Poster X5.128 : EGU26-11742

Emulating tropospheric chemistry mechanisms with deep neural networks

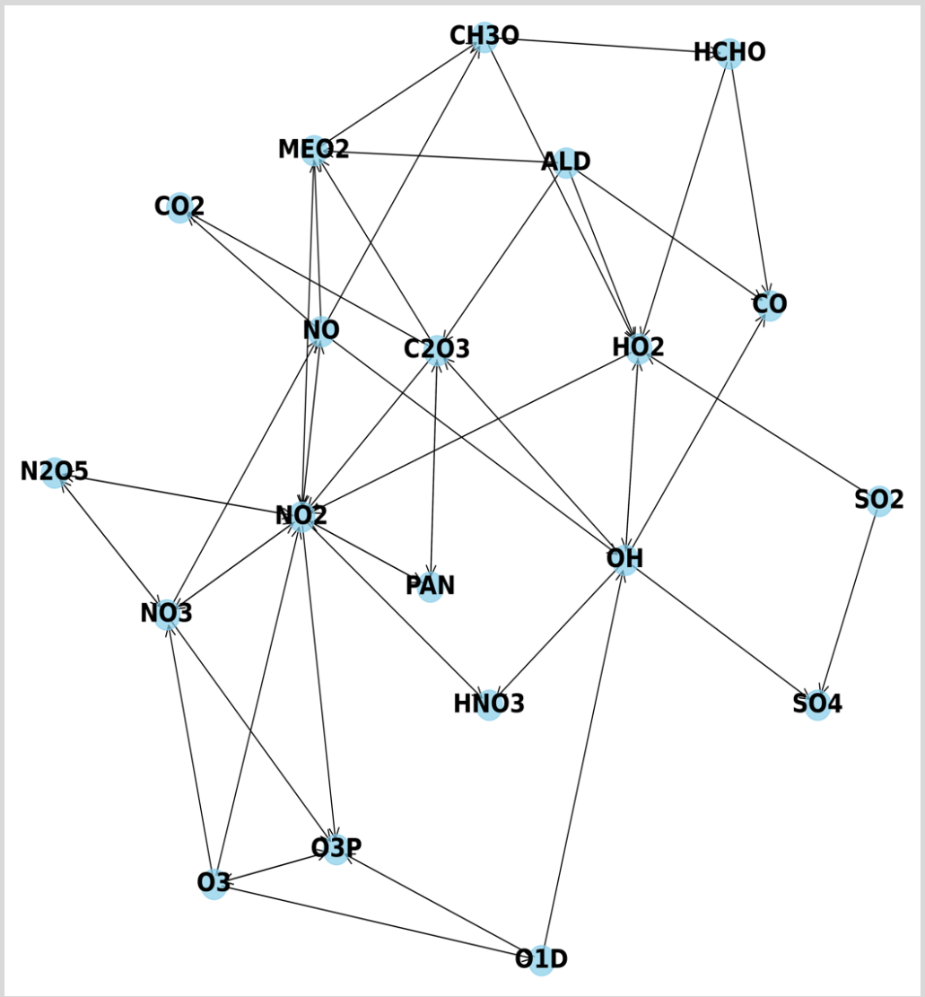
Alessio Melli[1], Hervé Petetin[1], Camille Mouchel-Vallon[1], Isidre Mas Magre[1], Oriol Jorba Casellas[1], Klaus Klingmüeller[2], Sergey Gromov[2], Leon Kuhn[2], Rolf Sander[2], Timothy Butler,[3] Markus Thürkow[3], Andrea Pozzer[2,4]
[1]Barcelona Supercomputing Center, Earth Science, Barcelona, Spain; [2]Atmospheric Chemistry Department, Max Planck Institute for Chemistry, Mainz, Germany; [3]Institute of Meteorology, Freie Universität Berlin, Berlin, Germany; [4]Climate and Atmosphere Research Center, The Cyprus Institute, Nicosia, Cyprus

Context: Solving the chemistry represents a significant (often dominant) part of the computational cost of chemistry-transport model (CTM) simulations.

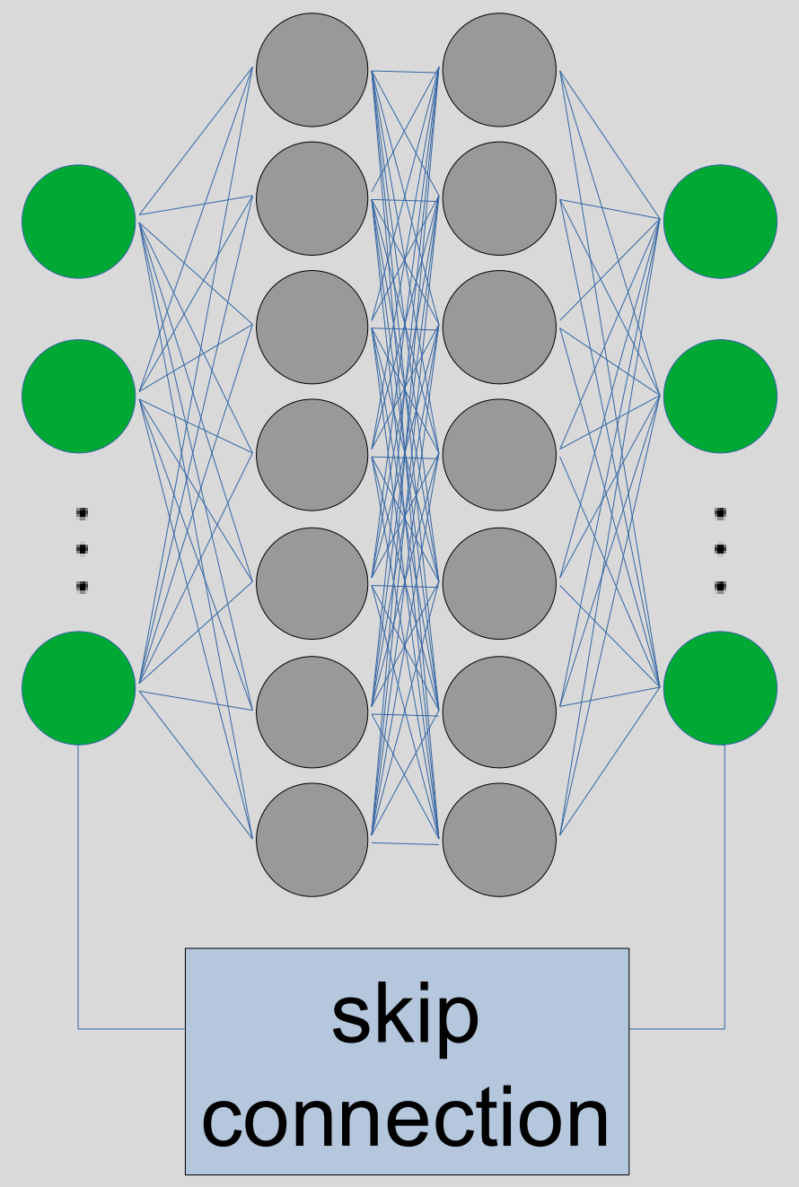
Objective: Speed-up CTM simulations using deep learning-based emulator of the chemistry

Case study: POLLU, a simplified tropospheric ozone formation mechanism

- Stiff dynamics
- Includes 20 species and 25 reactions
- No dependence on *T* and *P*
- Constant photolysis



Architecture: Fully connected multilayer perceptron with skip connection (fc-MLP)



20 input features: conc at *t*

20 target features: conc at *t*+1

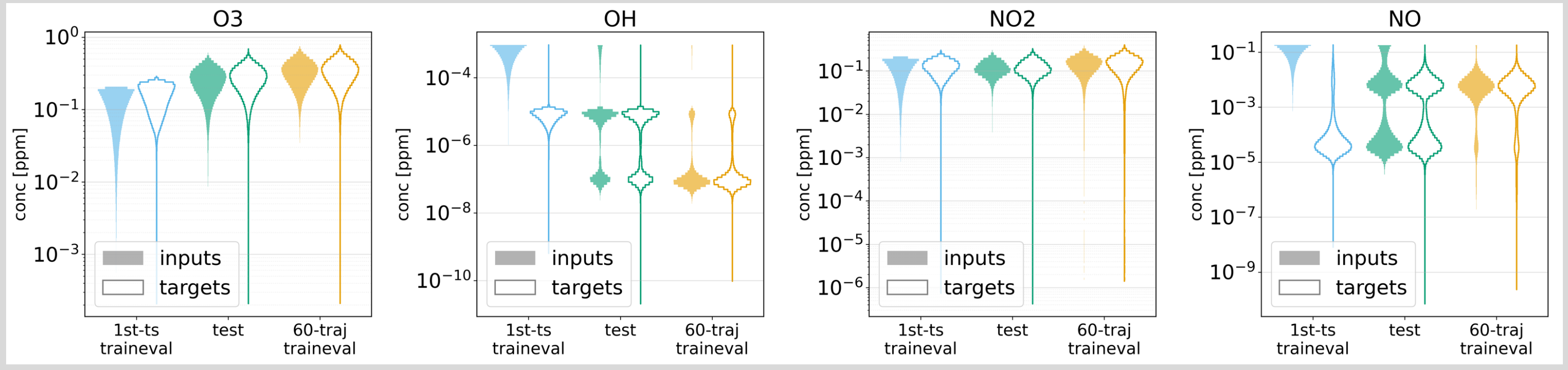
The skip connection improves gradients (but removes the design choice of predicting concentration or concentration change)

Hyperparameter tuning: tested different values of learning rate, fc-MLP width and depth. Chosen values:

- 1st-ts: `lr=1e-3`, `width=1024`, `depth=4`
- 60-traj: `lr=1e-4`, `width=512`, `depth=4`

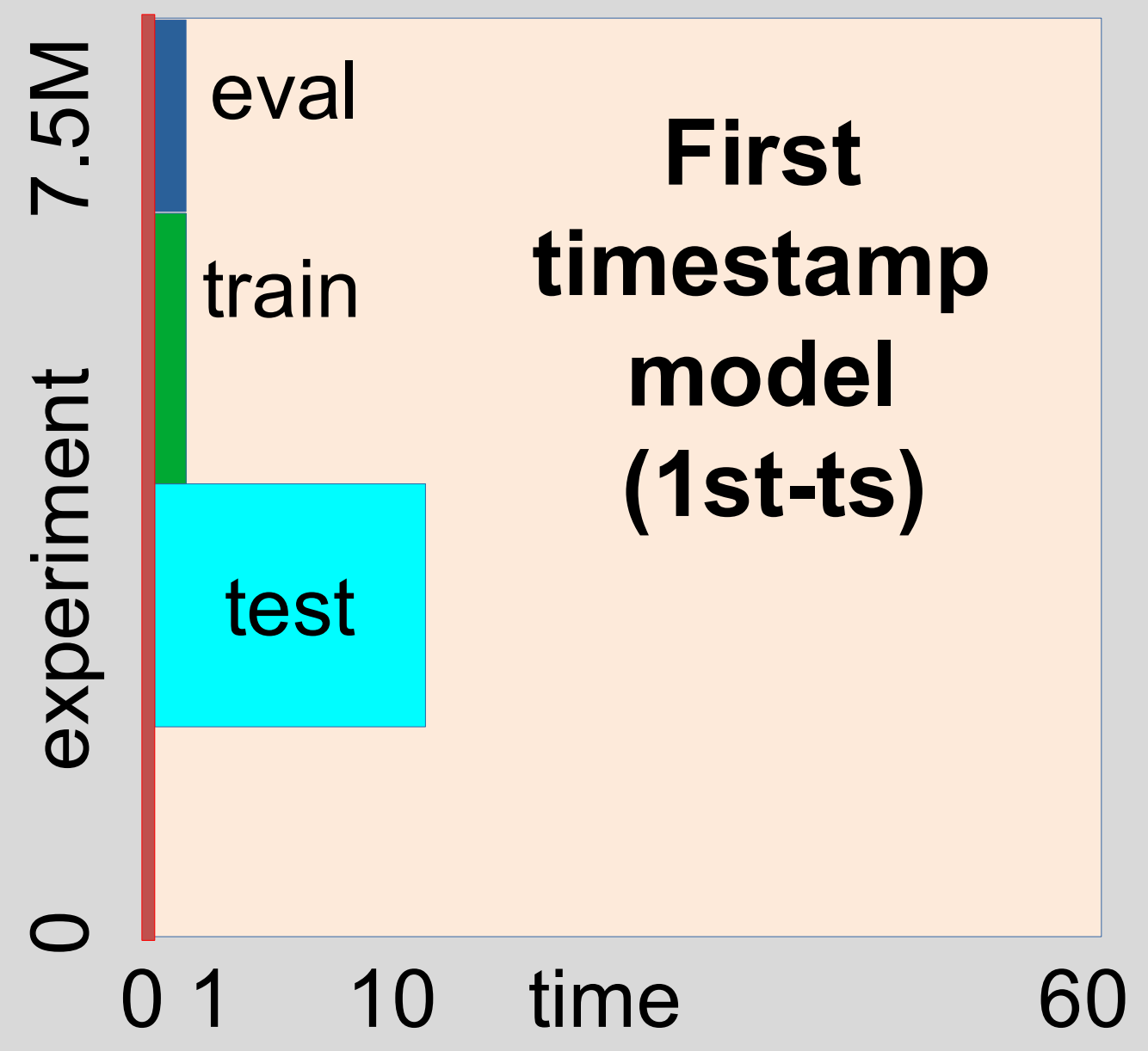
Data distribution

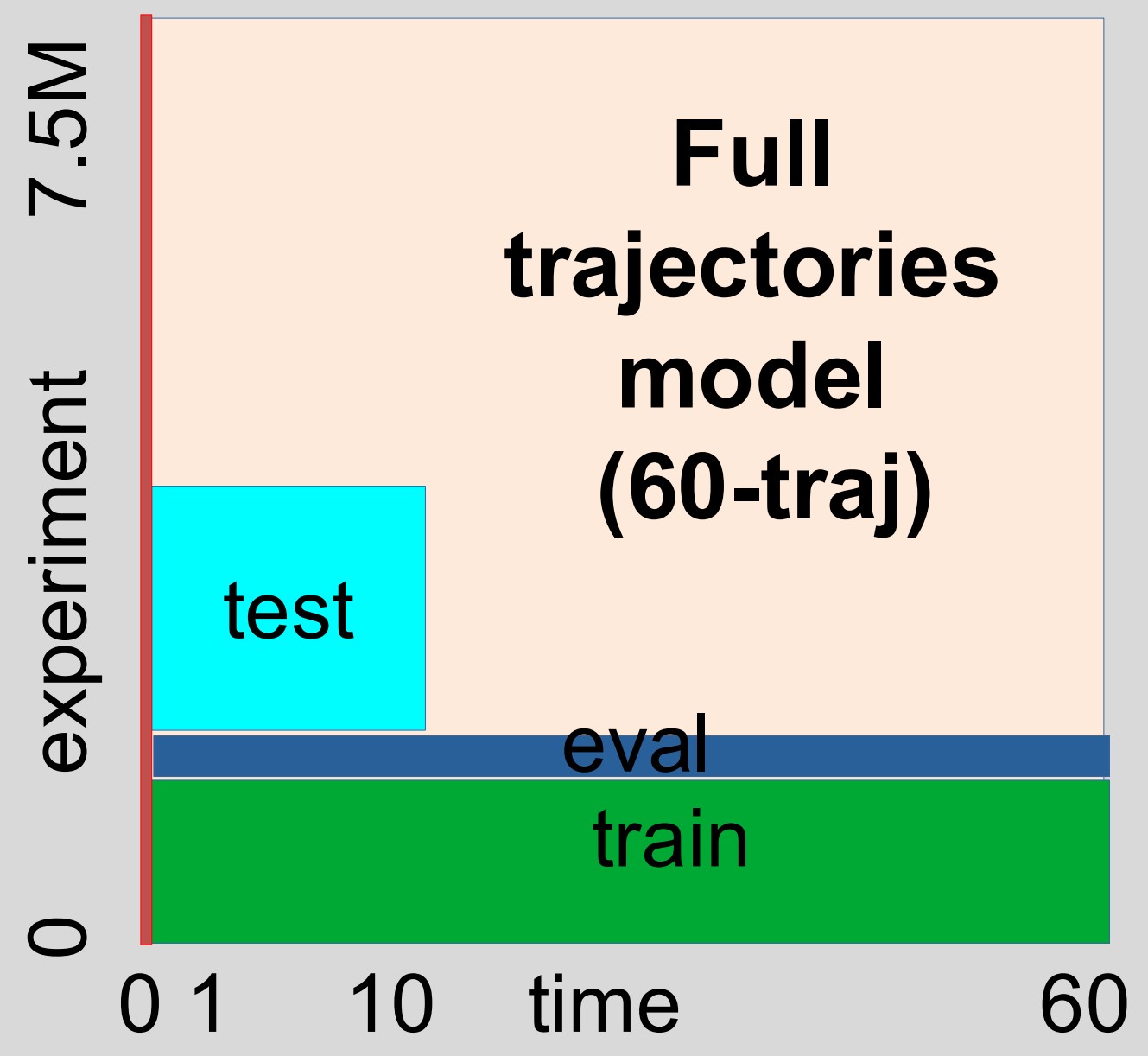
Different distributions between inputs and targets of the subset will affect performance of auto-regressive predictions



Data generation

1. Generate 7.5 millions uniformly sampled initial conditions between a given min-max range, using Sobol sequence (red line) which allows uniform filling of the 16D hypercube,
2. Run BSC's boxmodel (CAMP) for 60 timestamps of 1 minute, generating 7.5M 1h-simulations or 442.5M data points (orange area)
3. Select two different train (green area) and eval (blue area) subsets to build two datasets
4. Analogously build a test subset (turquoise area)





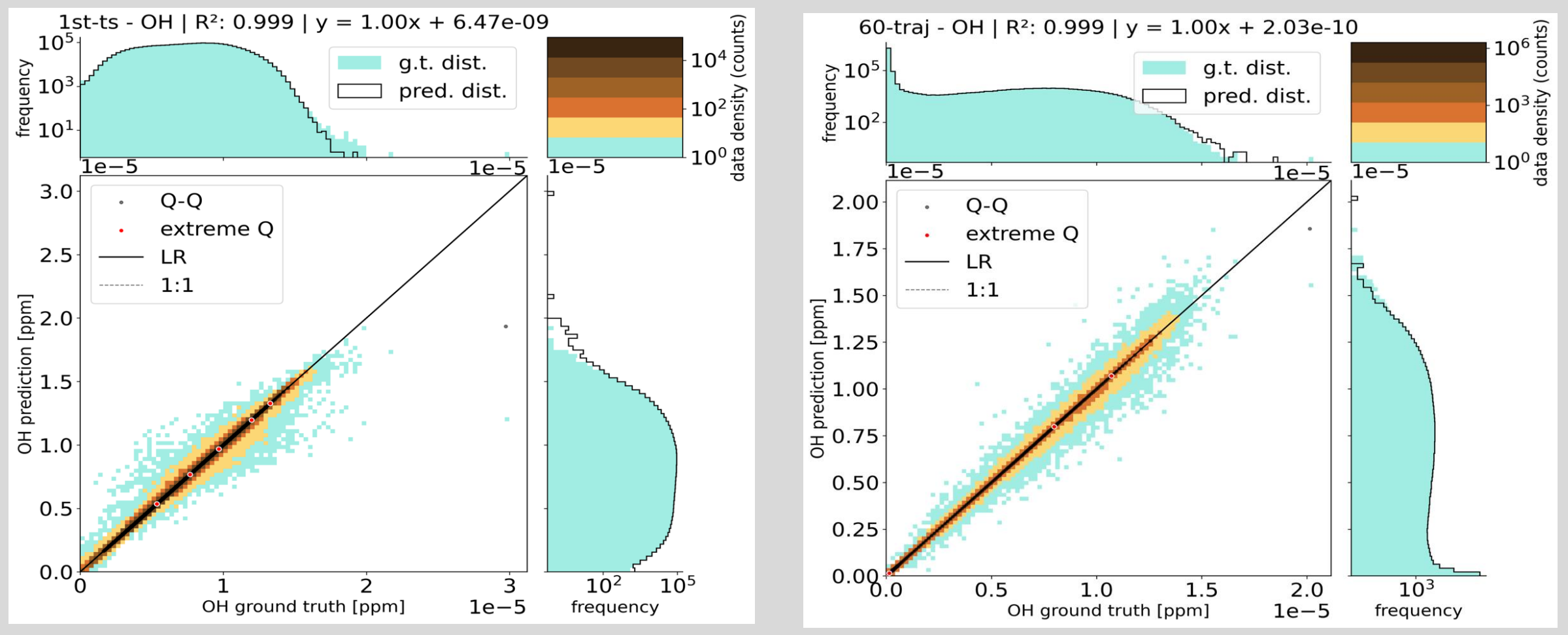
data points

- train: 3.6M
- eval: 2.4M
- traineval: 6M
- test: 13.7M

n.b.: each area of the same color has the same surface thus number of data points

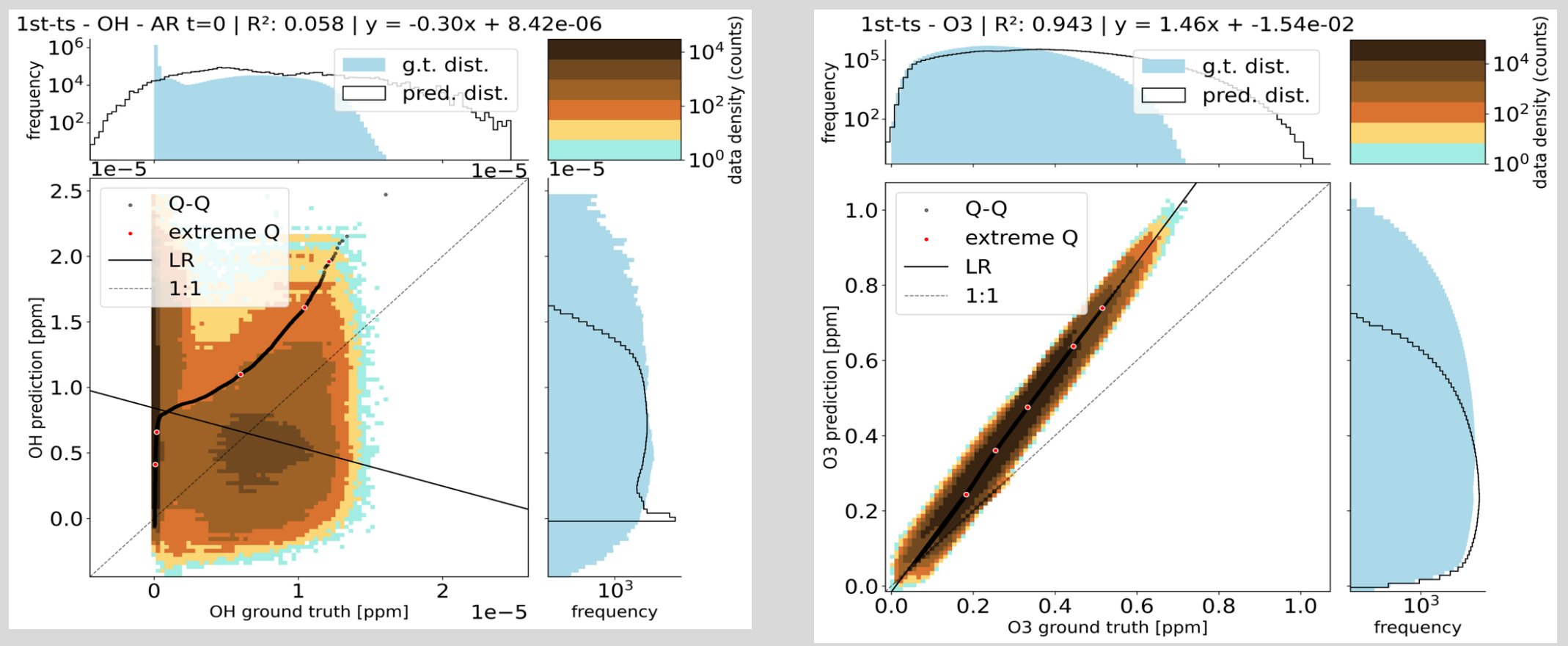
Performance of one-timestep predictions

One-step performance on the train and eval subsets is good: 19/20 species with PCC above 0.999 for both models.

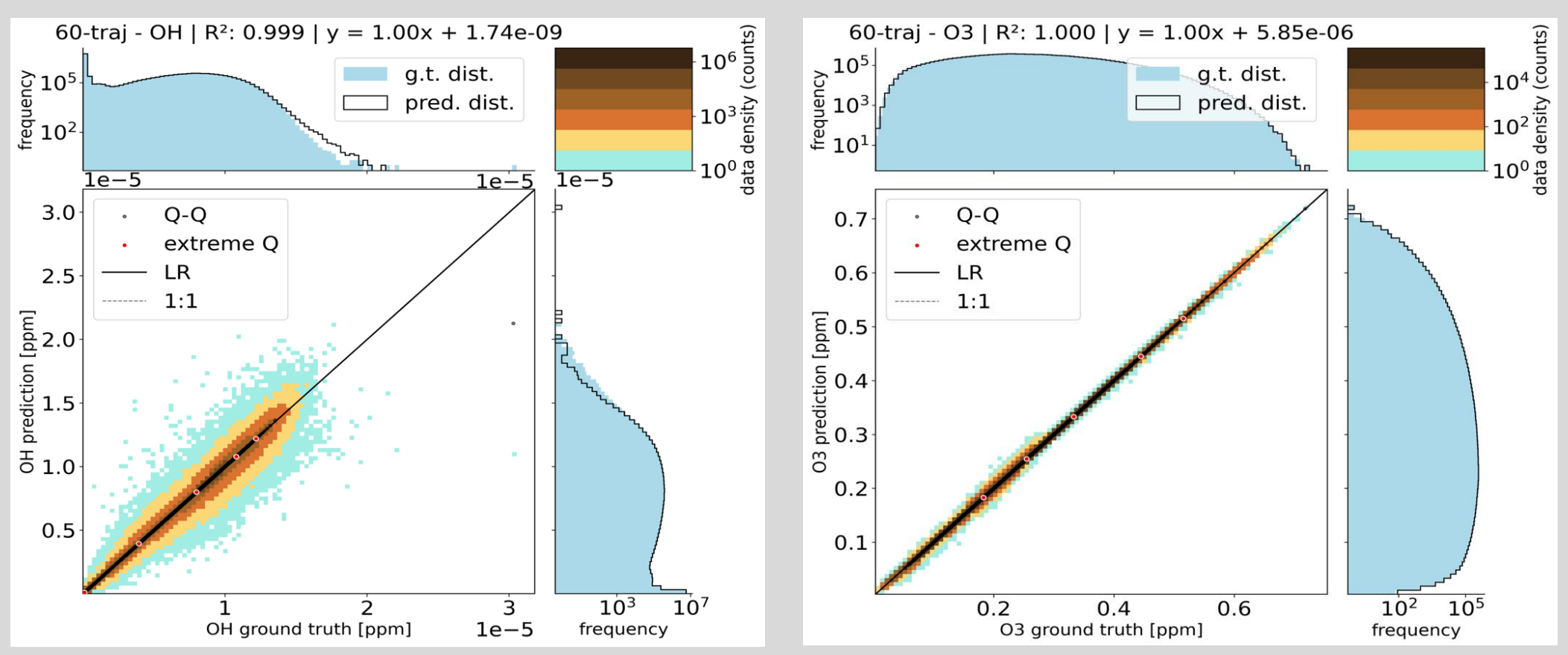


On the test subset (comparable for both models): only the 60-traj model continues to show good performance, the 1st-ts model showing strong degradation likely due to samples outside the initial range used for training. Worst (but still reasonable) performance of 60-traj model on C2O3 (R2=0.88, slope=0.95).

1st-ts



60-traj



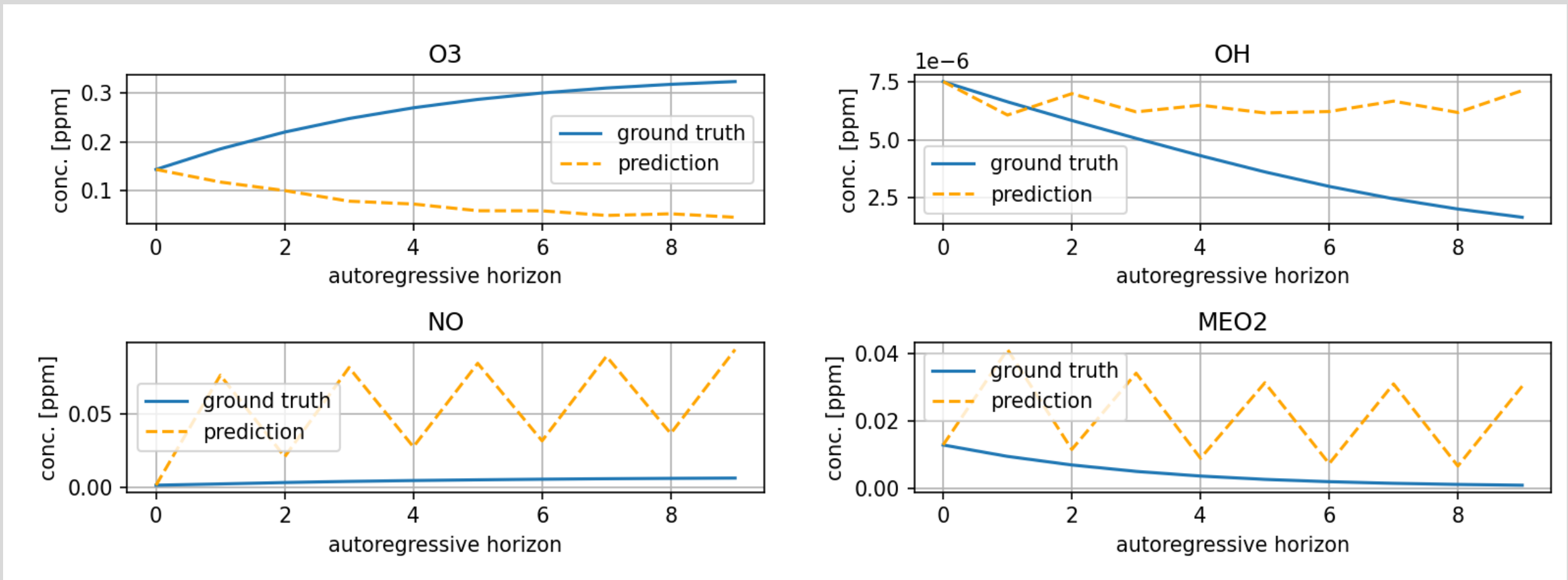
Acknowledgement

The poster part of the action PCI2024-155075-2, funded by MICIU/AEI/10.13039/501100011033 and co-funded by the European Union. HP has received funding from the Ministerio de Ciencia e Innovación (MCIN) and Agencia Estatal de Investigacion (AEI) through the Ramon y Cajal grant (RYC2021-034511-I, MCIN/AEI/10.13039/501100011033 and European Union NextGenerationEU/PRTR). CMV acknowledges his AI4S fellowship within the "Generación D" initiative by Red.es, Ministerio para la Transformación Digital y de la Función Pública, for talent attraction (C005/24-ED CV1), funded by NextGenerationEU through PRTR.

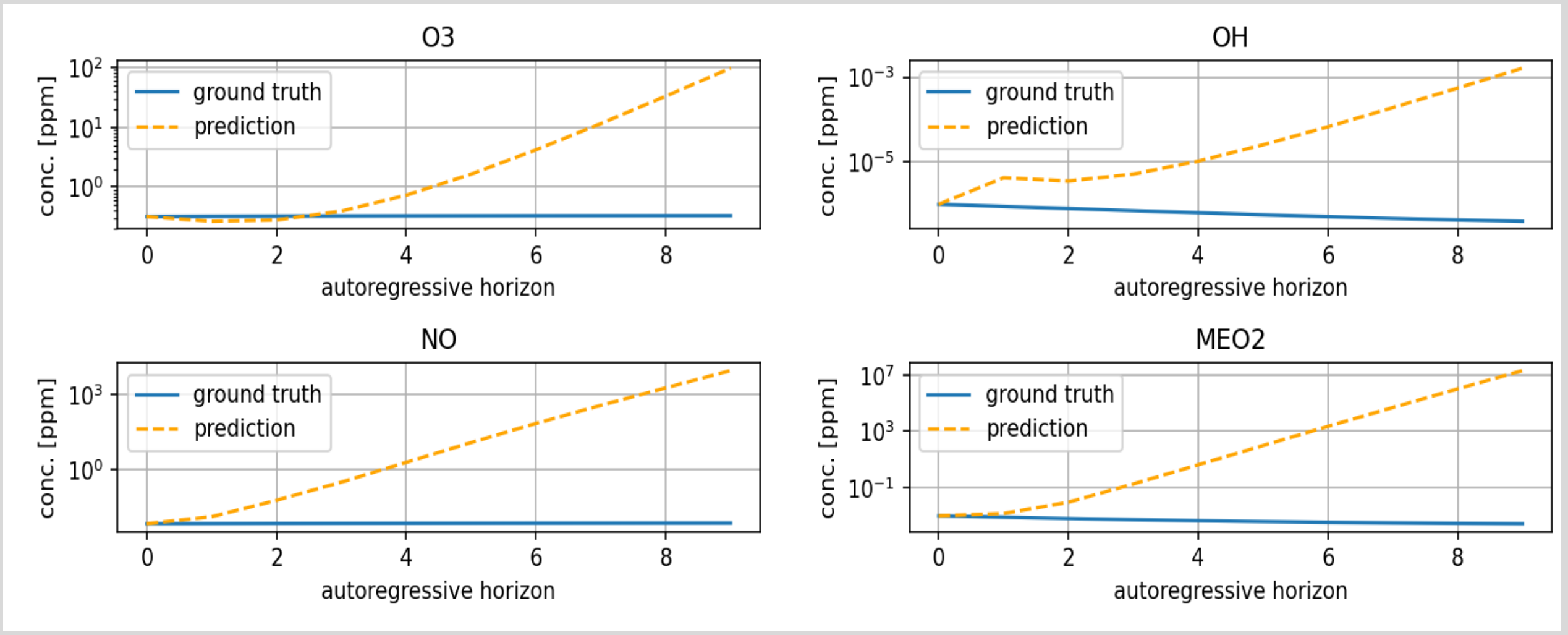
Performance of multi-timestep auto-regressive predictions

- On the eval subset, the performance of auto-regressive predictions is poor
- Training on trajectories (60-traj model) does n ot lead to improved skills for predicting trajectories
- Actually 60-traj shows even worse performance with often larger divergence

1st-ts



60-traj



Open questions

- Some species behave globally worse than others (e.g. acetyl peroxy radical C2O3) given the strongly skewed distribution. Can a tailored preprocessing or different dataset design help?
- How does such a reasonably good performance in one-step predictions degrade so fast in autoregressive mode? How to reduce these errors?

Perspectives

- Improve preprocessing and/or dataset design on short-lived species
- Include model constraints based on physics
- Explore more sophisticated AI architectures
- Assess speed-up and error accumulation in real-conditions within a CTM simulation