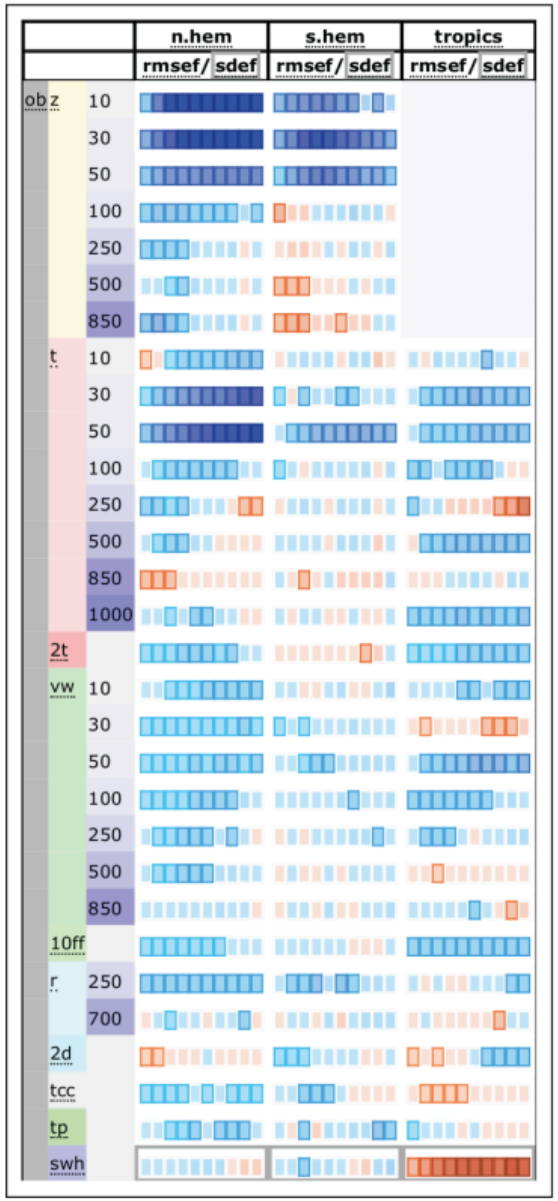


Training online with the standard DA setup (12h window)

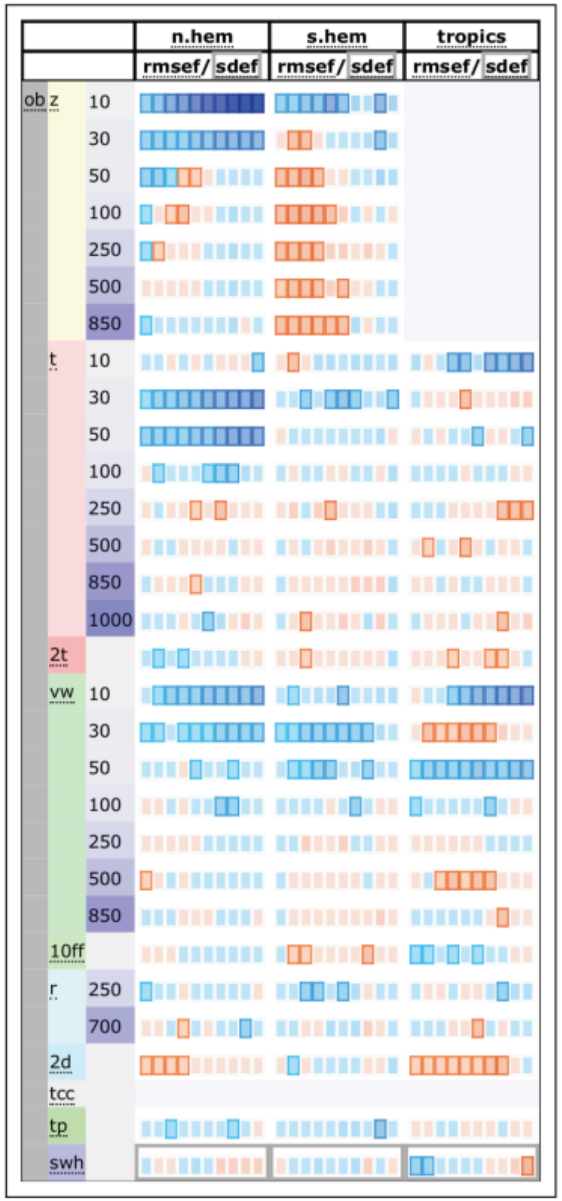
- Online training (fine-tuning) in summer 2022 at TCo399 (from Farchi et al., 2025)
- Largest impact visible in the stratosphere



Online - Reference



Online - Offline

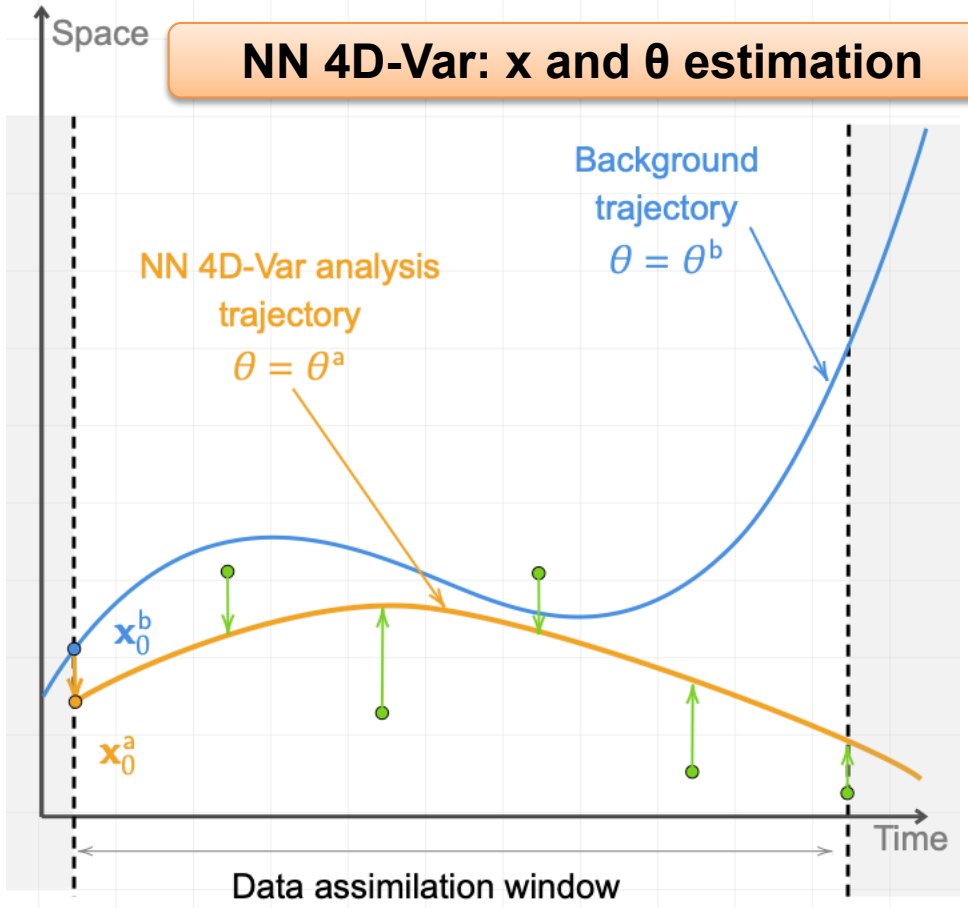


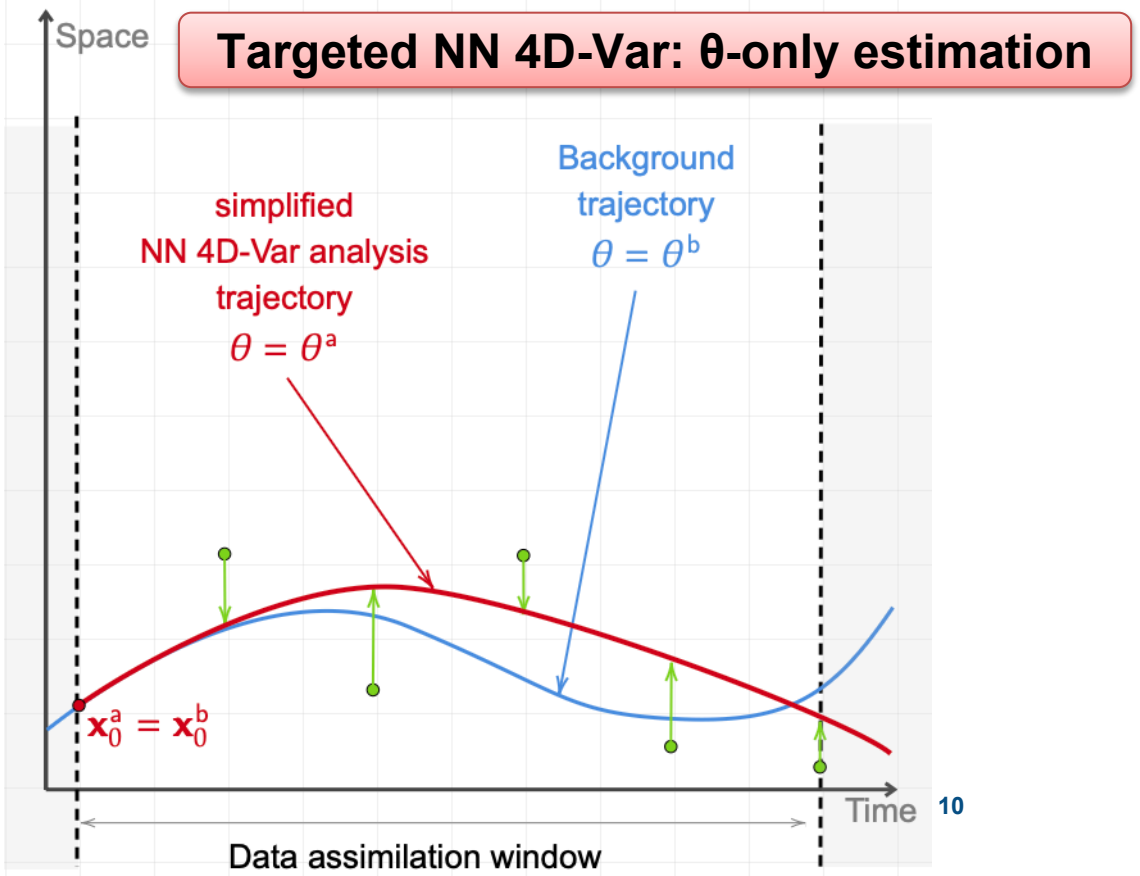
Simplifying NN 4D-Var to focus on the NN parameters

- To further increase the impact of online training, we need 4D-Var to focus on the NN parameters:

$$\mathcal{J}(\theta, \mathbf{x}_0) = \frac{1}{2}\left\|\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right\|^2_{\mathbf{B}^{-1}} + \frac{1}{2}\left\|\theta - \theta^{\mathrm{b}}\right\|^2_{\mathbf{P}^{-1}} + \frac{1}{2}\sum_{k=0}^{L}\left\|\mathbf{y}_k - \mathcal{H}_k \circ \mathcal{M}_{k:0}\left(\theta, \mathbf{x}_0^{\mathrm{pa}}\right)\right\|^2_{\mathbf{R}^{-1}}$$

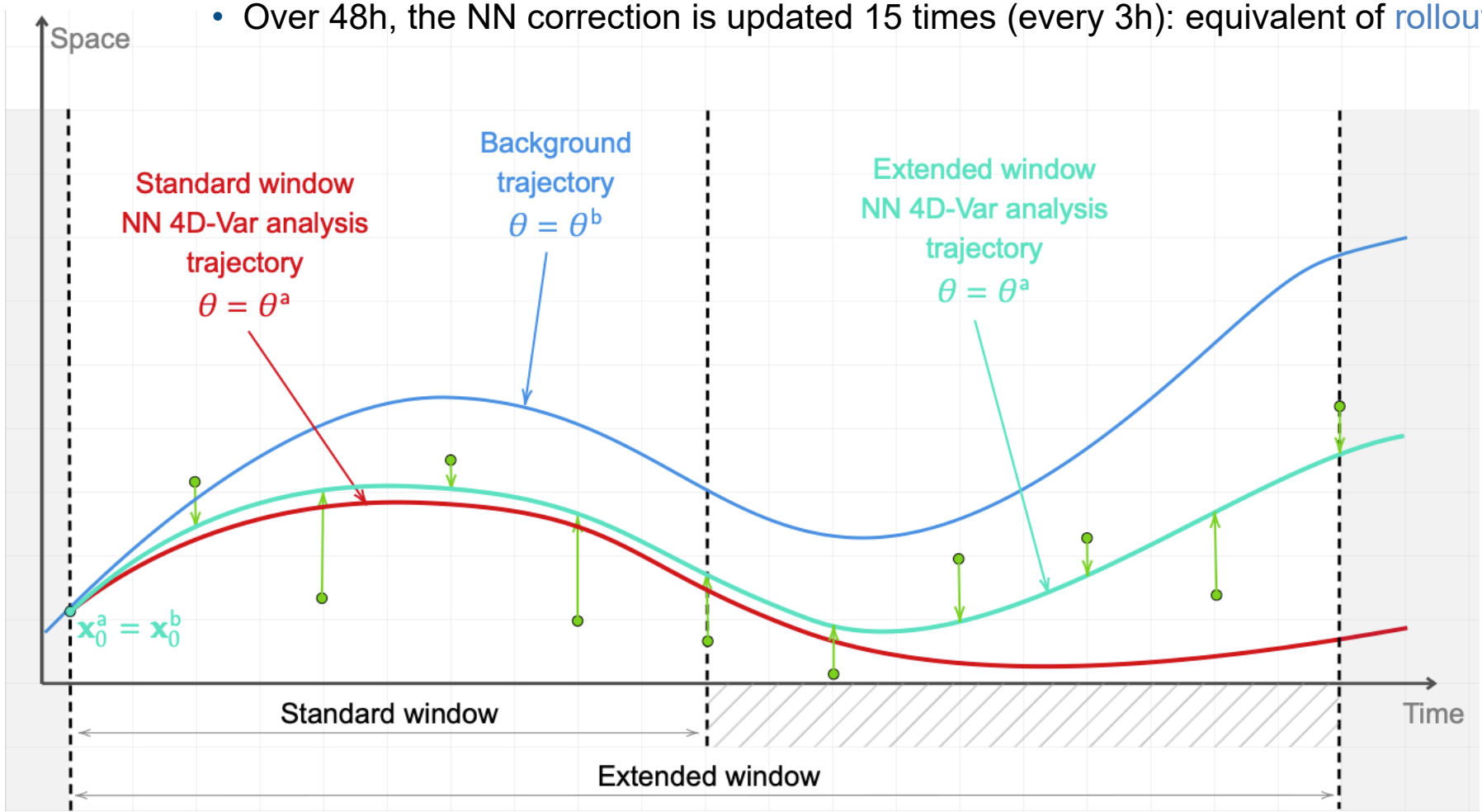
- Where $\mathbf{x}_0^{\mathrm{pa}}$ is a precomputed analysis (e.g. the operational analysis)





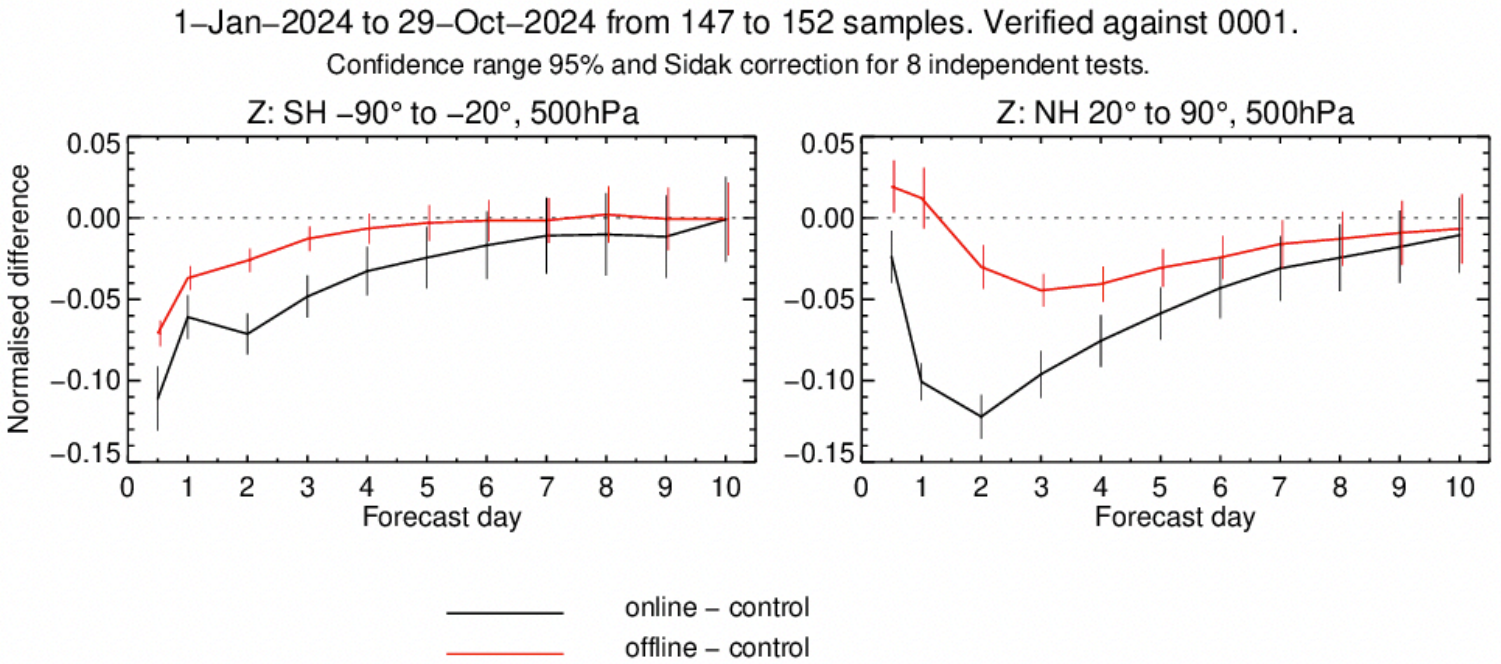
Extending the DA window from 12h to 48h

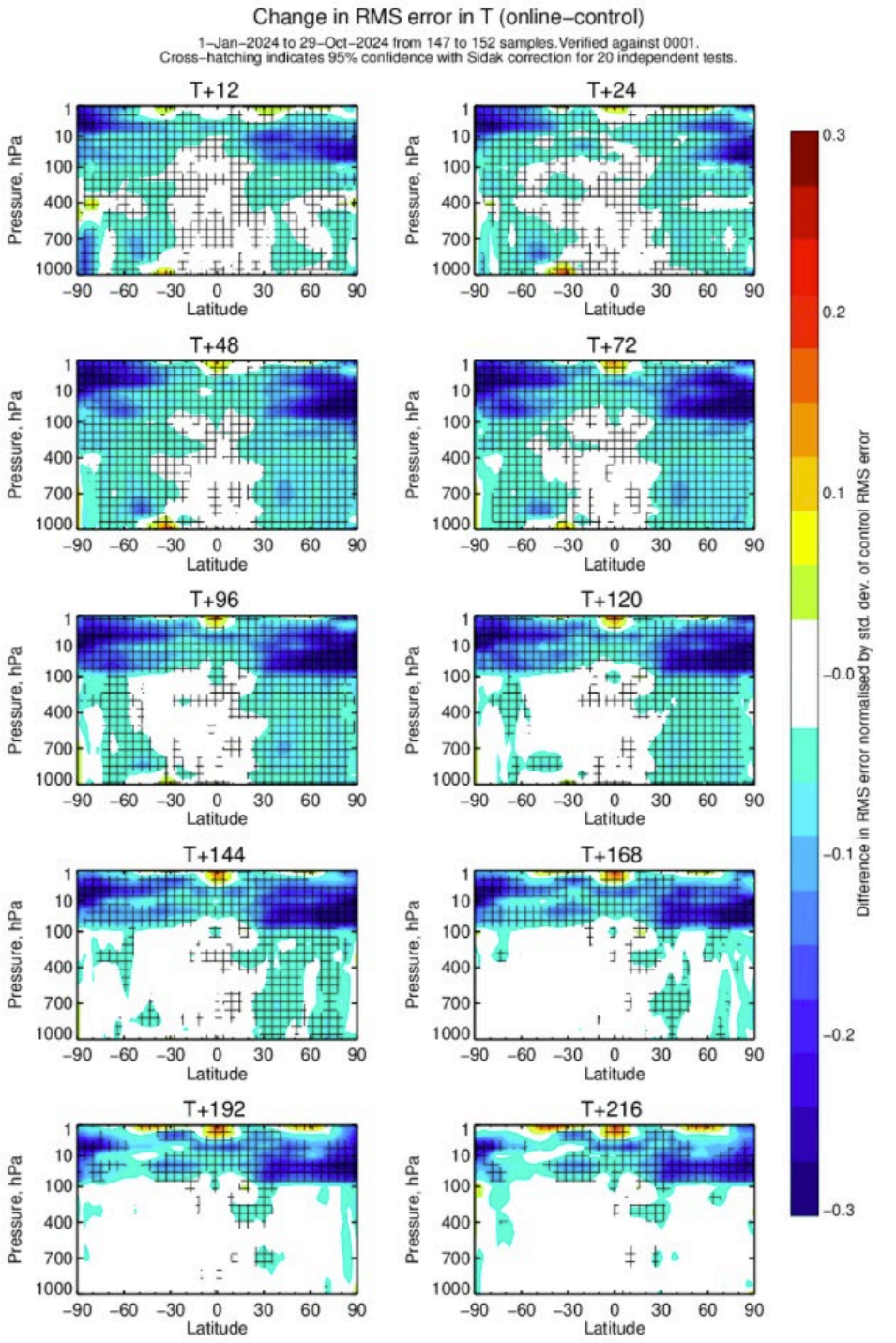
- The length of the DAW is extended from 12h to 48h:
  - More observations are available to constrain the parameters of the NN
  - It helps capture model errors that develop over longer time-scales
- Over 48h, the NN correction is updated 15 times (every 3h): equivalent of rollout for MLWP models



Training online with 48h windows

- Online training (fine-tuning) in 2024 at TCo399
- Much stronger impact than with 12h windows
- Caveats:
  - 48h windows entails 36h of future observations
  - Results are obtained with a sequence of flow-adapted NNs, not one fixed network

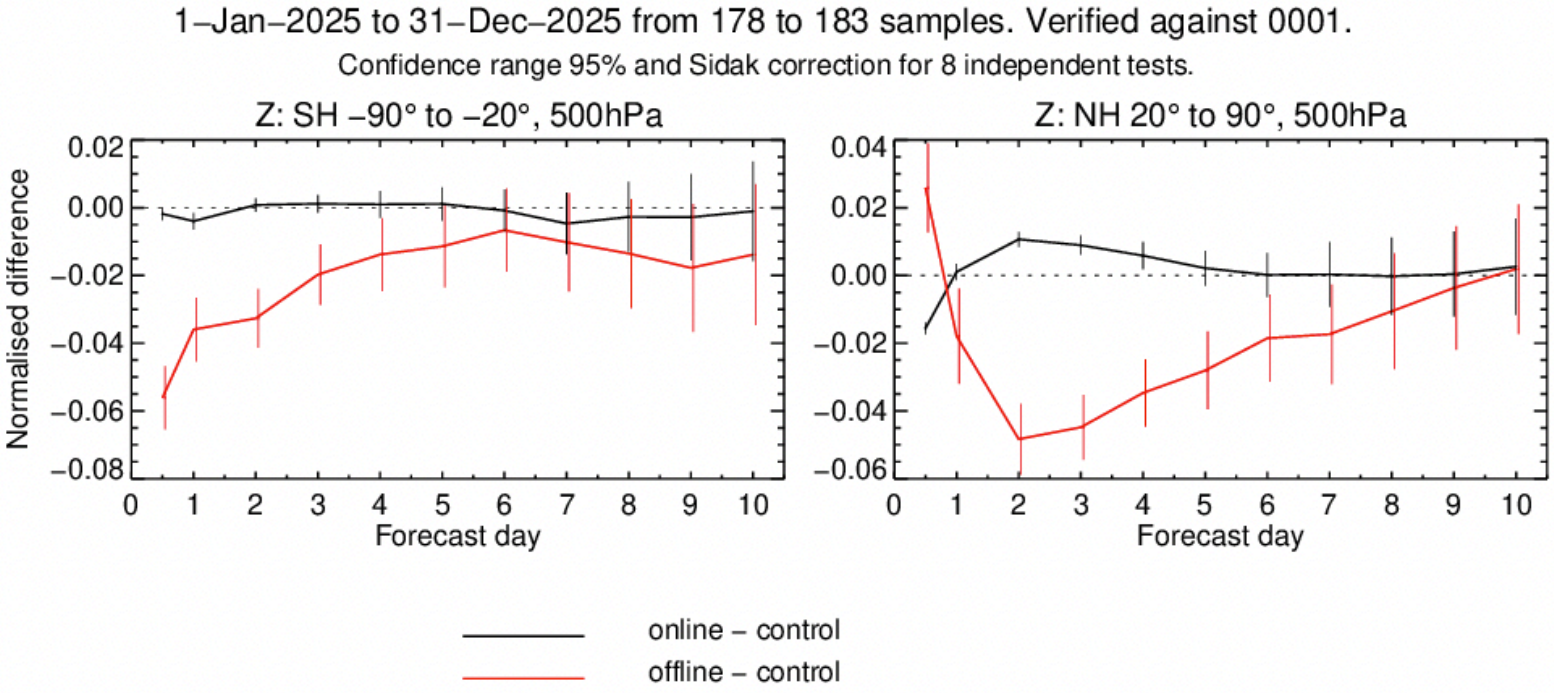


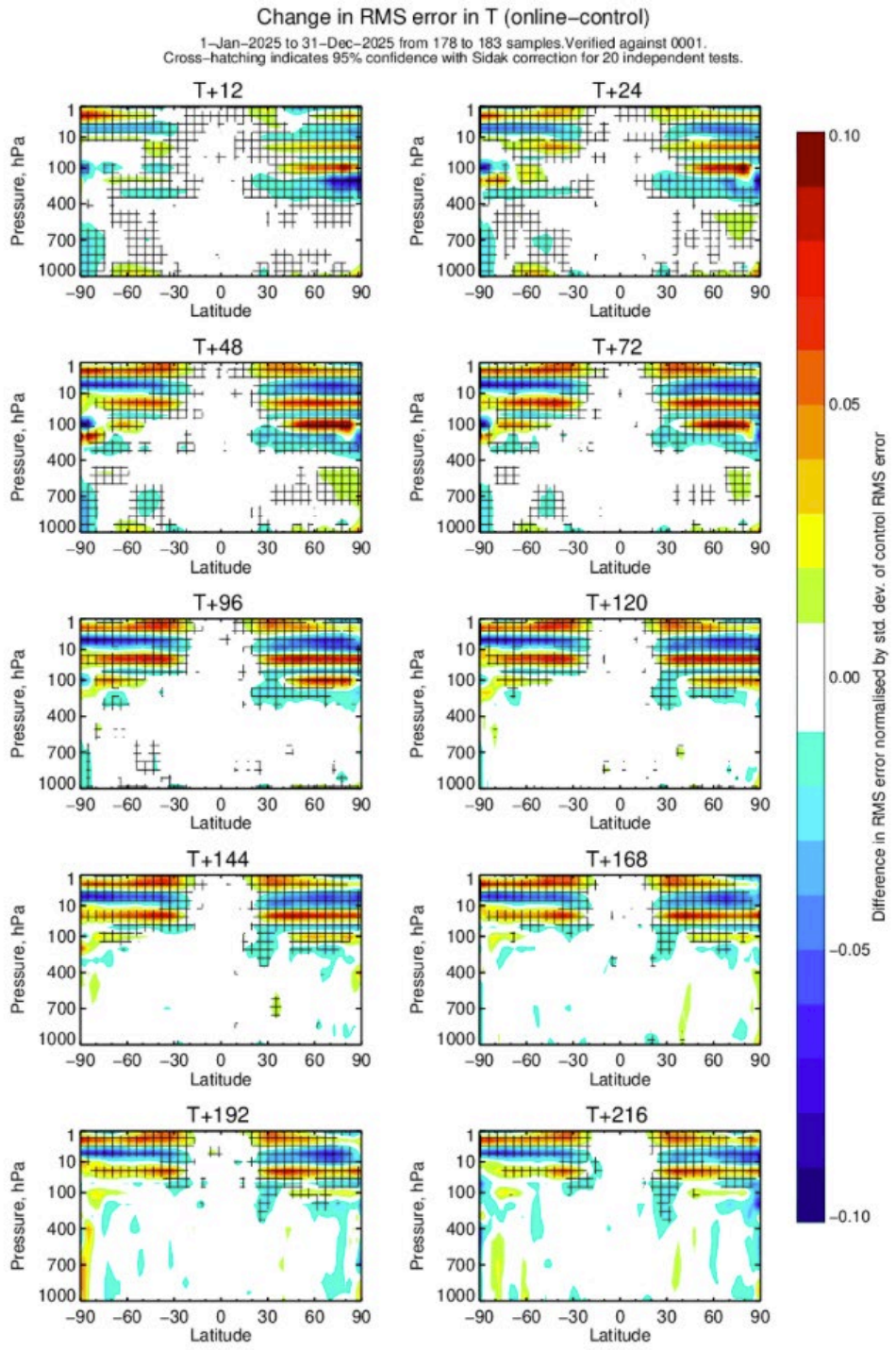


ECMWF

Accuracy of the online hybrid model

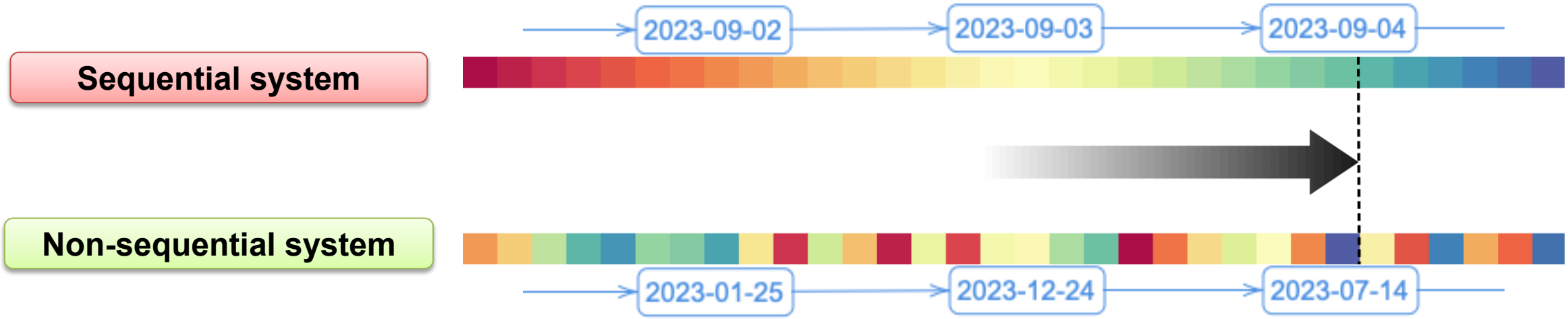
- Forecast-only experiments in 2025 (unseen data) at TCo399
- The online NN correction does not generalise to unseen data





Limitations of a sequential NN 4D-Var system

- NWP models are not autonomous, and model errors have seasonal components
- NN 4D-Var has been tested in a sequential setup
  - NN parameters may only keep in memory information about model errors in the recent past
- We need to relax the sequential assumption of the system



NN 4D-Var – a reimplementation of SGD?

	NN 4D-Var	Targeted NN 4D-Var	SGD
Control Variable	x and θ	Only θ	Only θ
Order of windows / sample	Sequential	Sequential / Random	Random
Parameter updated mechanism	$\theta(t+1) = \theta(t) + \delta\theta^{a}(t)$	$\theta(t+1) = \theta(t) + \delta\theta^{a}(t)$	$\theta(t+1) = \theta(t) - \gamma\nabla_{\theta}\mathcal{L}(\theta(t))$

- There is a clear similarity between one cycle of the targeted NN 4D-Var and one iteration in the SGD algorithm, with

$$\delta\theta^{a}(t) \Leftrightarrow -\gamma\nabla_{\theta}\mathcal{L}(\theta(t))$$

Batching NN 4D-Var

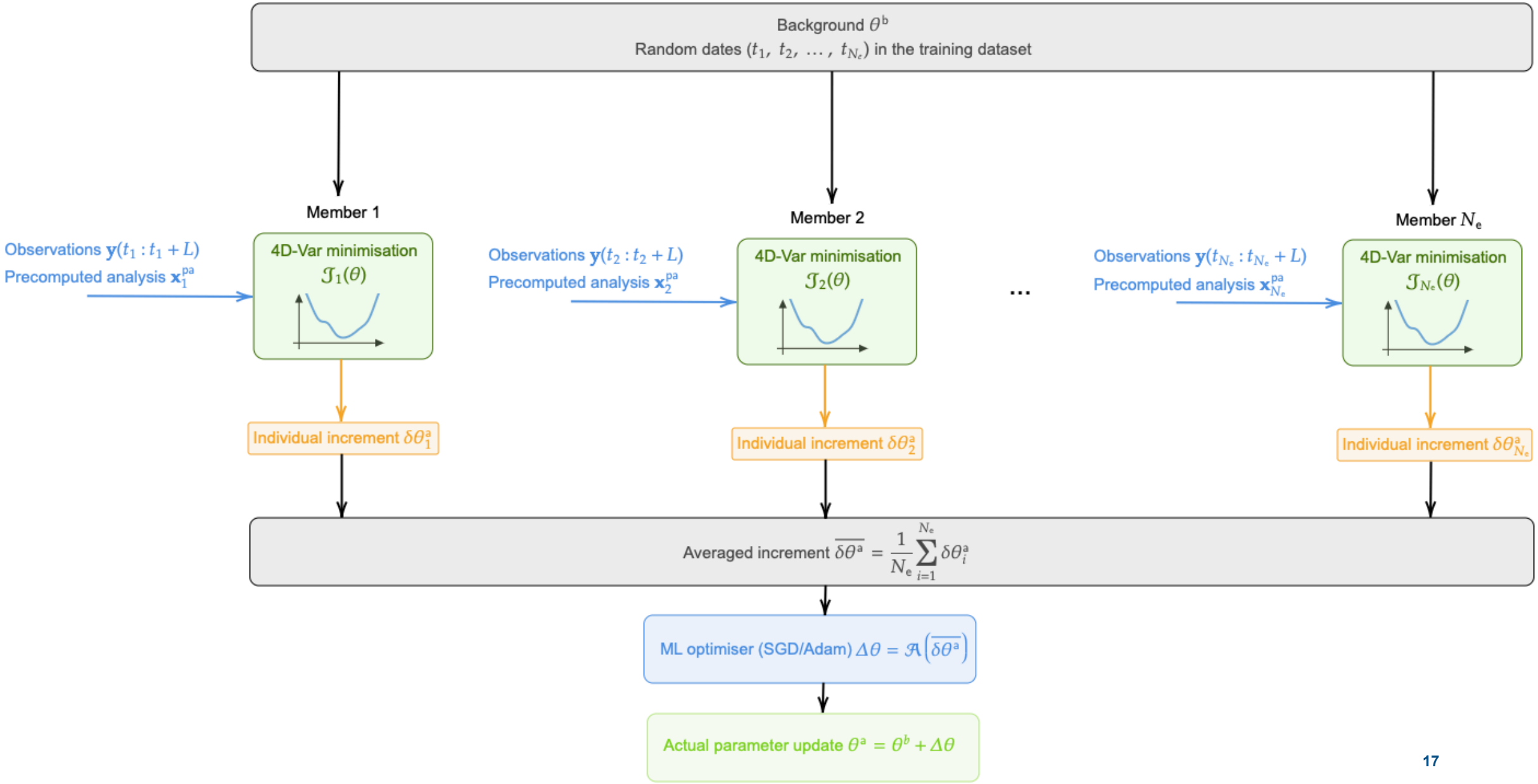
- In a batched gradient descent algorithm, the parameter update rely on the averaged gradient over a (mini) batch:

$$\overline{\nabla_\theta \mathcal{L}\left(\theta(t)\right)} = \frac{1}{N_e}\sum_{i=1}^{N_e} \nabla_\theta \mathcal{L}^i\left(\theta^i(t)\right)$$

- Pushing the analogy further, we define a batched-variant of NN 4D-Var, where several windows are processed in parallel

- This effectively multiplies by N_e the number of observation windows that the NN see in a fixed number of optimisation steps

Batching NN 4D-Var



Implementation of the batched NN 4D-Var

- We built the batched NN 4D-Var on the EDA framework
- Base time → fictitious time to synchronise the ensemble members
- Each member gets its own (random) date from the dataset:
  - Needs to retrieve observations, initial conditions, etc.
  - Needs to clean-up, with additional safeguards to ensure that members do not interfere with each other
- Synchronisation step added to merge individual analyses
- This is where the NN optimisation step takes place!

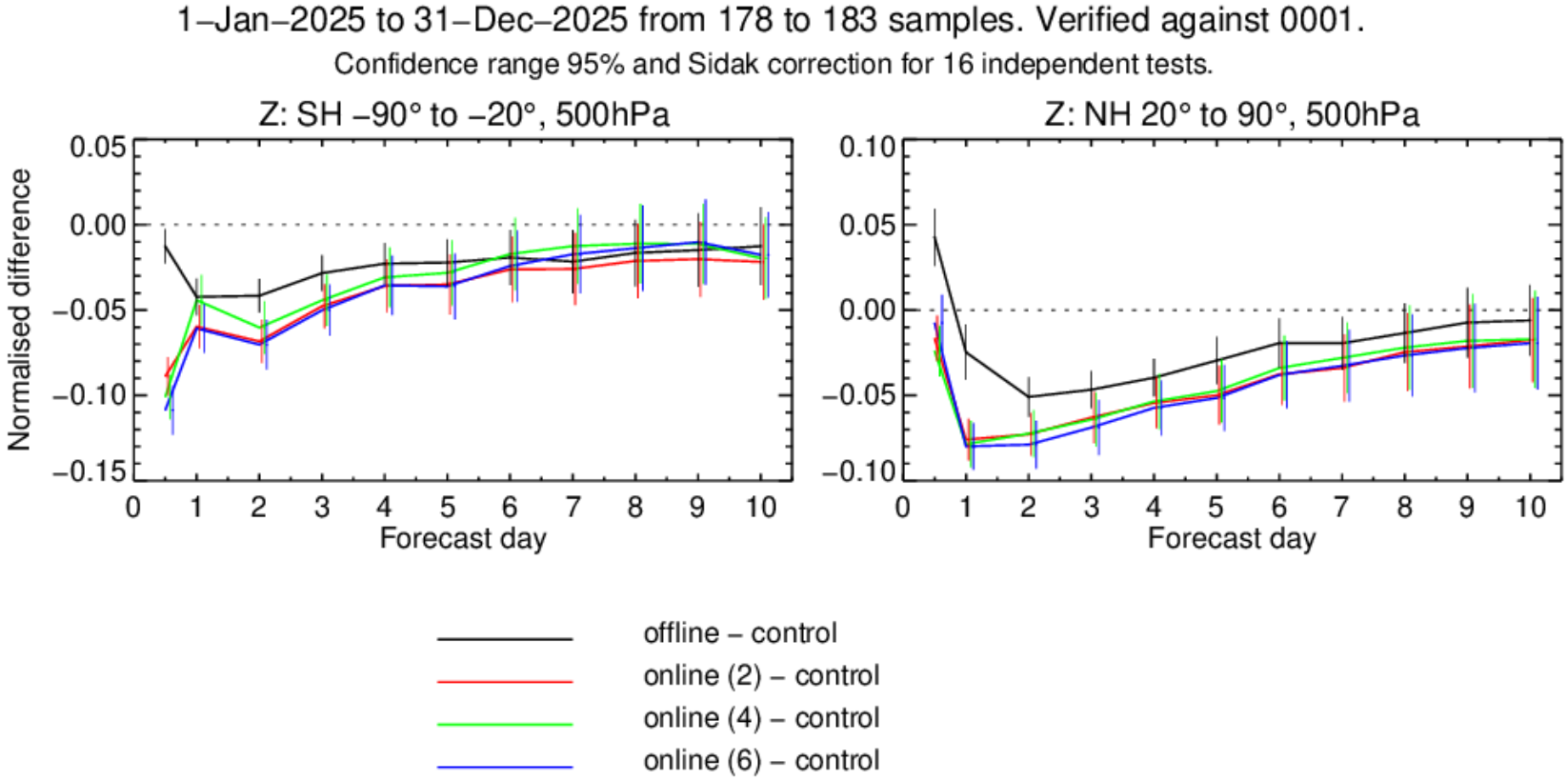


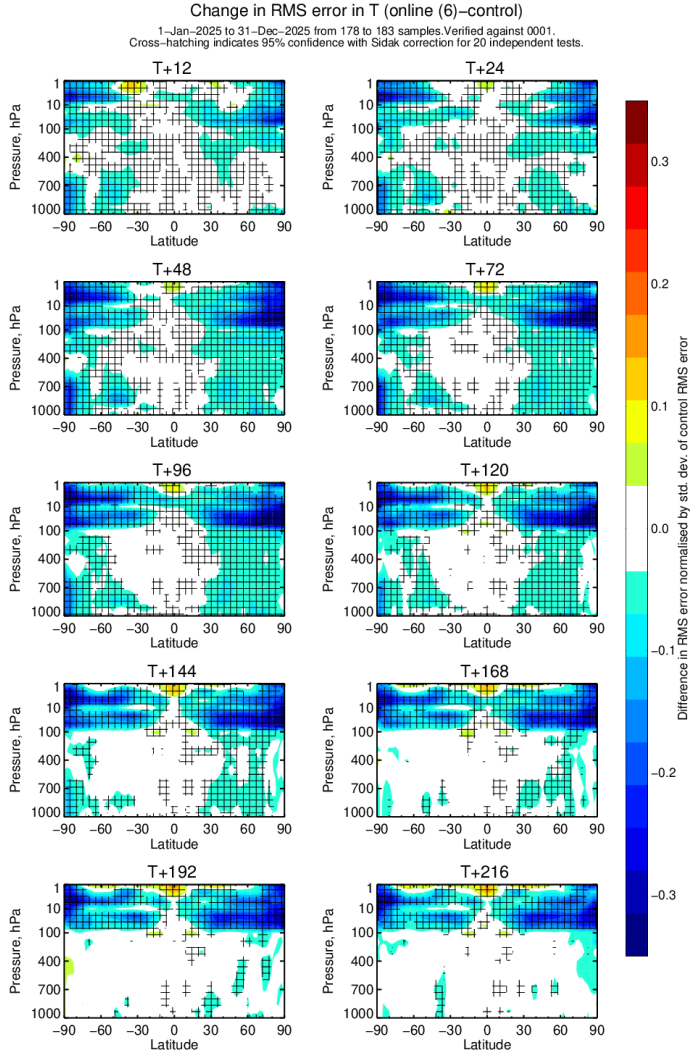
ECMWF

Accuracy of the online hybrid model

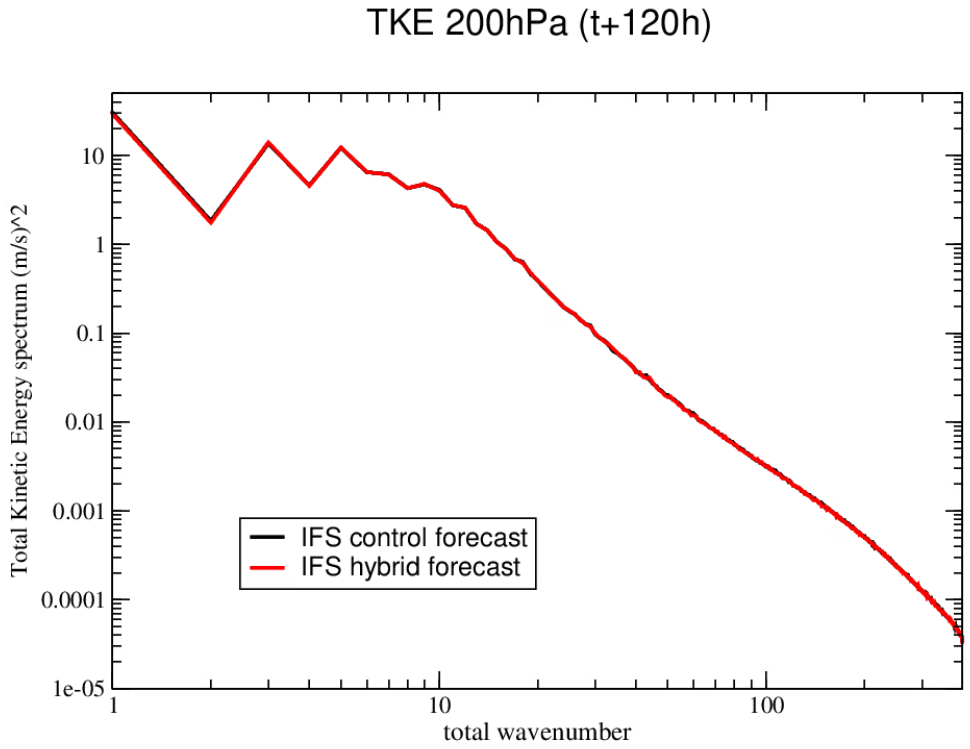
- Online training at TCo399, with 8 members
- Training dataset: 732 48h-windows over 2024
- Hybrid model evaluated in forecast-only experiments in 2025 (unseen data) after approximately 2, 4, and 6 fictitious months of training

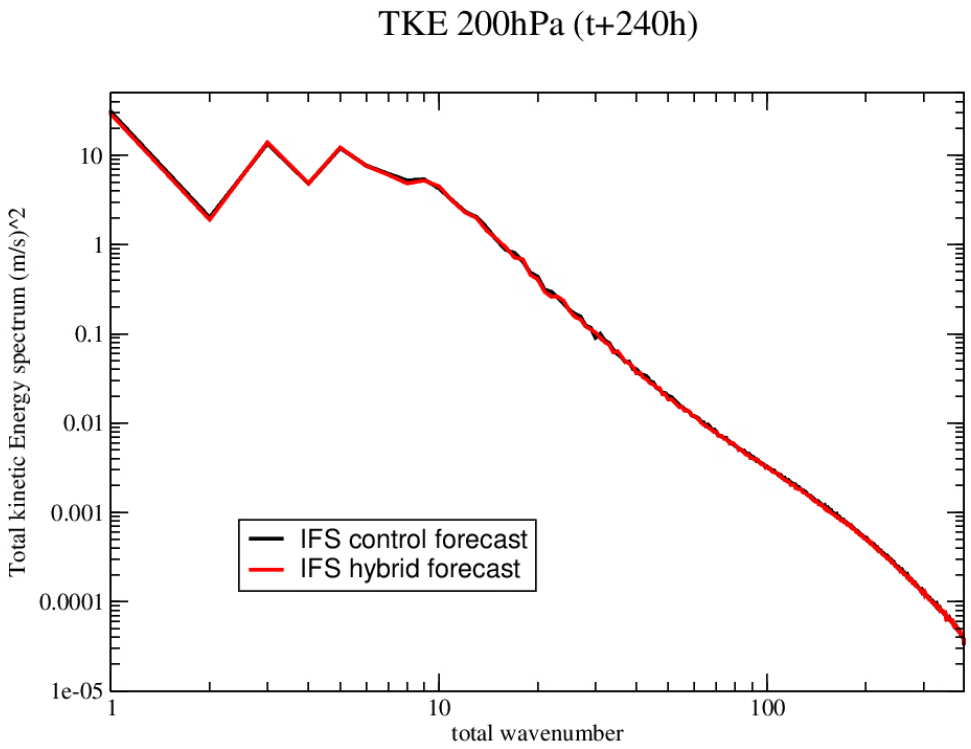
Fictitious training time	Optim. steps	DAWs	Epochs
2024-01-01 to 2024-03-01	31	248	0.34
2024-01-01 to 2024-05-02	62	496	0.68
2024-01-01 to 2024-07-01	92	736	1.01





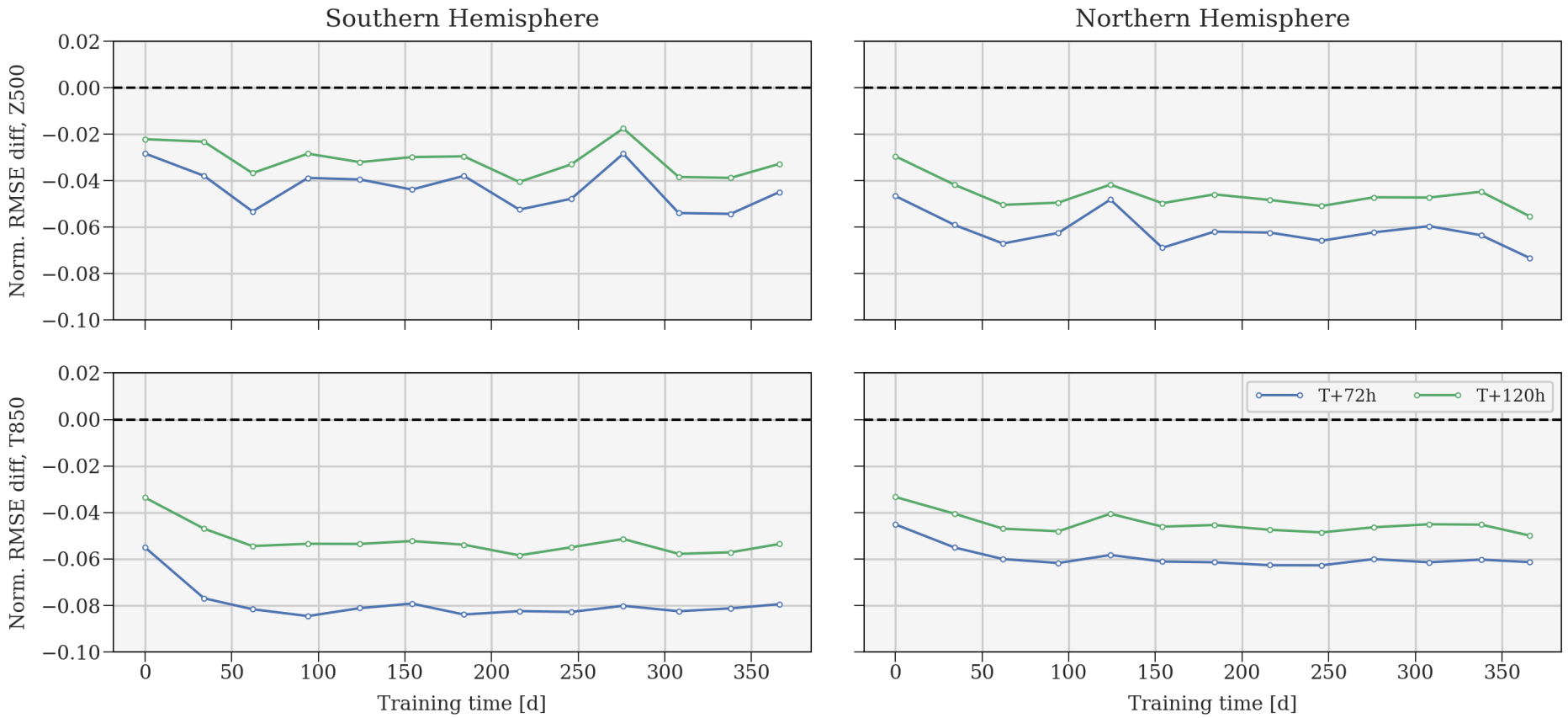
Physical consistency





- Dynamically indistinguishable from the IFS forecasts

How fast does the model learn?



- Most of the improvements are learned in the first ~100 days
- Longer datasets (multiple years) may be needed to extract more information

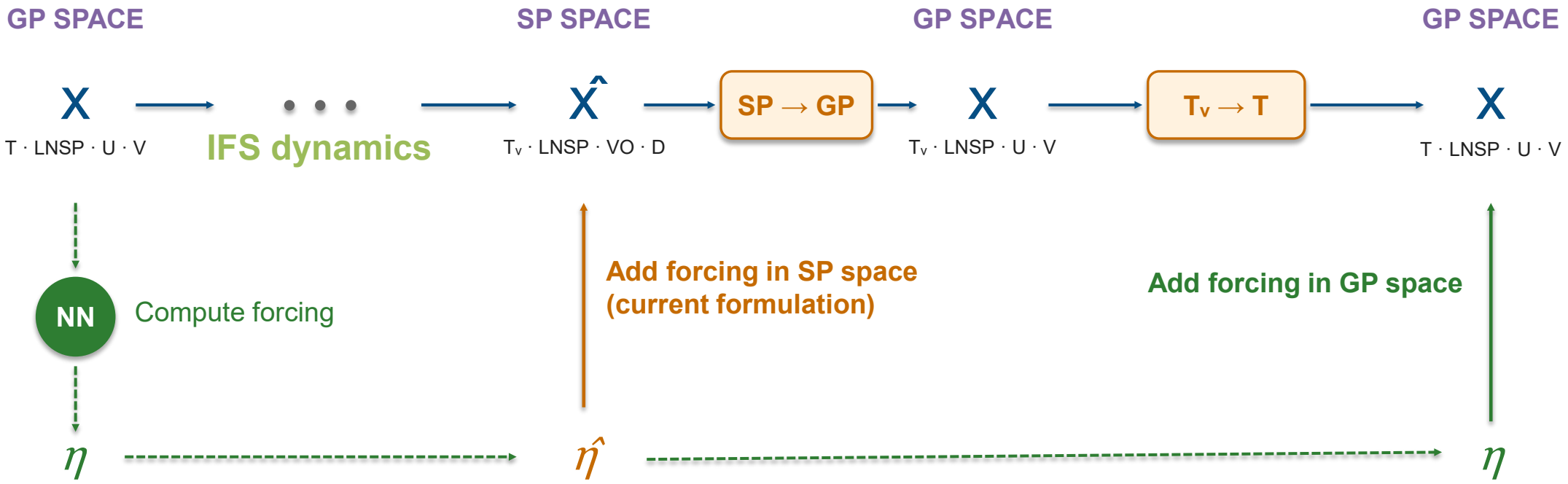
From a robust training system to a better hybrid model

Batched NN 4D-Var provides a robust framework for training the correction

Where we expect further gains:

- Revisit the corrected variables and where the forcing enters the IFS
- Improve tropical performance
  - Represent humidity
  - Implement and evaluate NN with spatial footprint via Torch C++
- Learn errors over longer timescales
  - Further extension beyond 48 hours has not yet improved results
  - Performance my be limited by the current weakly non-linear formulation of incremental 4D-Var

Where should the NN forcing enter the IFS time step?



- In most cases, a forcing in GP space requires an additional SP to GP transformation
- But it also provides means to correct directly T instead of Tv, consistently with offline pre-training
- In addition, in GP space, the forcing for winds is added to U/V instead of VO/D (in SP space)

U/V is more natural for a NN-based correction

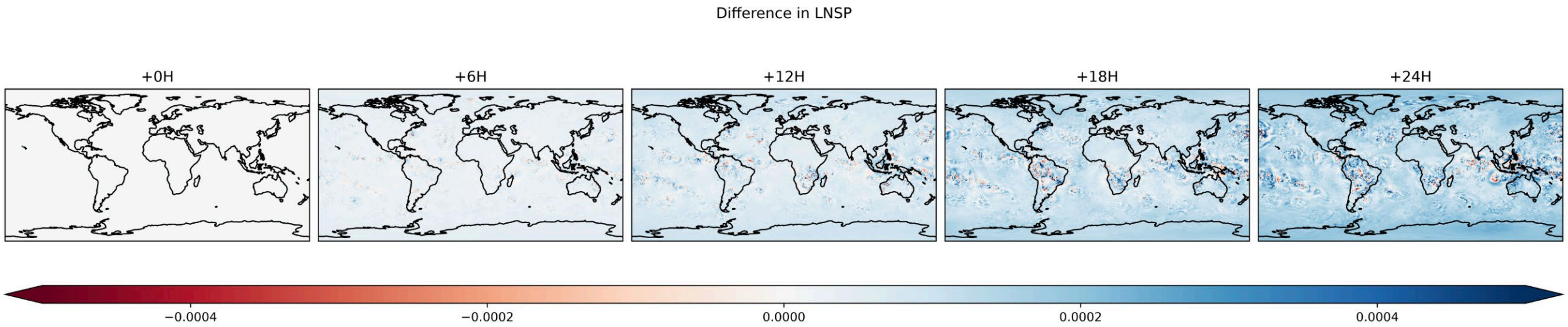
- The NN is evaluated in GP space, where the winds are represented by U/V

- VO/D satisfy zero-mean constraints by construction in the spectral model. In practice, a non-zero mean forcing for D generates a slow drift in LNSP

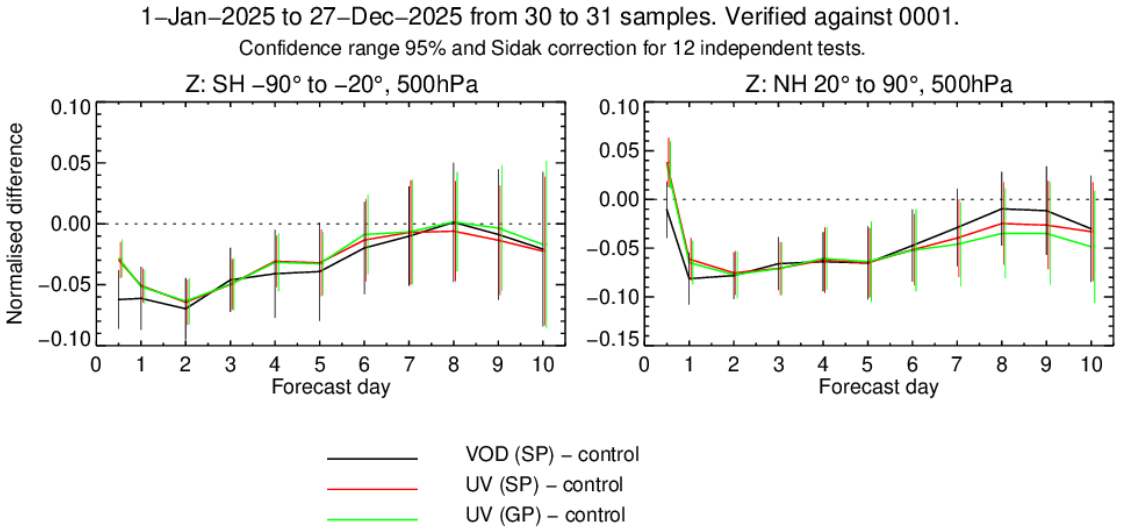
- Furthermore, increments for U/V are larger scale than increments for VO/D, and are hence likely more predictable

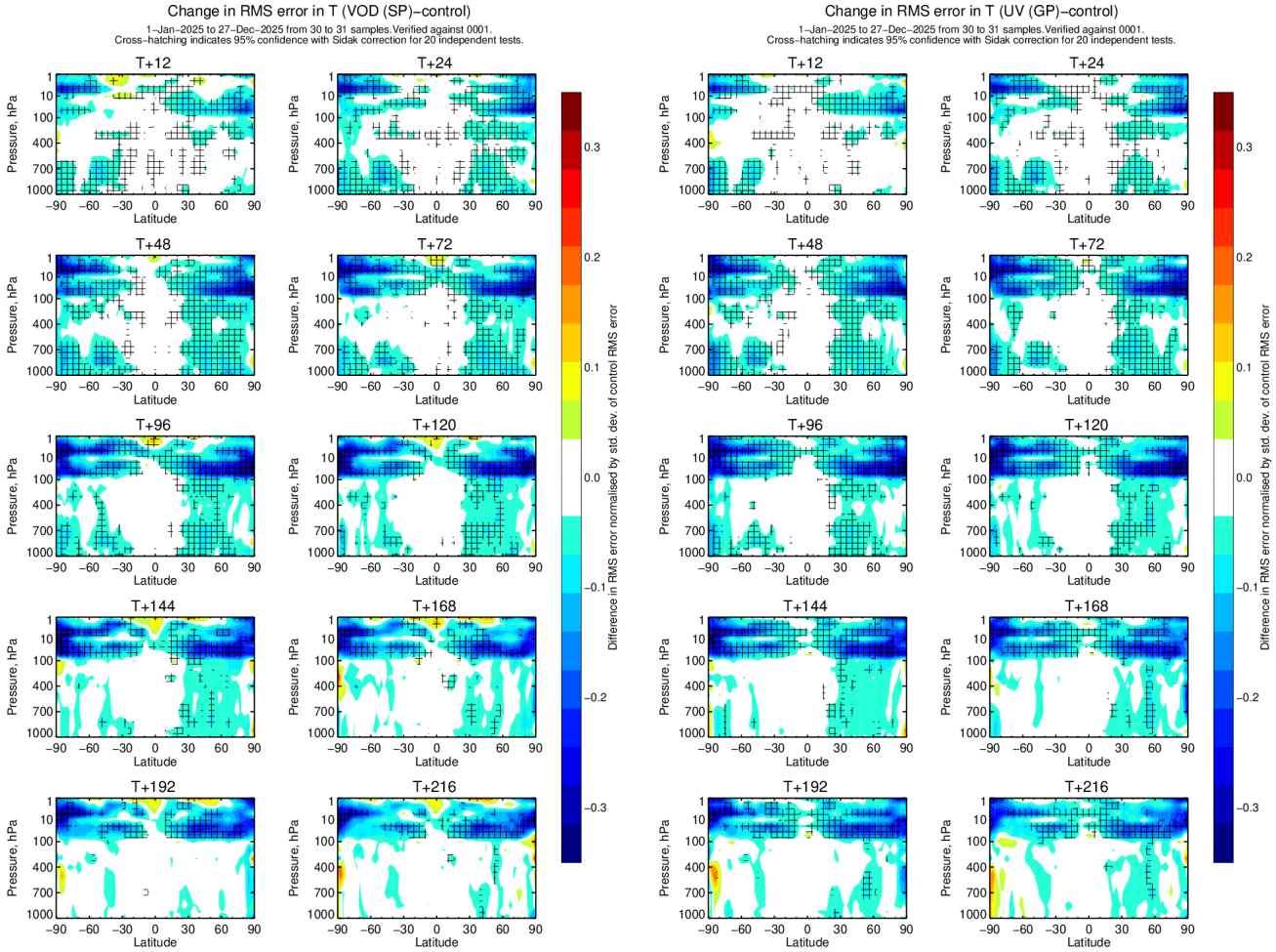
Relative MSE on the validation set

Name	Lnsp	T	Vo	D	U	V
No correction	1.000	1.000	1.000	1.000	1.000	1.000
VOD NN	0.794	0.610	0.862	0.918		
UV NN	0.820	0.615			0.766	0.834



U/V is more natural for a NN-based correction



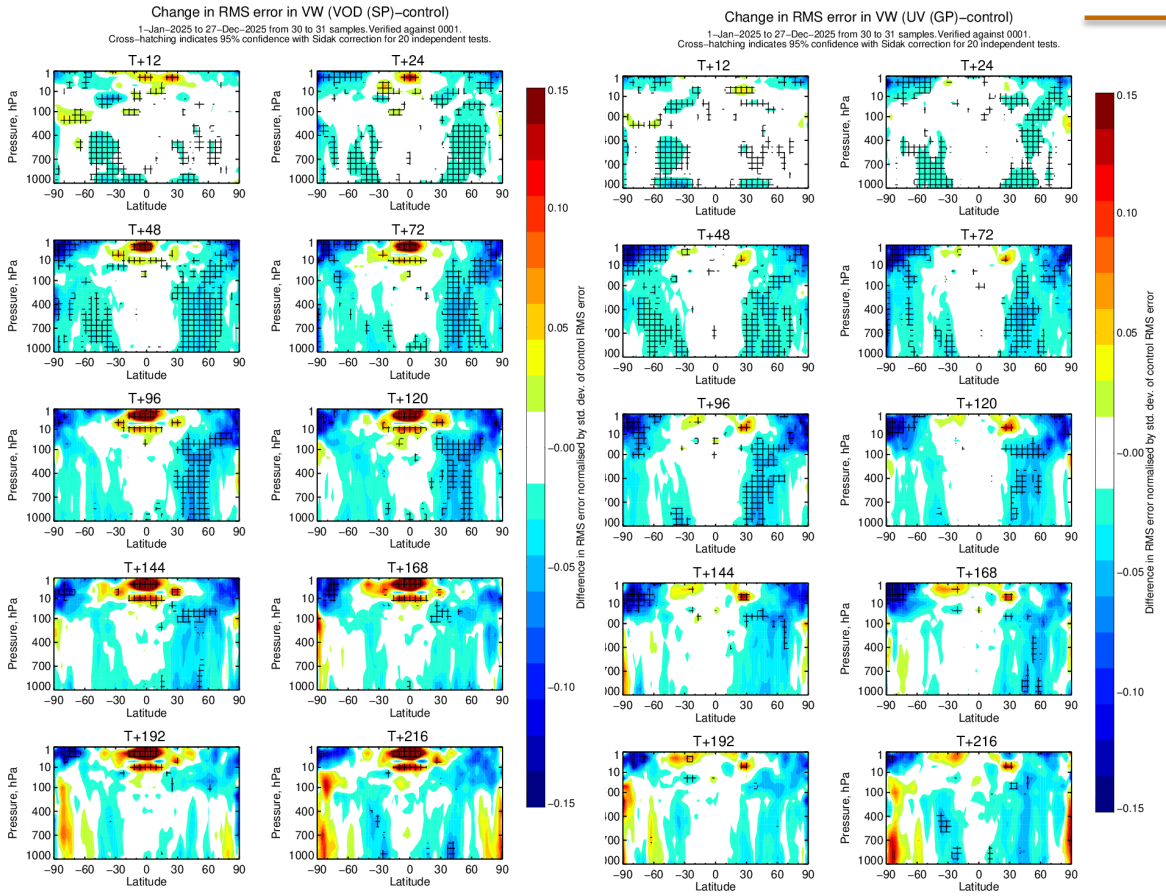


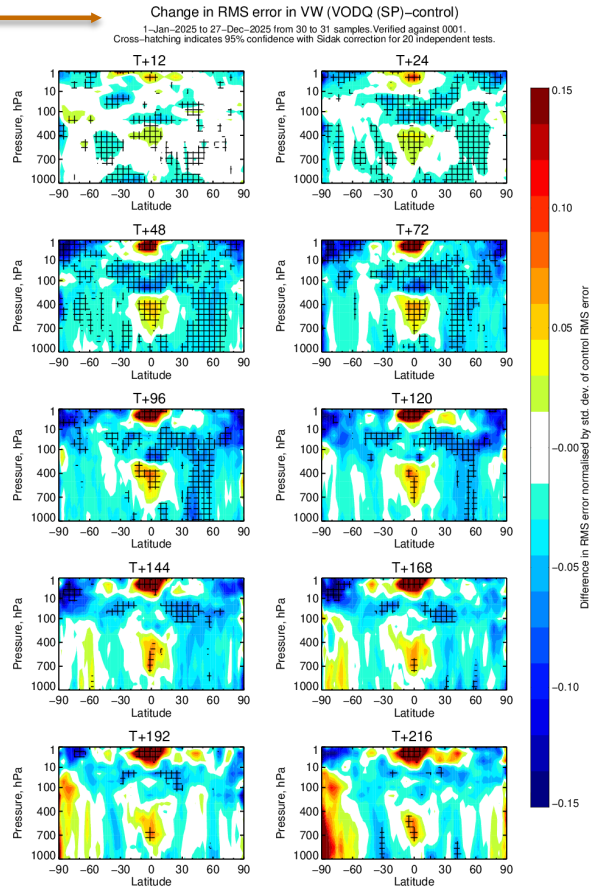
- Firsts results indicate that the accuracy of a NN trained with U/V is on par with that of a NN trained with VO/D

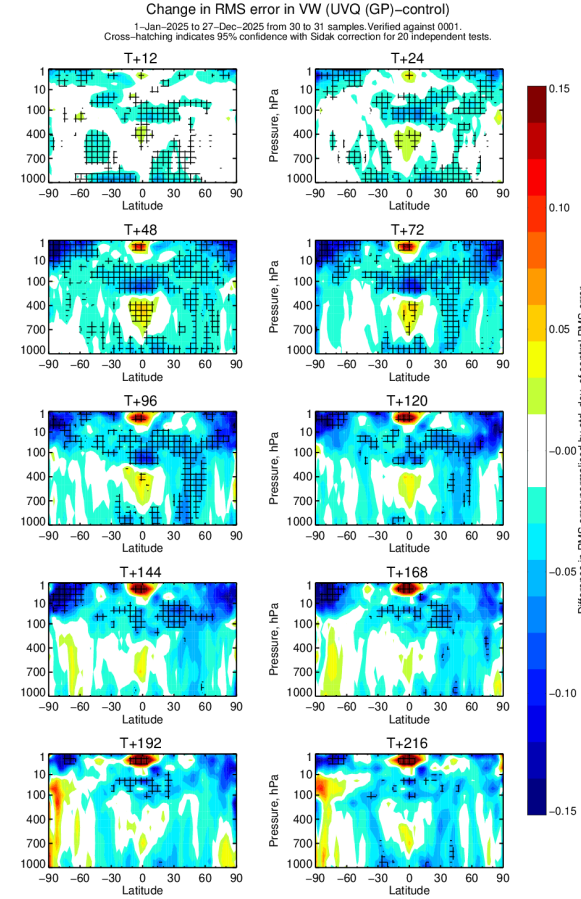
Adding humidity to improve the tropics

- The poor performance in the tropics was partially related to the LNSP drift in the VO/D formulation
- Q plays a significant role in the tropics: we decided to include Q in the NN

Add Q, Reactivate the tropics

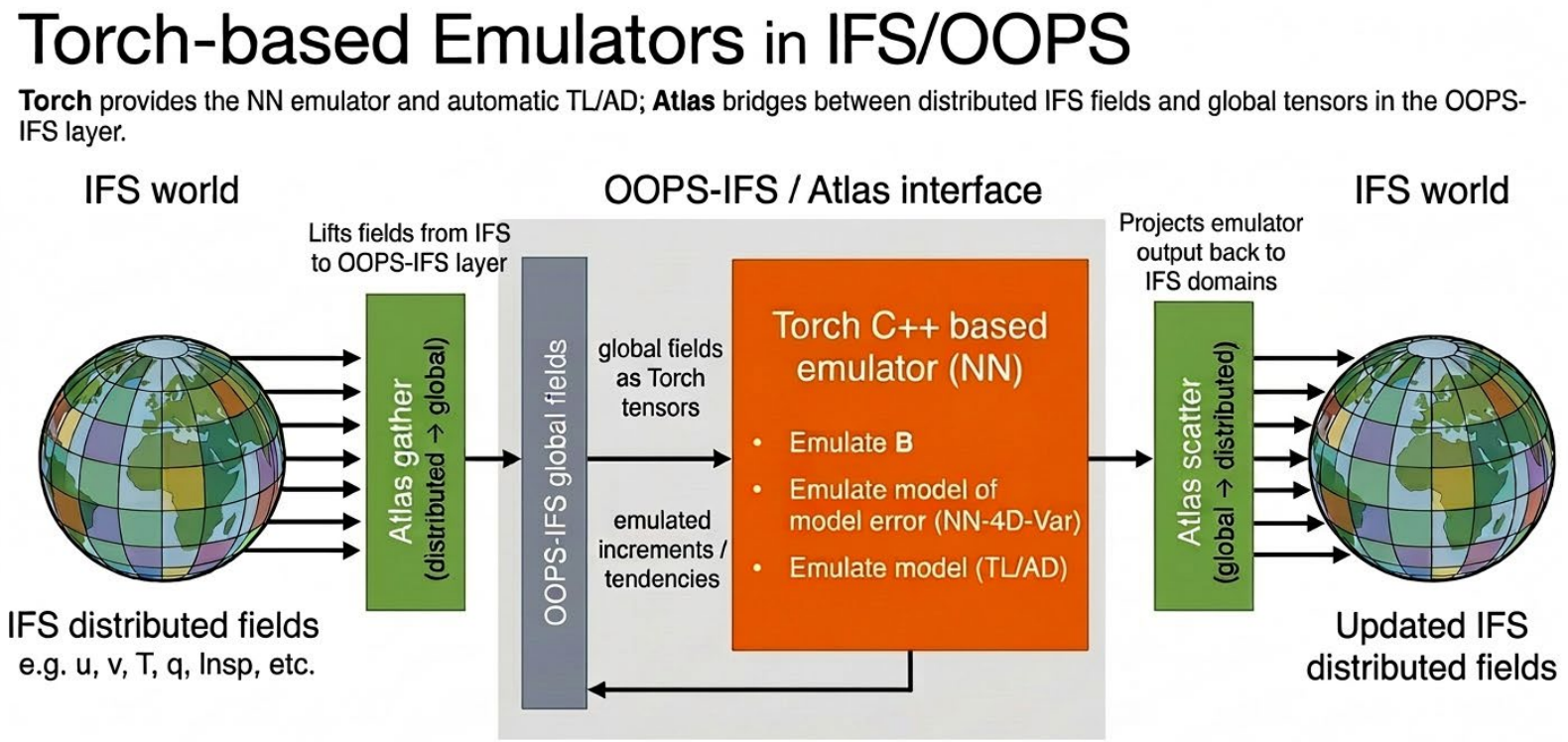






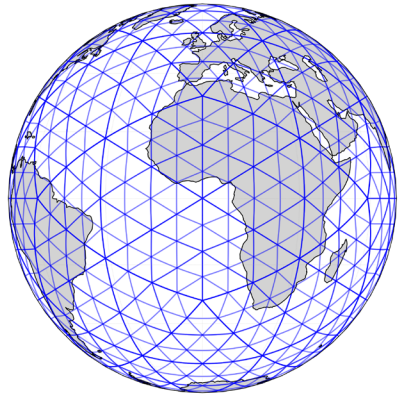
Enabling more expressive NN correction with Torch C++

- NN library in Fortran: only supports vertical / column NNs
- New solution: combine Atlas (for communication) and the C++ API of Torch (for the NN)
- **Strategic** effort with great help from Olivier Marsden, Willem Deconinck, and Benjamin Ménétrier (Met. Norway)



Torch provides the **NN emulator** and **automatic TL/AD**; Atlas bridges between **distributed IFS** fields and **global tensors** in the OOPS-IFS layer.

Towards Graph NNs



- Torch opens a path beyond independent column networks
- A Graph NN could communicate between neighbouring grid points
- Key choices include working resolution, message-passing depth, and expressiveness
- At T31 resolution (the NN resolution in the online experiments) the GNN marginally more accurate than the column NN after the offline pre-training step
- This needs to be confirmed in the online experiments, but we expect the GNN to shine at higher resolution.

Relative MSE on the validation set at T31

Name	LNSP	T	U	V	Q
No correction	1.000	1.000	1.000	1.000	1.000
Column NN (~1M parameters)	0.864	0.662	0.806	**0.866**	0.693
GAT (~10M parameters)	**0.846**	**0.643**	**0.789**	0.867	**0.690**

Conclusions

- We have developed a robust batched NN 4D-Var system to train a NN-based correction of model error
- The batched NN 4D-Var system is unique:
  - It leverages the existing 4D-Var and the EDA frameworks to substitute stochastic gradient descent
  - It is, to our knowledge, the only hybrid model in which model error is learned directly comparing model forecasts to observations
  - The hybrid model forecasts are dynamically indistinguishable from that of the IFS
- Several extensions are already being explored to both further improve the performance of the system and reduce its cost
- An exciting avenue that we are beginning to explore is the application of the NN correction in a cycled 4D-Var setup:
  - Improve the accuracy of the analysis;
  - Get a consistent treatment of systematic model errors in both analysis and forecast steps.