Advances in the application of deep neural networks for the retrieval of cloud properties for Sentinel-5 Precursor (S5P) / TROPOMI

Sentinel-5P Mission: 5 years Anniversary
14th October 2022

Fabian Romahn[1], Diego G. Loyola[1], Victor Molina Garcia[1], Ana del Águila[1], Ronny Lutz[1], Athina Argyrouli[2]
(1) German Aerospace Center (**DLR**), (2) Technical University of Munich (TUM)

DLR

S5P CLOUD Product

The operational S5P CLOUD product consists of the following main parameters:

1. cloud fraction (CF) → retrieved by OCRA

with cloud model CRB (clouds as reflective boundaries):

2. cloud height (CH)
3. cloud albedo (CA)

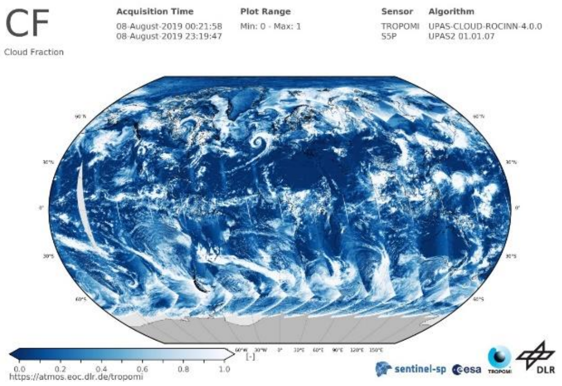
→ retrieved by ROCINN CRB

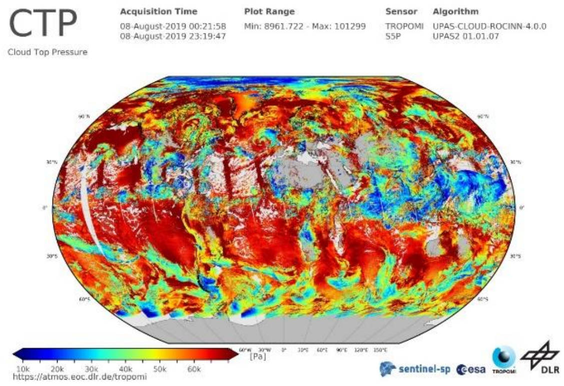
with cloud model **CAL** (clouds as layers):

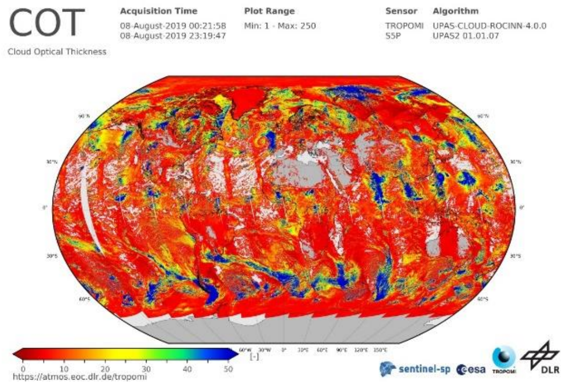
4. **cloud top height** (CTH)
5. **cloud optical thickness** (COT)

→ retrieved by **ROCINN CAL**

For the retrieval, **ROCINN** uses a **direct inversion** approach







Machine Learning in Remote Sensing

Why Machine Learning?

- Dramatically increased amount of data with latest generations of earth observation satellites
- Near real time requirements (NRT) for many products

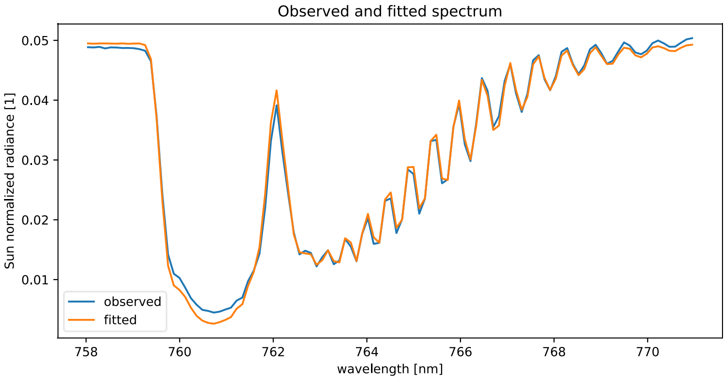
→ Retrieval algorithms have not only to be accurate but also to be very fast
→ Application of machine learning techniques to improve performance compared to classical algorithms

Machine Learning for Inversion Problems:

- Atmospheric retrieval problems can be formulated as inversion problems:

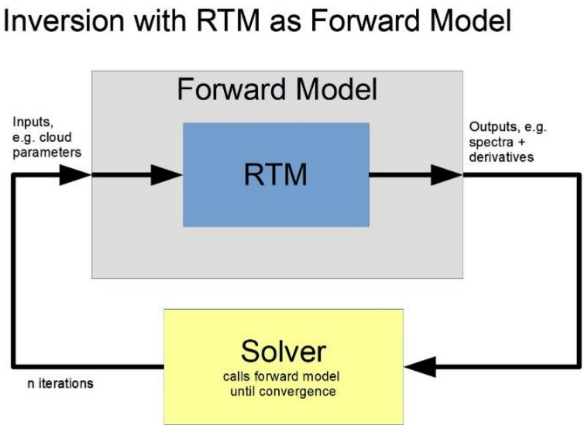
  Find parameters x that minimize residual $||F(x) - y||_2$ between a known vector y and the mapping of the parameters $F(x)$ - where F is a predefined function

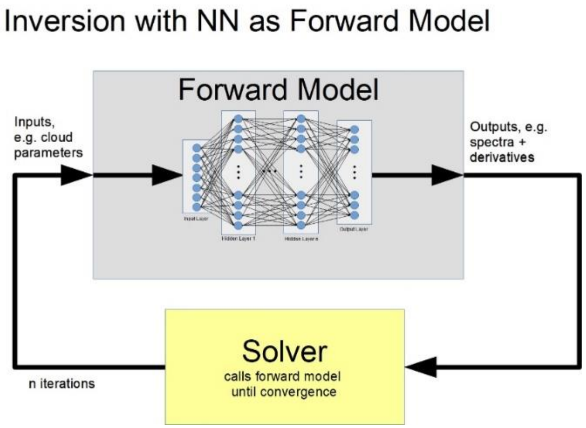
- In context of atmospheric retrieval:
  - x: State of atmosphere
  - y: Measured spectrum
  - F: Radiative transfer model (RTM)



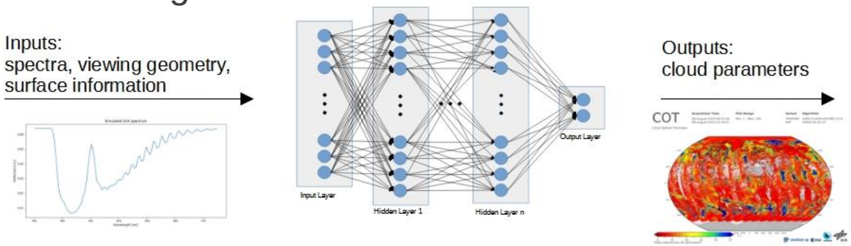
Two approaches for neural networks use

1. NN as **forward model** of a spectral fitting algorithm:
   - implements $F: X \rightarrow Y$, *state of atmosphere → spectrum*
   - substitutes and approximates the RTM
   - gradients (w.r.t to retrieval pamareters) usually need to be provided for solver
   - called in each iteration





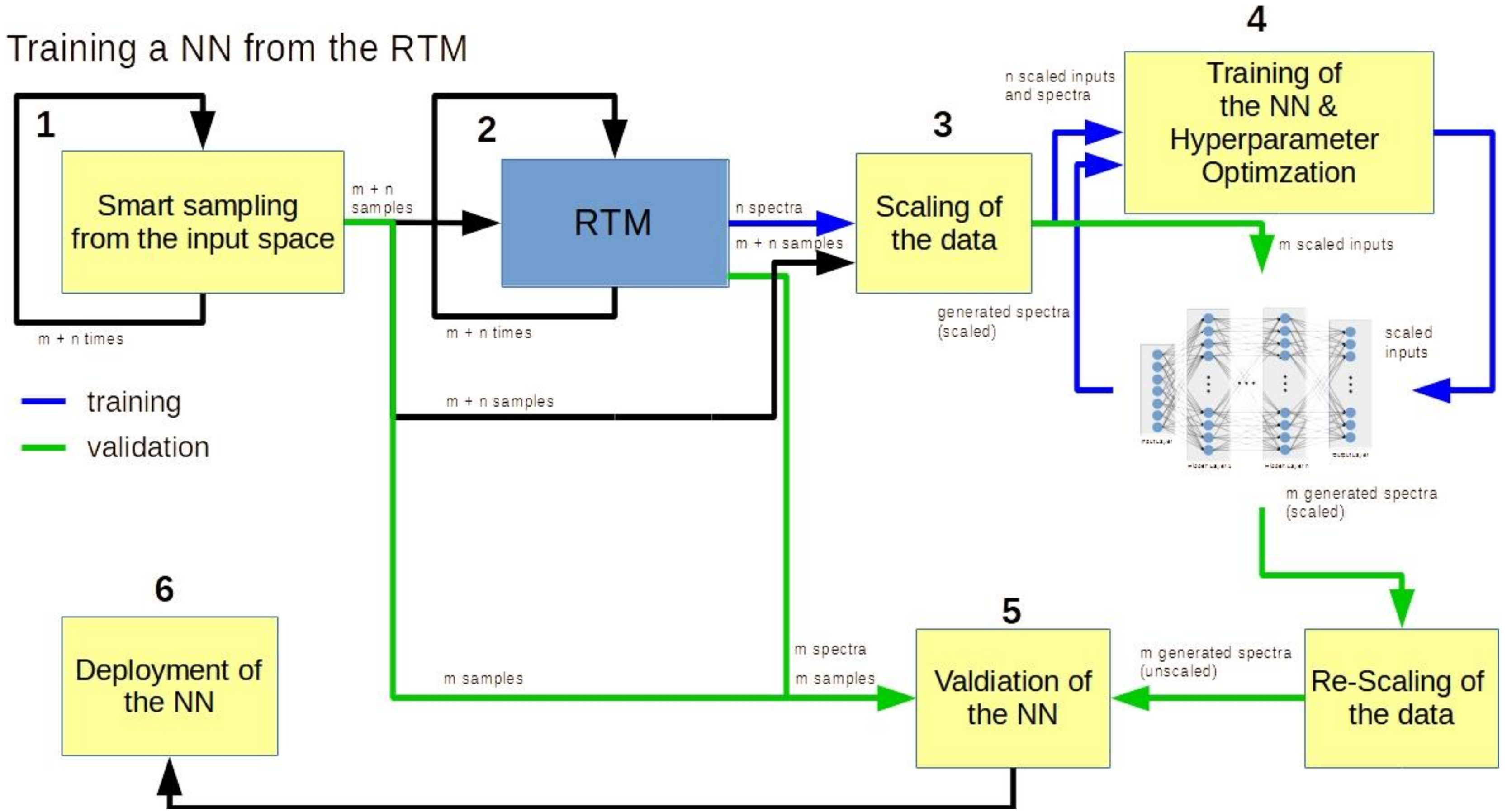
2. NN for **direct inversion**:
   - implements $F^{-1}: Y \rightarrow X$, *spectrum → state of atmosphere*
   - F^{-1} is generally unknown, can only be inferred through samples
   - No gradients needed after learnnig
   - called only once



1. NN as forward model - lifecycle chain

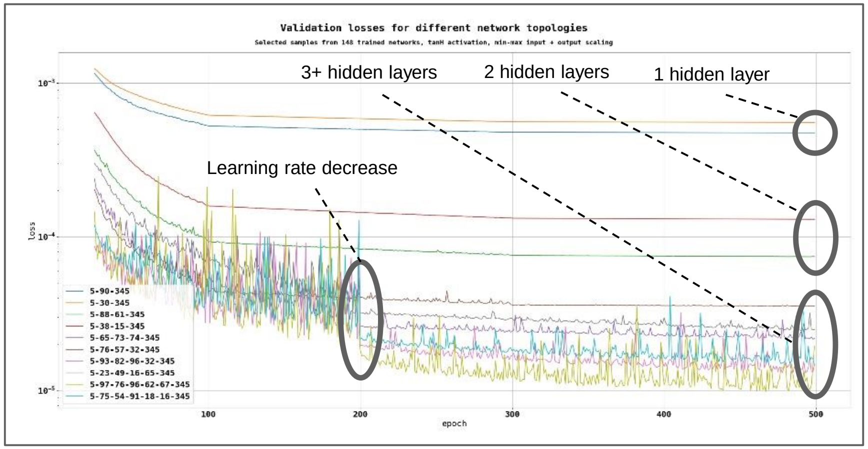
How to get from RTM to NN for algorithms in S5P, S4, …?

→ General procedure to replace RTM of an inversion algorithm by a NN:

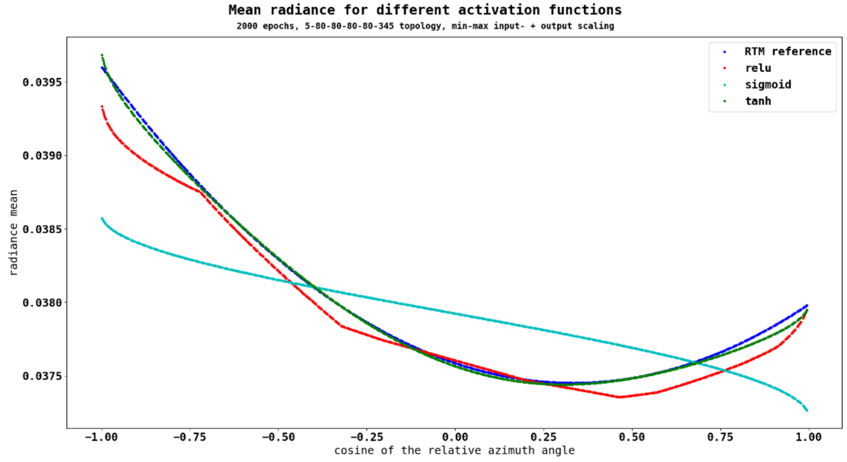


Evaluation of NNs

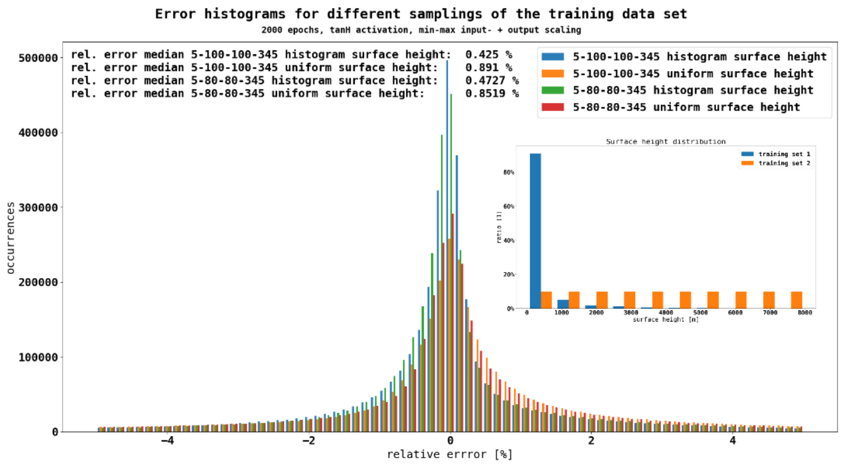
1. NN performances for different **topologies**



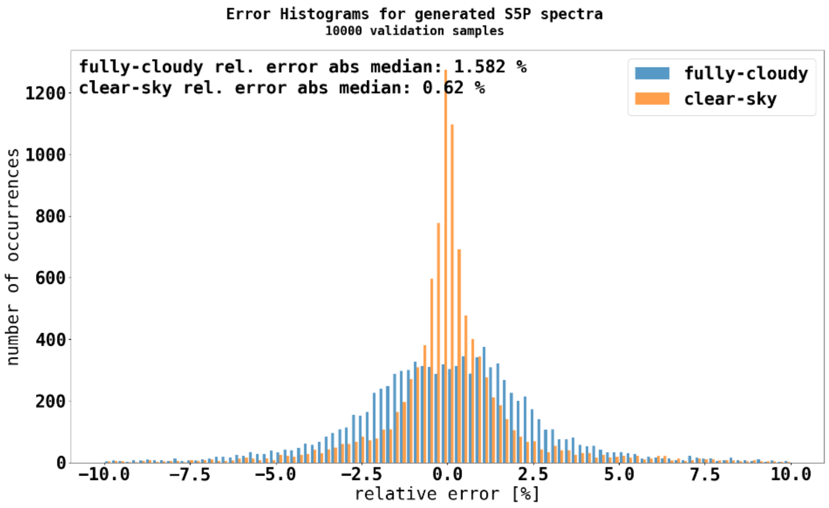
2. NN performances for different **activation functions**



3. NN performances for different **dataset samplings**



4. **S5P** NN performance - **clear-sky** and **fully-cloudy**

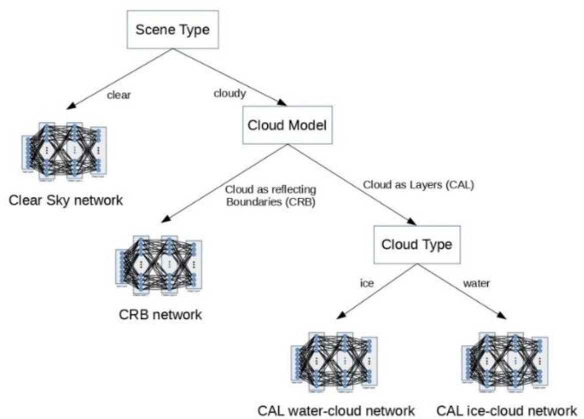


Performance and Application

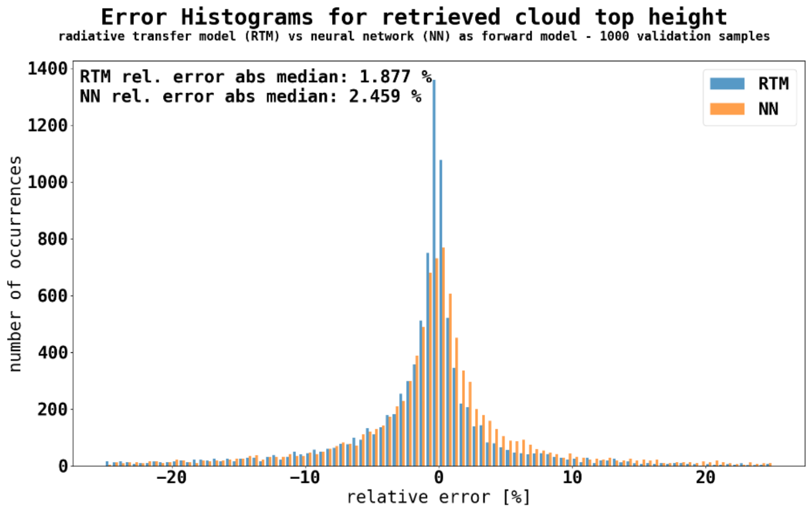
- With a proper configuration the NN can have sufficient accuracy to **replace the RTM as forward model** in the inversion algorithm

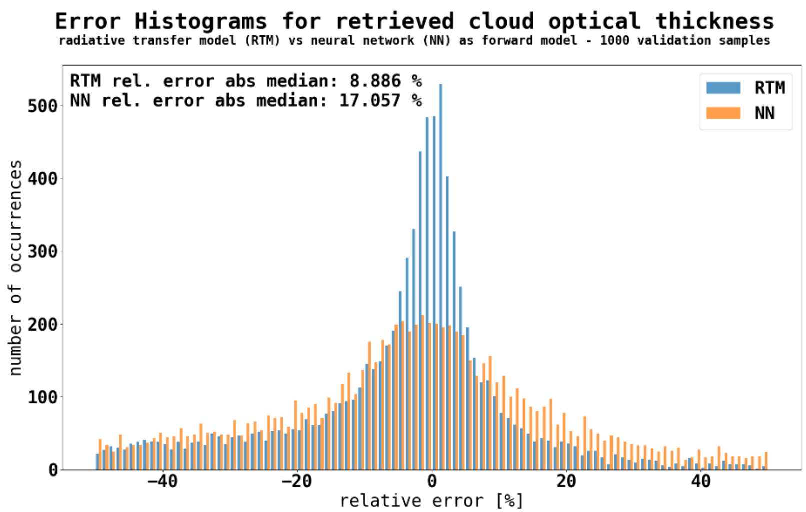
- The performance of the NN is **orders of magnitude** faster compared to the RTM: 250000 spectra: RTM: **17h**, NN **15s**
  → necessary for NRT retrieval
  → potential for improvements in the inversion algorithm (e.g. global optimization)

- NN lifecycle chain allows training and integration of different **specialized NNs** for different scenarios



Specialized NNs for different cloud scenarios used for S5P





Comparison of retrieval errors for cloud properties (CTH and COT) for RTM and NN as forward model

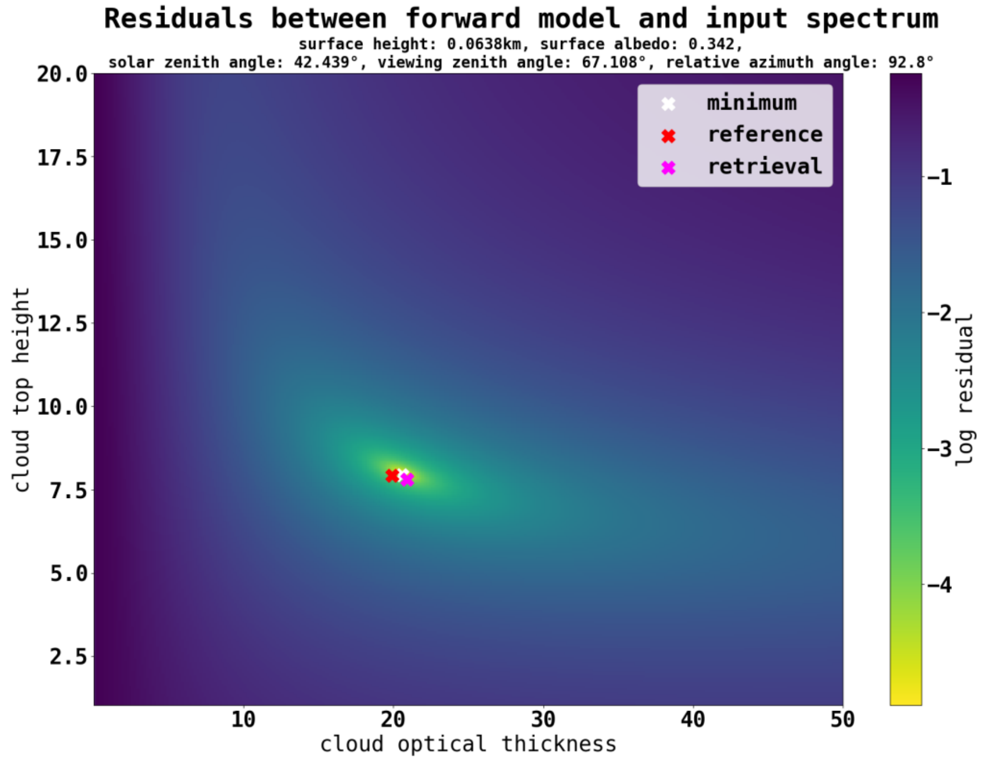
Spectral fitting challenges

- With a neural network as forward model, a spectral fitting algorithm can be used for the retrieval of the atmospheric parameters

- However, this is still challenging:
  - spectral fitting problem is generally ill-posed
    → **local minima**
  - real data contains noise in measurements

- → ROCINN algorithm (part of the operational S5P / current S4 CLOUD product) uses **Tikhonov Inversion**,
  which adds a regularization term to the optimization problem

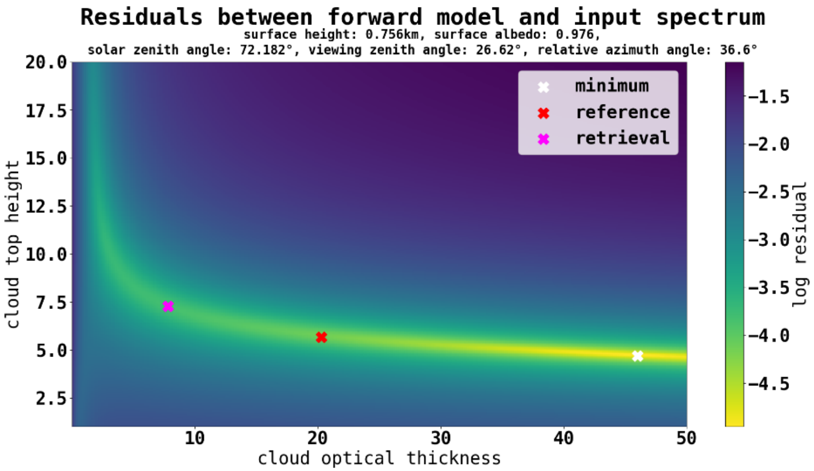
- For difficult cases, good **a-priori** values for the retrieval parameters are still important



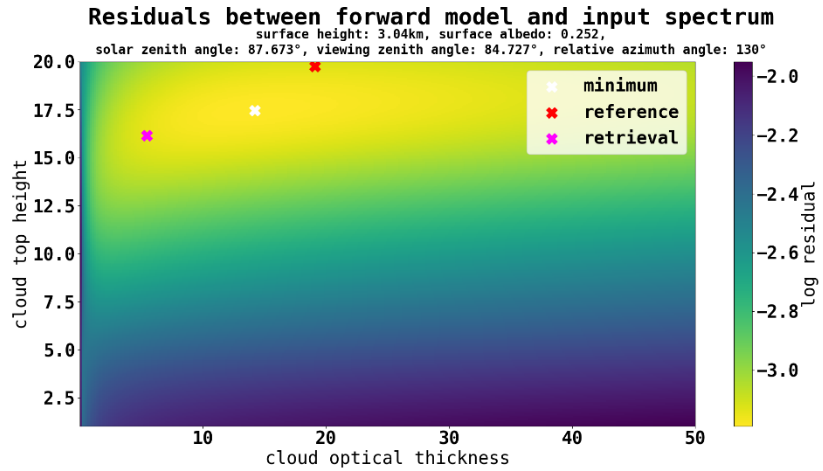
Residual map for an „easy“ problem with a well defined global minimum

Spectral fitting challenges - examples

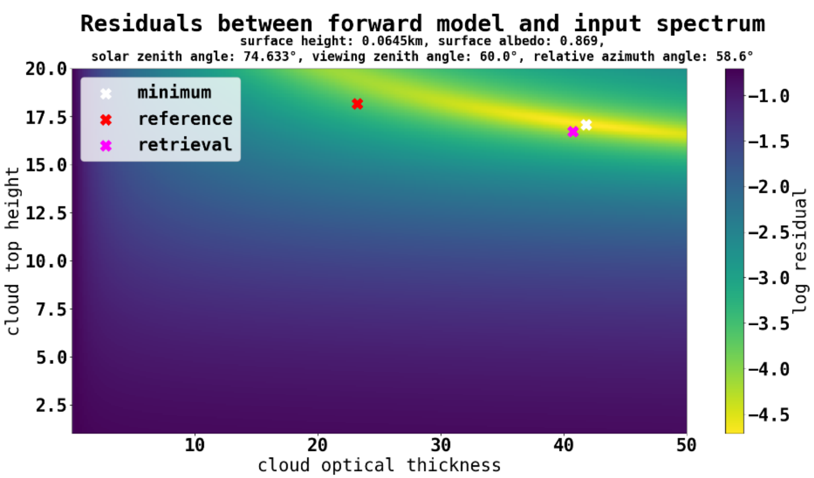
‘stretched‘ minimum:



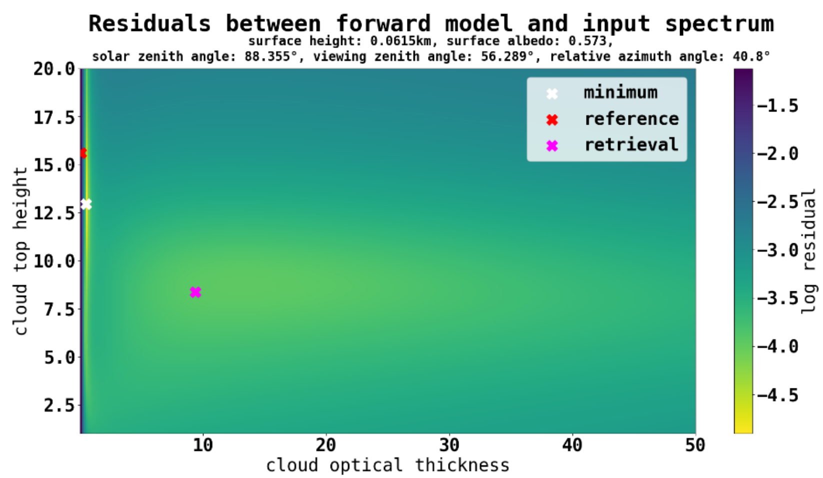
diffuse minimum:



forward model error:



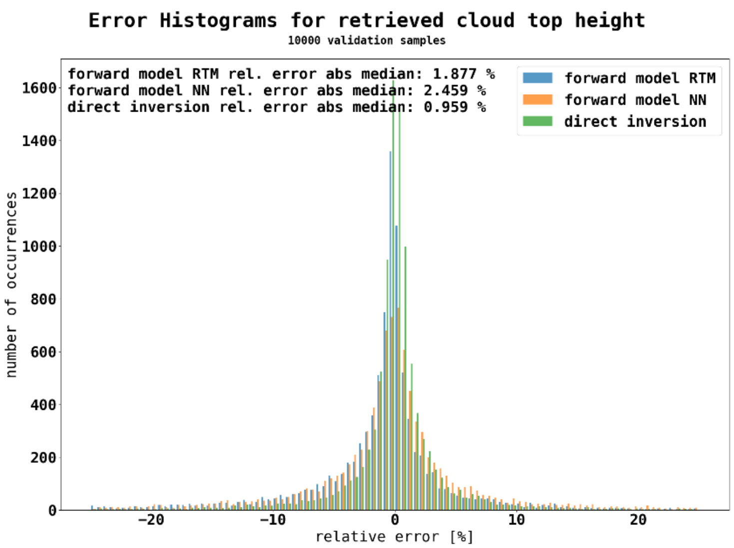
local minima

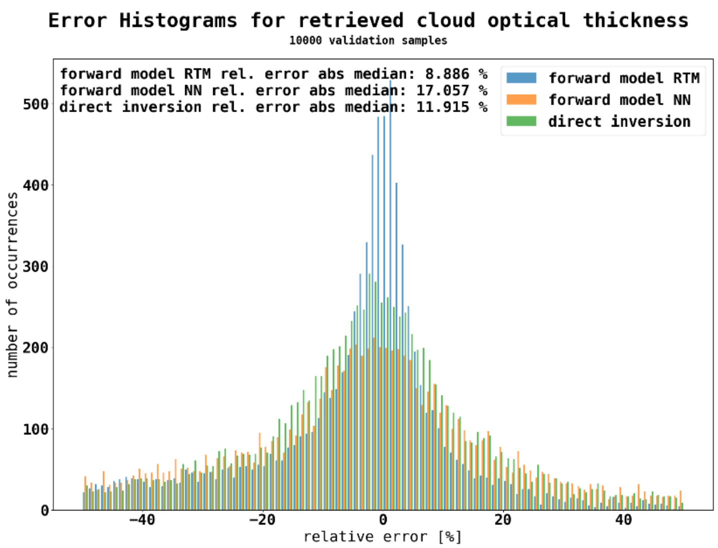


2. NN for Direct Inversion

- NN for direct inversion can avoid some of the issues of the spectral fitting:
  - no **fine adjustment** of the retrieval algorithm (e.g. regularization parameter, tolerances for convergence, etc.), all settings via the hyperparameters and training of the network
  - no **a-priori** necessary
  - only **one call** (iteration) per problem

- Input: spectra, viewing geometry, surface parameters, Output: cloud parameters
  (topologies: NN as FM: 7-66-77-26-89-78-94-99-107, NN for direct inversion: 112-80-80-80-80-2)
- Evaluation on validation dataset:





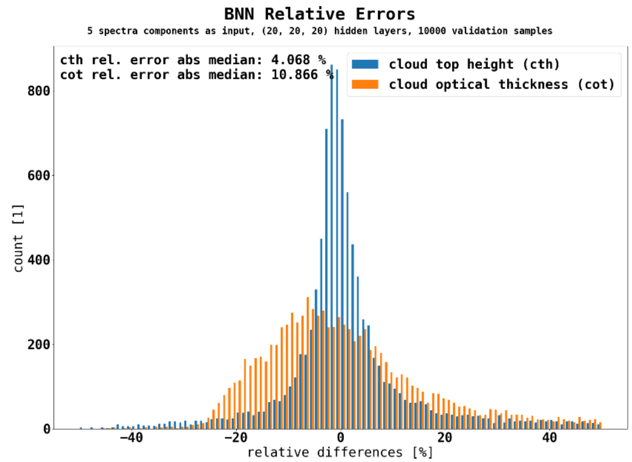
→ **Best results** for cloud top height: 0.96% vs. 2.46% (NN as FM), 1.88% (RTM as FM)
→ **Improved results** for cloud optical thickness: 11.92 % vs 17.06% (NN as FM), 8.89% (RTM as FM)

Bayesian Neural Networks

- Drawback: No indication for the quality of the results for the direct inversion NN („*blackbox*“)
- In contrast to the spectral fitting with e.g. iterations, convergence, residual, etc.

→ **Bayesian neural networks (BNN)**:

- learns uncertainties in model parameters
- output is a probability distribution
- more complex and are harder to train
  → use of autoencoders to reduce complexity

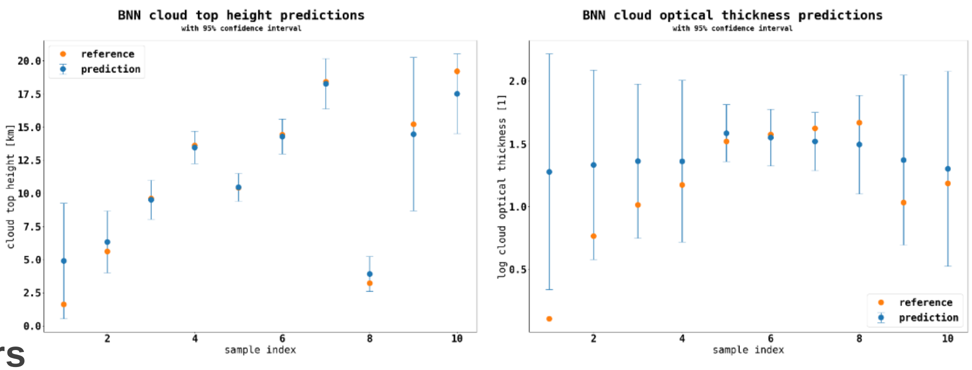


BNN relative retrieval errors for CTH and COT from validation data set

Evaluation:

1. Overall, BNN performs slightly worse than the conventional NN (taking the means as output)
   - learning is harder (much slower), current results likely not optimal
   - for many deep topologies (> 3 hidden layers) learning is not successful

2. Standard deviation of ouptuts allows definition of a confidence interval
   - reference values are mostly inside
     → **reliable quantification of errors**



Retrieved CTH (left) and COT (right) values for 10 random samples

Conclusions and Outlook

1. **NN as forward models:**
   - can improve speed of existing retrieval algorithms by orders of magnitude through substitution of the radiative transfer model → *near real time* applicable
   - NN lifecycle chain offers training and integration of specialized NNs
   - many properties from classical retrieval algorithms are inherited:
     - retrieval diagnostics
     - difficulties with ill posed problems, local minima
   - performance allows for potential in inversion algorithm improvements

2. **NN for direct inversion:**
   - easy to apply, good initial performance, no a-priori needed
   - conventional NNs are „black boxes“, no error quantification
   - BNNs as a possibility to overcome this:
     - provide error quantifications
     - more complex and harder to train

→ NNs for direct inversion, especially BNNs with error quantification, have great potential for retrieving cloud properties for S4 / S5P as an alternative to the current forward model approach

- Further investigations in hyperparameter selection and learning have to be made
- Invertible neural networks (INN), that learn forwards and backwards and can also provide distributions are another interesting approach that should be followed

For further questions, please contact me: Fabian.Romahn@dlr.de