Neural Supermodeling

Wim Wiegerinck (w.wiegerinck@science.ru.nl) **SNN Adaptive Intelligence** *and* **Donders Institute for Brain, Cognition and Behavior, Radboud University Nijmegen, The Netherlands**

1. Supermodels (SUMOs): interactive ensemble of existing models

- **Proposal to improve climate modeling[1]**
  - As an alternative to conventional noninteractive ensemble methods
  - Models are good, but imperfect
- **Supermodel = *Ensemble* of dynamically coupled models**

Individual model dynamics

$$\dot{x}^i_\mu = f^i_\mu(\boldsymbol{x}_\mu)$$

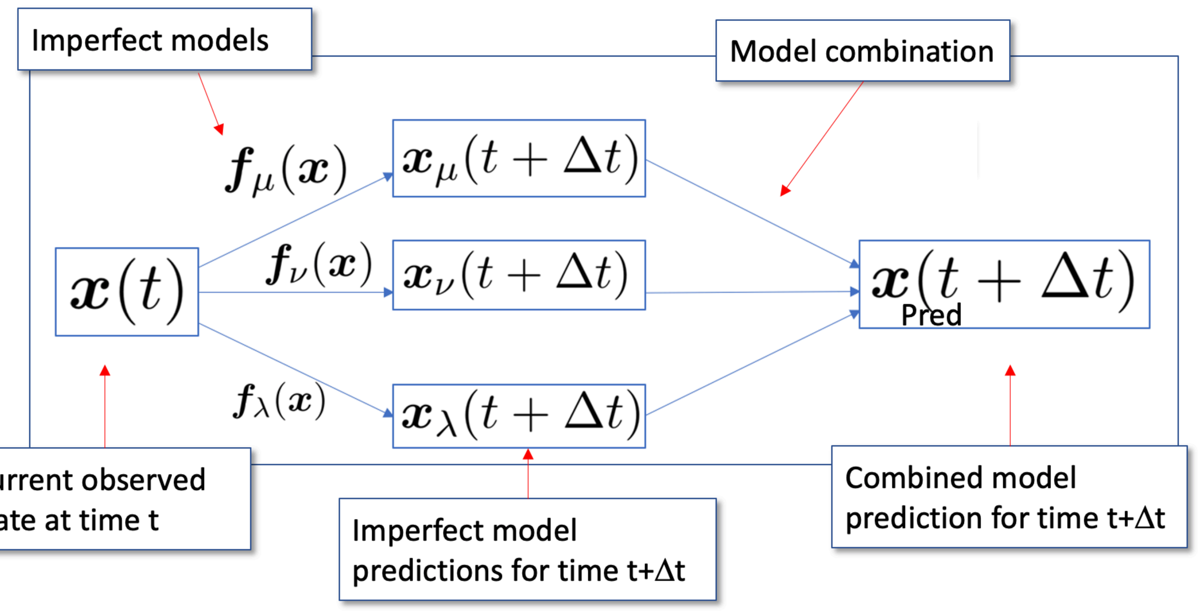
SUMO coupling by weighted averaging (see [2] and [3])

$$\dot{x}^i = \sum_\mu w^i_\mu f^i_\mu(\boldsymbol{x})$$

SUMO couplings need to be optimized

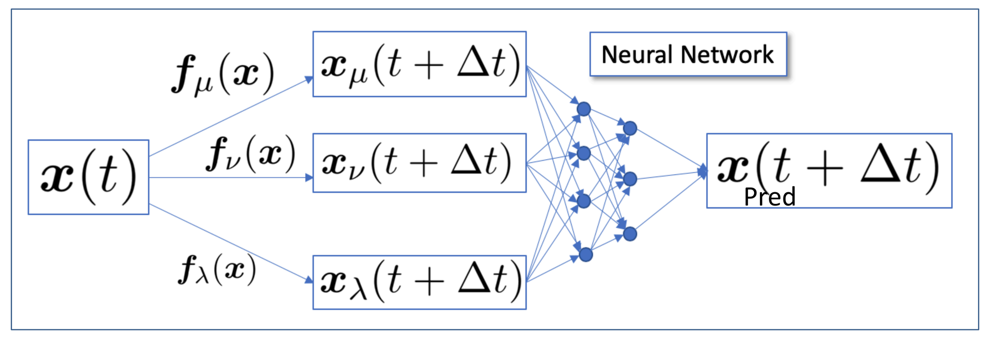
- **e.g. by minimizing short term (Δt) prediction error *E***
- ***Theoretical optimal if models are combined with Δt → 0***
- ***Works very good if Δt is small***
- ***But this may be impractical***
- ***We would like to work with larger Δt***

FOCUS** of this poster: **OPTIMIZING** the **combination** of **imperfect models** for **one step ahead prediction** for given value of **Δt



2. Neural model combination

Model combination can be done using linear models (such as e.g. weighted averaging), but also by a neural network.



3. Research Question

Can a convolutional neural network (CNN) improve linear / weighted averaging methods for model combination used in e.g. ensembles or SUMO?

Is this feasible? With so many variables involved and parameters to optimize? Based on limited amount of data?

First step: **proof of principle** with *artificial* assumed ground **truth** model and *artificial* **imperfect** models.

4. Experimental scenario

Three level quasi-geostrophic model T21 [4]

Simulates the wintertime atmospheric flow in the Northern Hemisphere quite realistically with a climatology with multiple weather regimes that are also found in observations.

Dynamical system for potential vorticity (PV)

$$\dot{q}_1 = \mathcal{J}(\psi_1, q_1) - D_1(\psi_1, \psi_2) + S_1,$$
$$\dot{q}_2 = \mathcal{J}(\psi_2, q_2) - D_2(\psi_1, \psi_2, \psi_3) + S_2,$$
$$\dot{q}_3 = \mathcal{J}(\psi_3, q_3) - D_3(\psi_2, \psi_3) + S_3,$$

Imperfect models

For benchmarking, we simulate four **imperfect models** by **perturbing parameters** of the **Truth** as in [3],

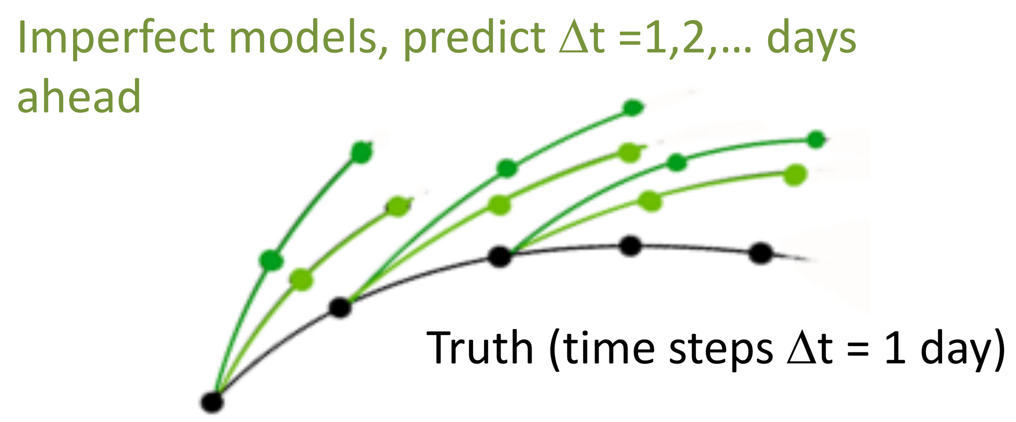
	τ_E	R_1	R_2
Truth	2.0	0.1150	0.0720
Model 1	1.5	0.1165	0.0705
Model 2	1.5	0.1130	0.0725
Model 3	2.4	0.1130	0.0705
Model 4	2.4	0.1165	0.0725

τ_E – timescale in days of the Ekman damping
R_1 – Rossby radius of deformation of the 200–500 hPa layer
R_2 – Rossby radius of deformation of the 500–800 hPa layer.

Data set

Daily observations q(x,y,z,t) $\stackrel{\text{def}}{=}$ q(t) were simulated by running the *truth* from t = 0 until t = 3000 (t in days).

Starting from each daily observations, the four imperfect models predicted Δt = 1 ... 7 days ahead. This yielded 2400 imperfect predictions $q_\mu(t+\Delta t; \mu)$ for μ=1..4 and Δt = 1 ... 7 days ahead (so 4x7x3000 predictions).



4. Linear models and deep Convolutional Neural Network (CNN)

Input and output layer

- Input: X*Y*M = 64*32*4 variables to represent imperfect model predictions q_μ(x,y,z,t+Δt) at time t+Δt and layer z.
- Output: X*Y = 64*32 variables to represent combined model prediction q_{pred}(x,y,z,t+Δt) at time t+Δt and layer z.
- Note: separate model trained for each z and each Δt

Linear models:

q_{pred}(x,y,z,t) =
- Σ_μ q_μ(x,y,z,t+Δt)/4 (Average)
- Σ_μ w_μ(z) q_μ(x,y,z,t+Δt) (Global Linear)
- Σ_μ w_μ(x,y,z) q_μ(x,y,z,t+Δt) (Linear per Gridpoint)

Hidden layers (only for CNN)

- 64*32*4 neurons per hidden layer
- Horizonal (X-direction) periodic boundary conditions
- Locally connected & nonlinear & weight sharing
- 4 nonlinear convolutional Layers
- 2 linear skip layers
- Hidden to output → no weight sharing

Training

- Training set: first 2400 days
- Input is scaled q → q^s per layer
- Trained per layer
- Loss = Mean squared error per layer MSE(q^s_{pred}(t+Δt) , q^s_{truth}(t+Δt))
- Optimizer:
  - linear algebra to solve w (linear models)
  - gradient descent , 200 iterations (CNN)

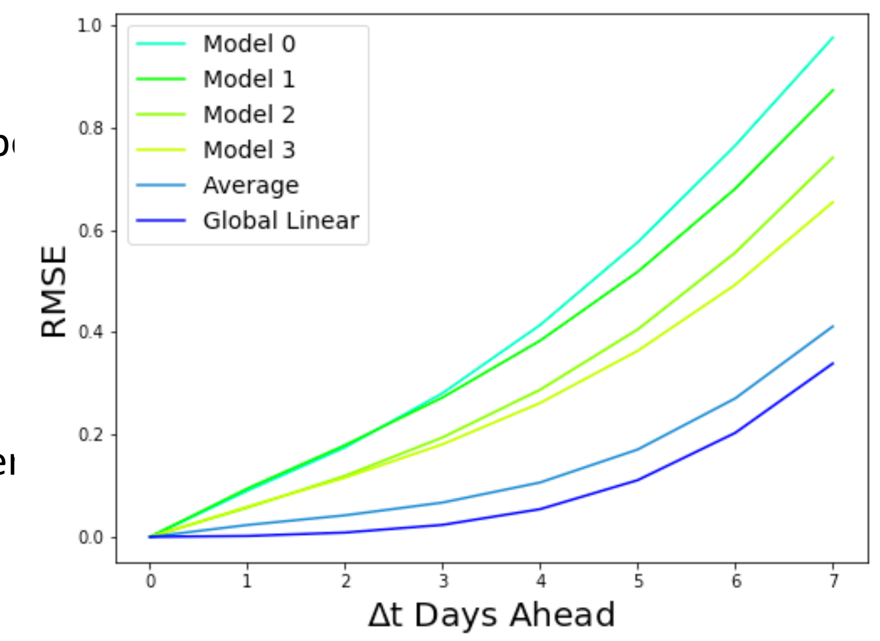
Test

- Test set: last 600 days
- Data scaled using scaler from training set
- Loss = RMSE (q^s_{pred}(t+Δt) , q^s_{truth} (t+Δt)) taking all layers into account

5. Experimental results

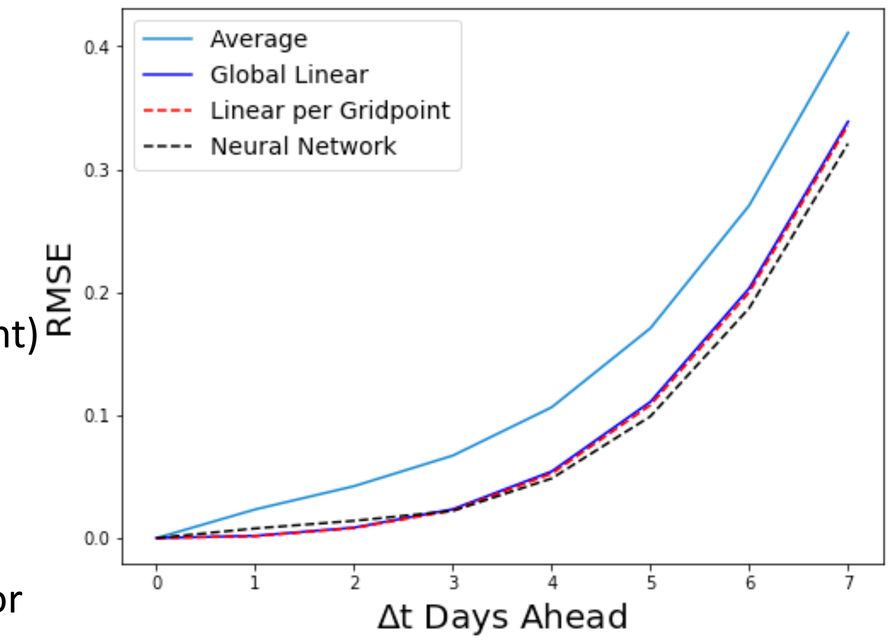
Figures show results on the **test set**.

Results in **top figure** confirm [3] that averaging improve upon individual imperfect model predictions and that global linear models (≅weighted average) improves even further



Results in **bottom figure** show that in this case, the linear per gridpoint and neural network improve only marginally (but statistically significant) upon weighted average per layer.

For Δt < 3, linear methods are superior to neural networks.



6. CONCLUSIONS

- **A neural network (CNN) with imperfect model predictions as inputs, *can* improve linear models in short term prediction.**
- **However, in this model simulation,**
  - **improvements upon linear models are marginal**
  - **linear methods are superior for *very short* term predictions**
- **So far only proof of concept on *artificial data of medium size atmospheric model.***

References

[1] van den Berge, L. A., et al. A multi-model ensemble method that combines imperfect models through learning, Earth Syst. Dynam., 2, 161–177. (2011).
[2] Wiegerinck, et al. (2013). On the limit of large couplings and weighted averaged dynamics. In *Consensus and synchronization in complex networks* (pp. 257-275). Springer, Berlin, Heidelberg.
[3] Schevenhoven FJ, Selten F. An efficient training scheme for supermodels. Earth System Dynamics. 2017;8(2):429-438
[4] Marshall, John, and Franco Molteni. "Toward a dynamical understanding of planetary-scale flow regimes." *Journal of the atmospheric sciences* 50.12 (1993): 1792-1818.

This work is partly funded by STERCP (ERC project 648982)