Current Status of CML Data Assimilation (P3)

K. Vobig

RealPEP

Deutscher Wetterdienst
Wetter und Klima aus einer Hand

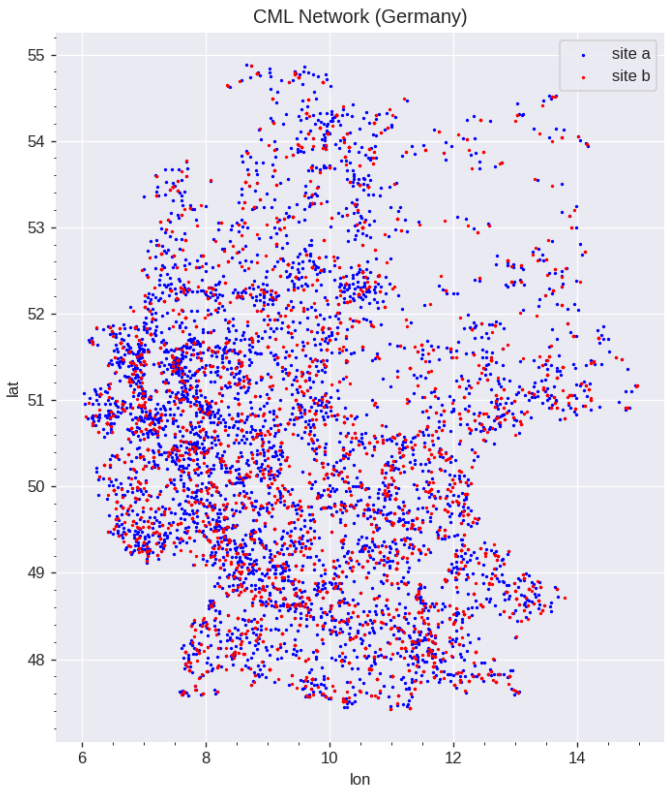
DWD

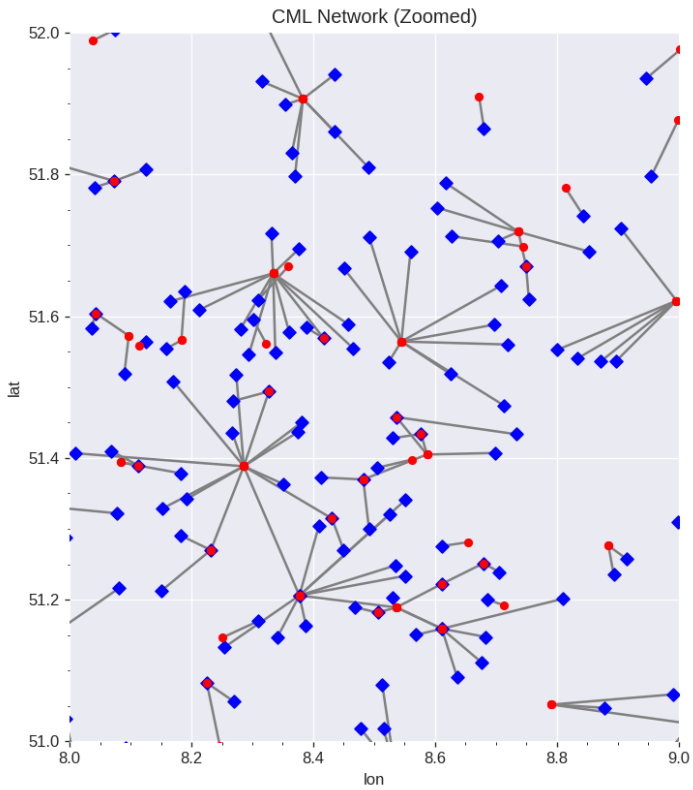
Introduction

- CML data already successfully employed for the estimation of rain rates (QPE) → P1
- overall objective here (P3): **data assimilation** of CML data in **numerical weather prediction models** for **improving QPF**
  - able to contribute to bridging the gap between QPN and NWP?
  - (How much) does it improve QPF?
  - How does it compare to Radar data assimilation?
- in the following: discussion of **technical details** of CML data assimilation and results of **case study**

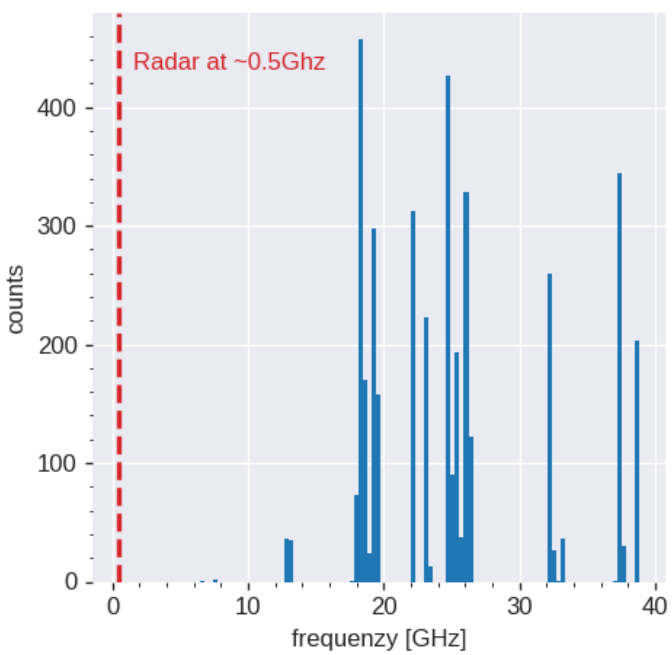
CML Network Overview I

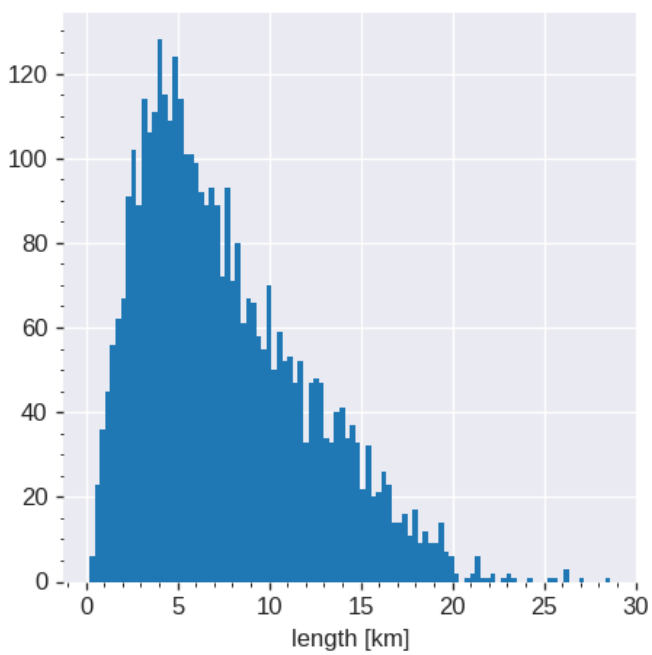
- each link consists of a sender and receiver (blue/red)
- ~4000 CMLs in current dataset for June 2019
- temporal resolution 1min

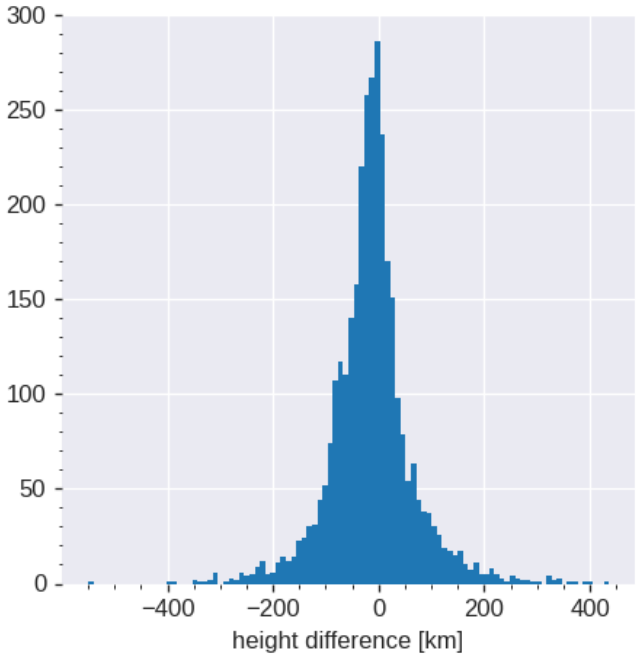




CML Network Overview II





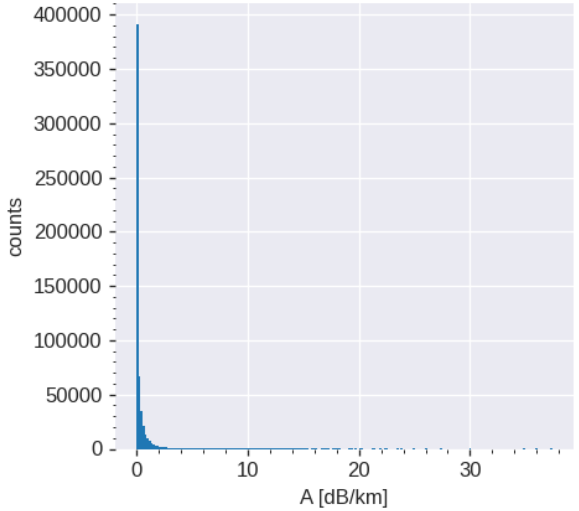


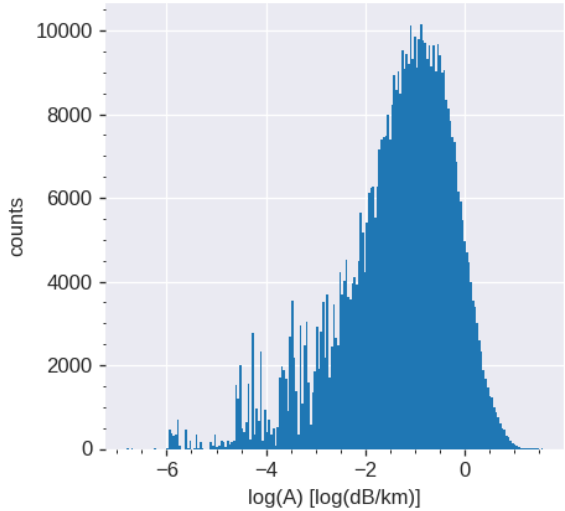
- CML frequency significantly above DWD Radar frequency
  → different physics involved!

CML Path-Integrated Attenuation

- use **path-integrated specific attenuation** for assimilation
  - referred to as A from now on
  - A [dB/km] = attenuation [dB] / distance [km]
  - direct relationship of A with rain rate (via power law)

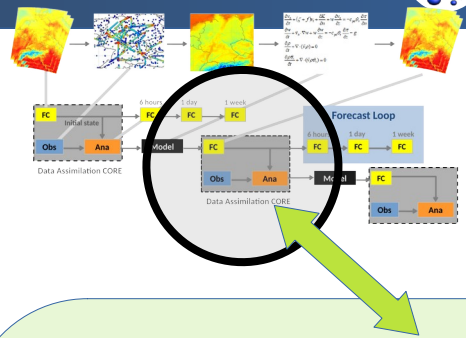
- most attenuations very small
- distribution “normal” in log space
- **outlook**: use f(A) for assim.

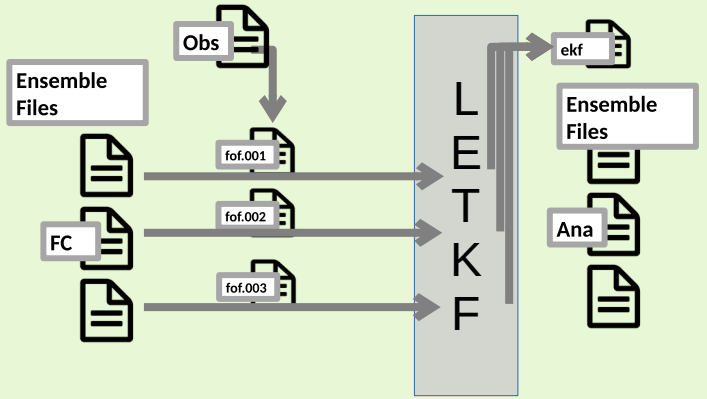




LETKF Assimilation System

- for assimilating observation data, **feedback/fof files** have to be generated
- each fof file (one for **each ensemble member**) contains all data relevant to the LETKF assimilation process for a specific assim. date
- particularly, for each observation there has to be a simulated **model equivalent**!

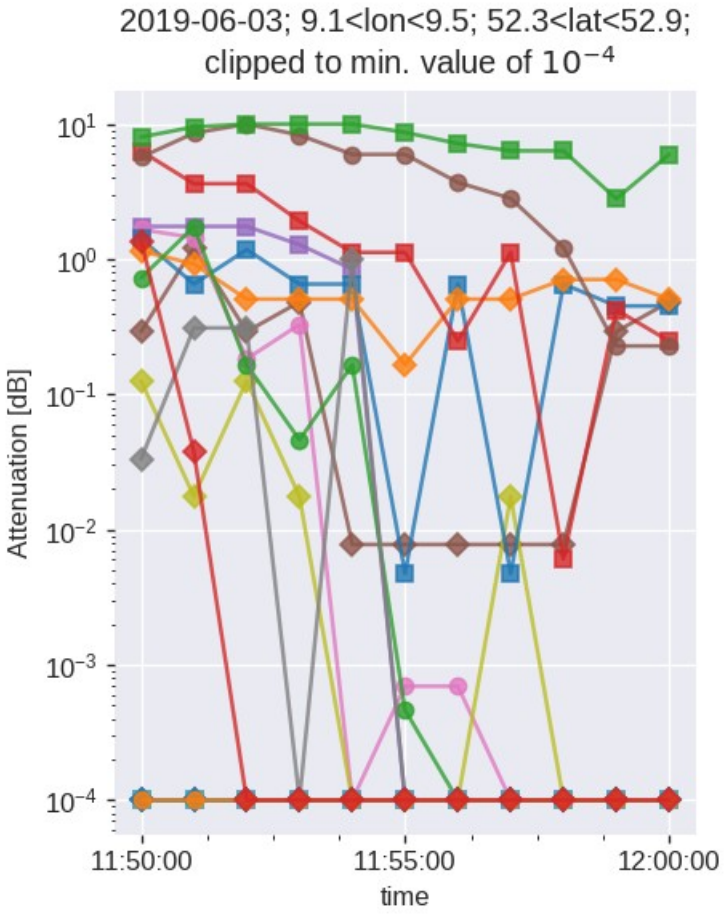




- fof.*: sim. + obs. quantities of ens. members
- LETKF produces increments depending on innovations + Kalman gain

Processing Observed Attenuations

- perform **temporal superobbing/smoothing**:
  - for getting observations relevant for an assimilation at t_0 calc. mean of A(cml, t) over a 10 min time window [t_0 - 10min, t_0]
  - **smooths** out **erratic fluctuations** of attenuations
- **outlook**: also perform spatial thinning and/or superobbing

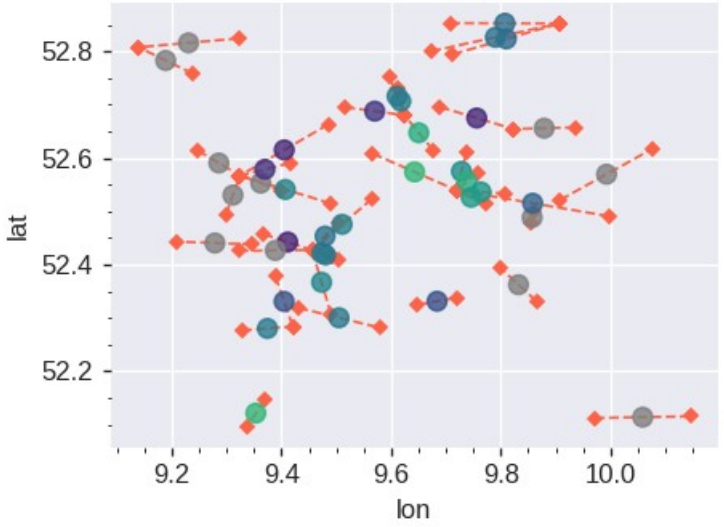


Simulation via EMVORADO

- using the **Radar forward operator EMVORADO** in offline mode for **simulating attenuations**, i.e., calculating relevant model equivalents
- two **main inputs** here (besides many other config. options):
  - ICON model fields (on regular grid) QV, QR, ...
  - auto-generated namelist containing all static information for each CML that should be simulated (equivalent to Radar stations):
    frequency, lat/lon/amsl of “station”, (single) azimuth/elevation of ray, ...
- extract **path-integrated one-way attenuation** from output
- perform EMVORADO run for **each ensemble member** (→model fields)!
- current **limitations**:
  - single EMVORADO run not able to simulate all (~4000) CMLs
  - simulation does not include water vapor attenuation

CML Feedback Files

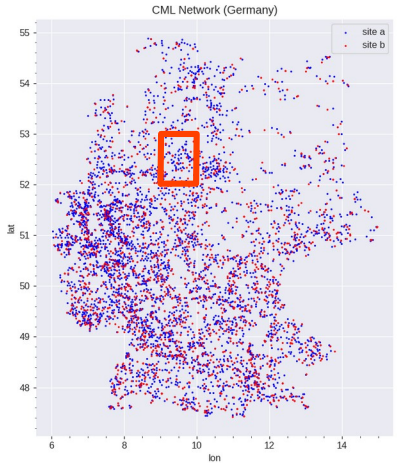
- **collect** processed **observed** and **simulated** data for specific **assim. date**
- use **halfway** lat/lon/level of each CML in feedback files
- CML data currently assimilated as SYNOP observation (obstype) and using an experimental codetype and varno
- use **relative observation error** of 20%
- write all data into feedback (netcdf) file



obstype	(d_hdr)	float32	1.0 1.0 1.0 1.0 ... 1.0 1.0 1.0 1.0
codetype	(d_hdr)	float32	991.0 991.0 991.0 ... 991.0 991.0
statid	(d_hdr)	|S10	b'1085' b'1208' ... b'1716' b'1727'
lat	(d_hdr)	float32	52.82 52.78 52.44 ... 52.51 52.37
lon	(d_hdr)	float32	9.23 9.189 9.278 ... 9.858 9.473
time	(d_hdr)	float32	60.0 60.0 60.0 ... 60.0 60.0 60.0
varno	(d_body)	float32	991.0 991.0 991.0 ... 991.0 991.0
obs	(d_body)	float32	0.0 0.0 0.0 ... 0.09817 0.0002519
level	(d_body)	float32	60.52 61.24 103.1 ... 99.79 110.8
state	(d_body)	float32	1.0 1.0 1.0 1.0 ... 1.0 1.0 1.0 1.0
e_o	(d_body)	float32	0.0 0.0 0.0 ... 0.01963 5.038e-05
veri_data	(d_veri, d_body)	float32	0.0 0.0 ... 0.01968 0.07365

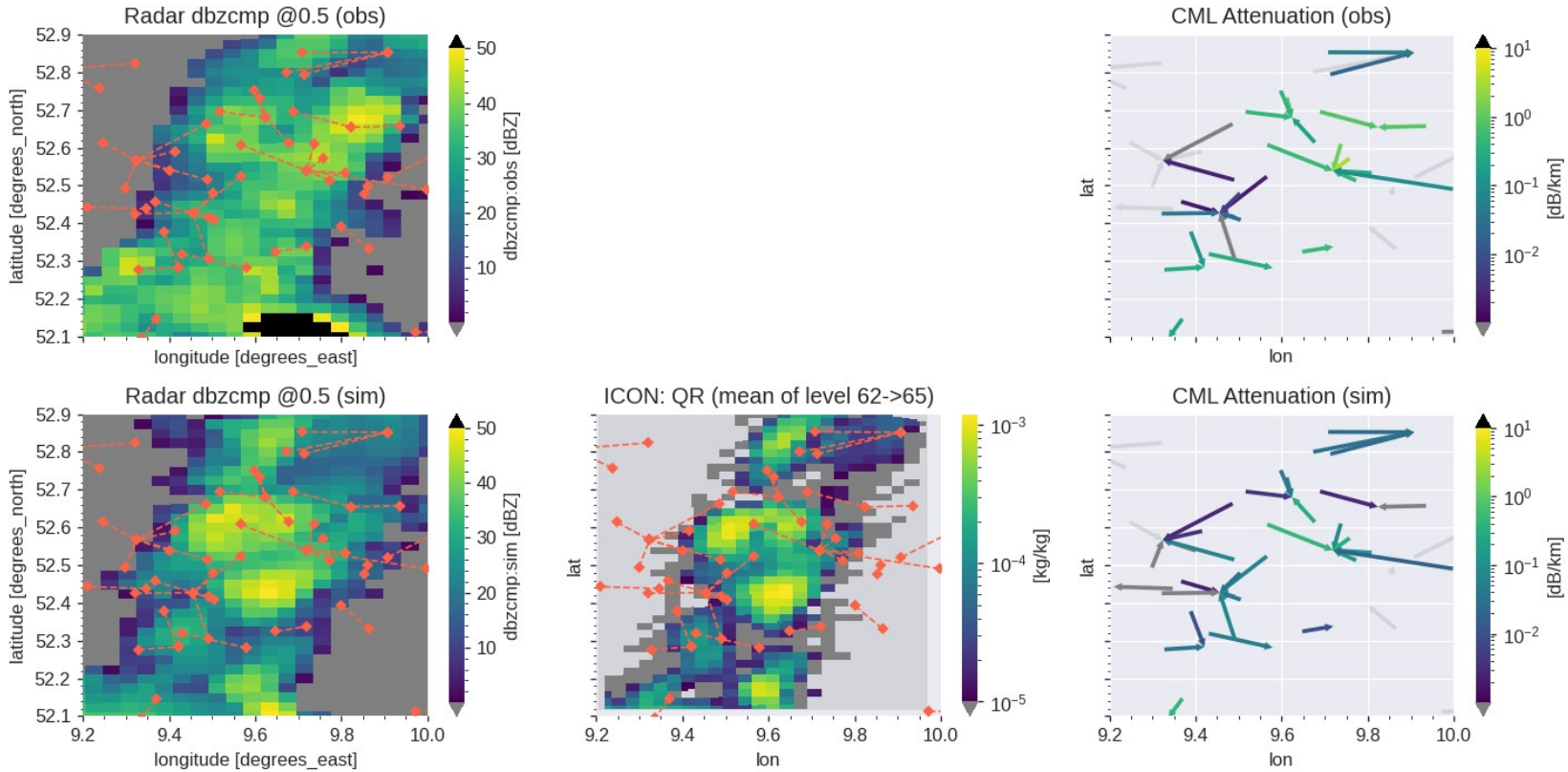
BACY Setup & Case Study Details

- perform assimilation on 2019-06-03 12:00:00
- only use CMLs within region 9.2°<lon<10°, 52.1°<lat<52.9°
  - evades EMVORADO limitation
  - 40 CMLs within this region



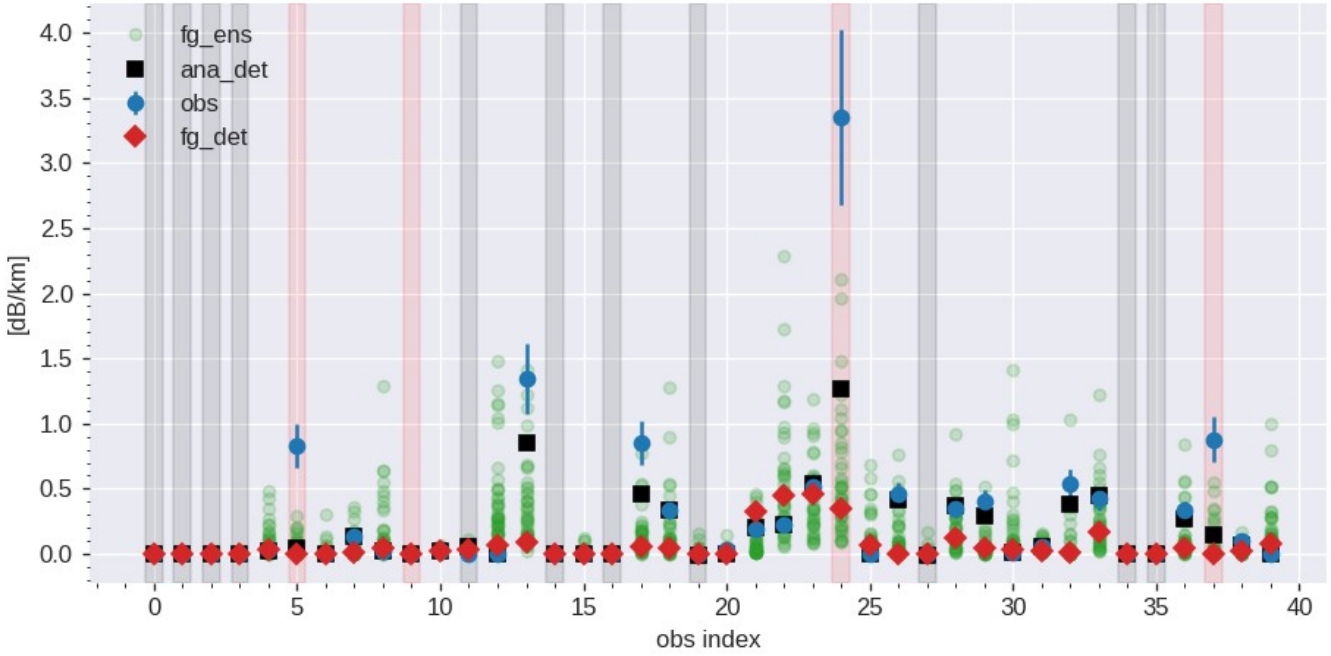
- **automated system** for the **construction** of CML **feedback files**
  - includes all necessary data processing steps
  - implemented (mostly) in Python
- integrated into new BACY experiment
  - **only CML** data is set to **active** here!

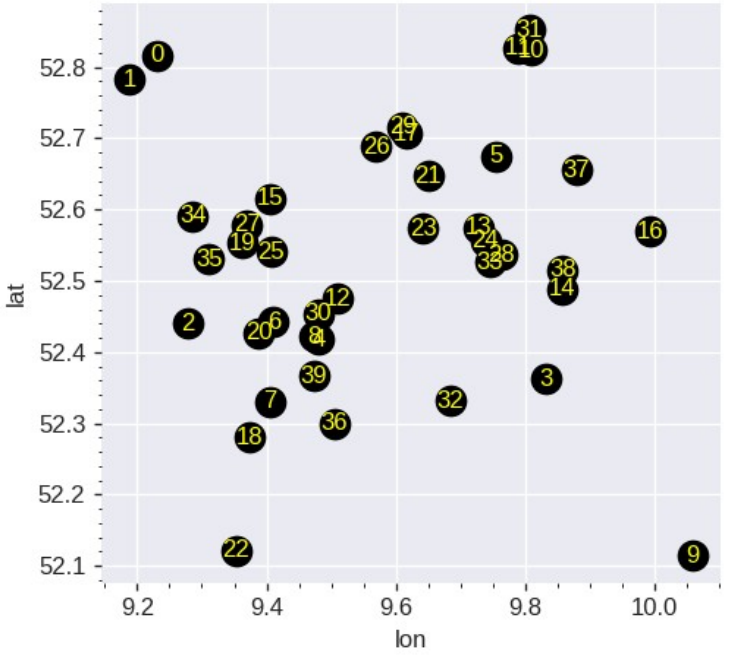
Overview: Pre-Assimilation



consistency/plausibility check based on Radar reflectivities, ICON model fields for QR, and CML attenuations: **passed**

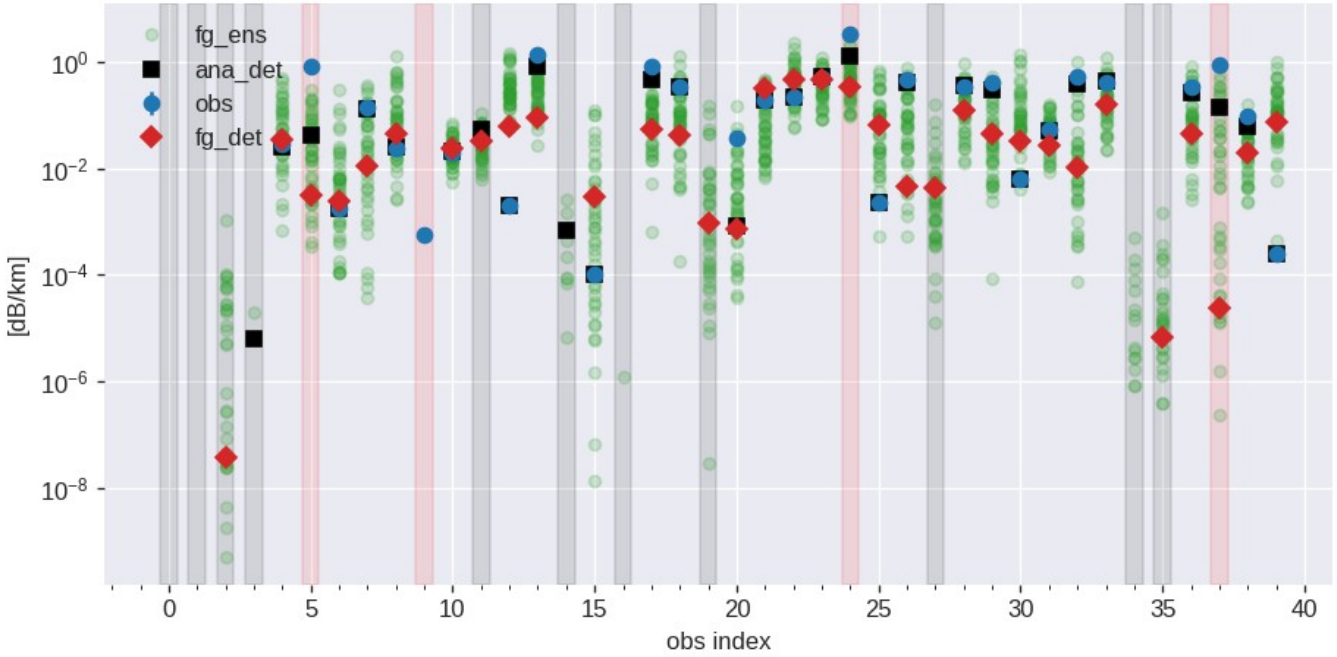
CML Assimilation Result

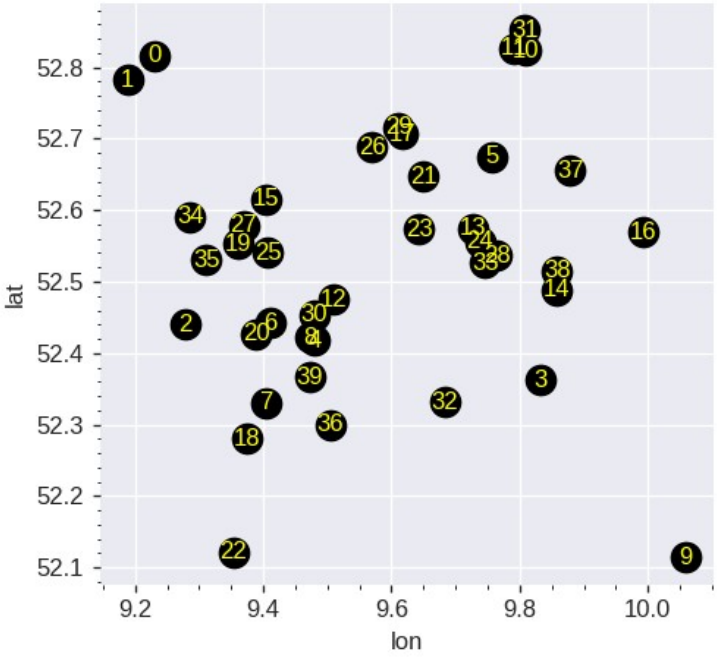




- representation of corresponding “ekf” file (LETKF output)
- shaded background → special assimilation state
  - passive (gray), rejected (red)

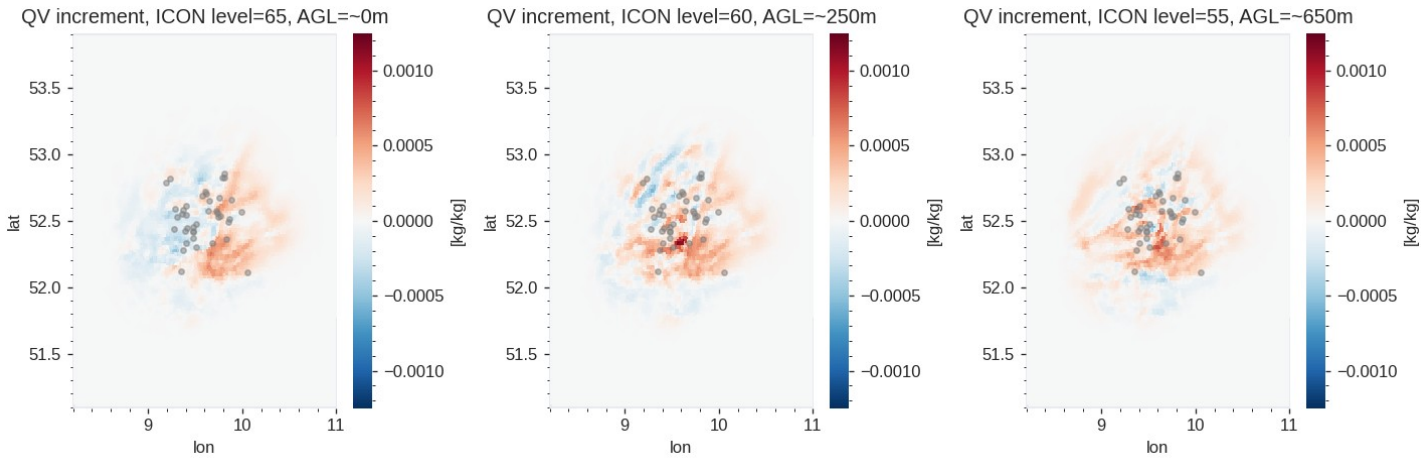
CML Assimilation Result (Log Scale)





- rescaled y-axis using logarithmic scale
- outlook: assimilation of log(A) might be interesting!

ICON Increments for QV



- LETKF increments for ICON model fields (for QV)
- main “ingredients” for LETKF increments: innovations, correlations/variances, localization
- **outlook**: further study of correlations $\rho(q_i(\mathbf{x}), A(\mathbf{x}))$
  - straightforward in single-obs. experiments!

Summary & Outlook

- first version for assimilating CML data (integrated into BACY)
- first assimilation results seem plausible
- possibility for many interesting studies now!

- evaluate/verify results of “CONV” vs “CONV+CML” BACY cycles
- study impact of several parameters: obs. error, localization, ...
- study transformation of data before assimilation (e.g. logarithmic)
- general quality control, spatial thinning/superobbing, bias correction (more data/statistics needed)
- EMVORADO developments

Thank you for your attention!