A 4D-Var inversion system based on the icosahedral grid model (NICAM-TM 4D-Var)

MAARTEN KROL (BASED ON WORK OF OTHERS)

Why this presentation?

* Based on two recent GMDD papers of Yosuke Niwa (Tsukuba, Japan)
* Links to new developments (IFS/4DVAR; e.g. ICON)
* Also contains interesting way to use time/space correlations within 4DVAR
* This latter might be useful for multi-tracer inversions

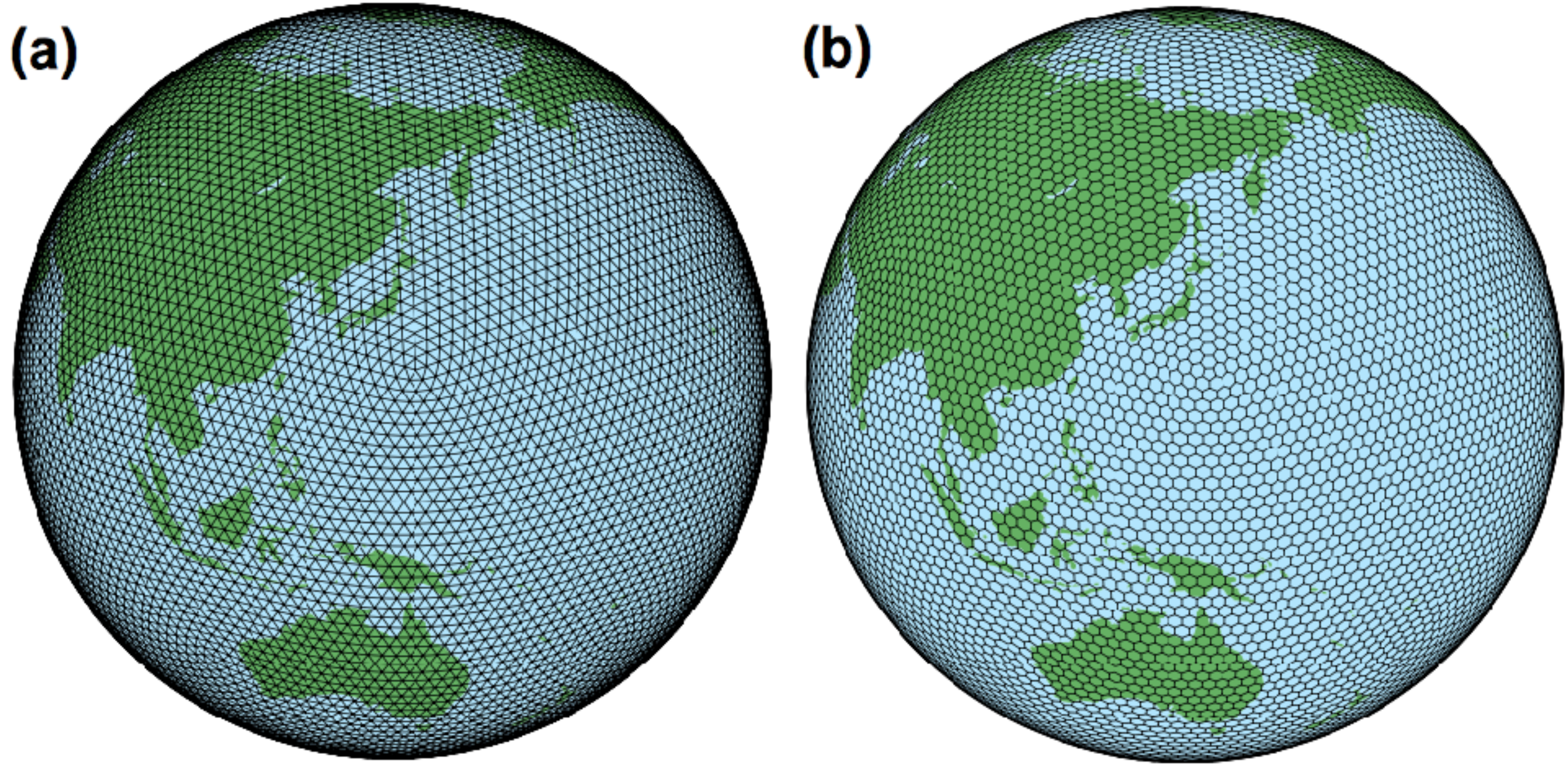


Figure 1. The grid distribution of NICAM glevel-5. Triangular elements produced from an icosahedron by five-times division (a) and control volumes constructed by connecting the mass centers of the triangular elements (b).

Excellent performance

- All the simulations are performed on PRIMEHPC FX100 with MPI parallelisation on 320 cores
- For the one-year-long simulation, the off-line forward model requires only 7 min., while the on-line model requires about 70 min.
- The corresponding (off-line) adjoint calculation also requires 7 minutes.
- Different “coupling” resolutions are presented (6-3-1 hour)

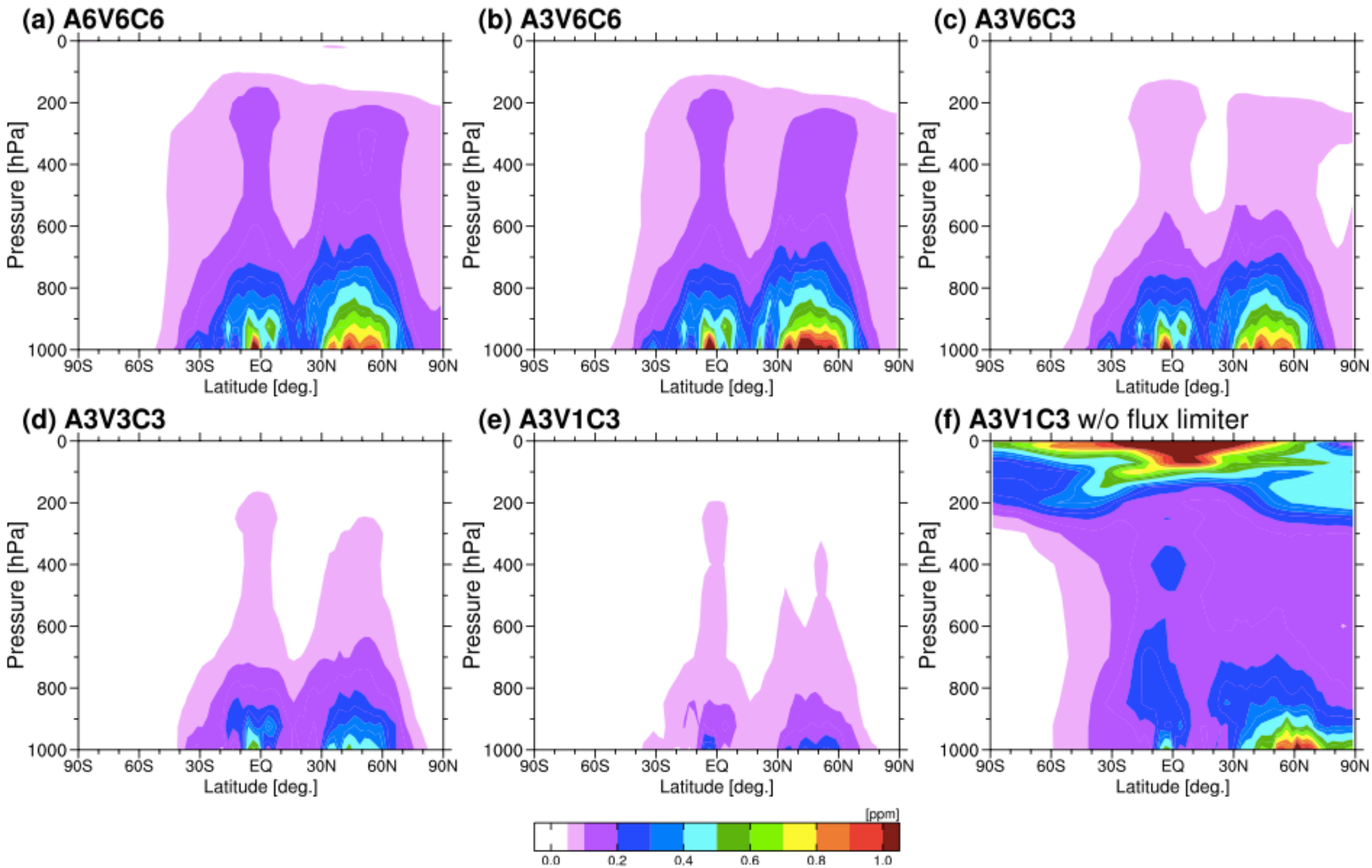


Figure 2. Zonal-mean latitude-pressure cross-section of root-mean-square difference (RMSD) of CO_2 concentration simulated by the off-line model against the on-line model. Time interval of the meteorological driver data is changed for each transport process as shown in Table 2 (a-e) and the same time resolutions are used as (e) but with the flux limiter switched off (f).

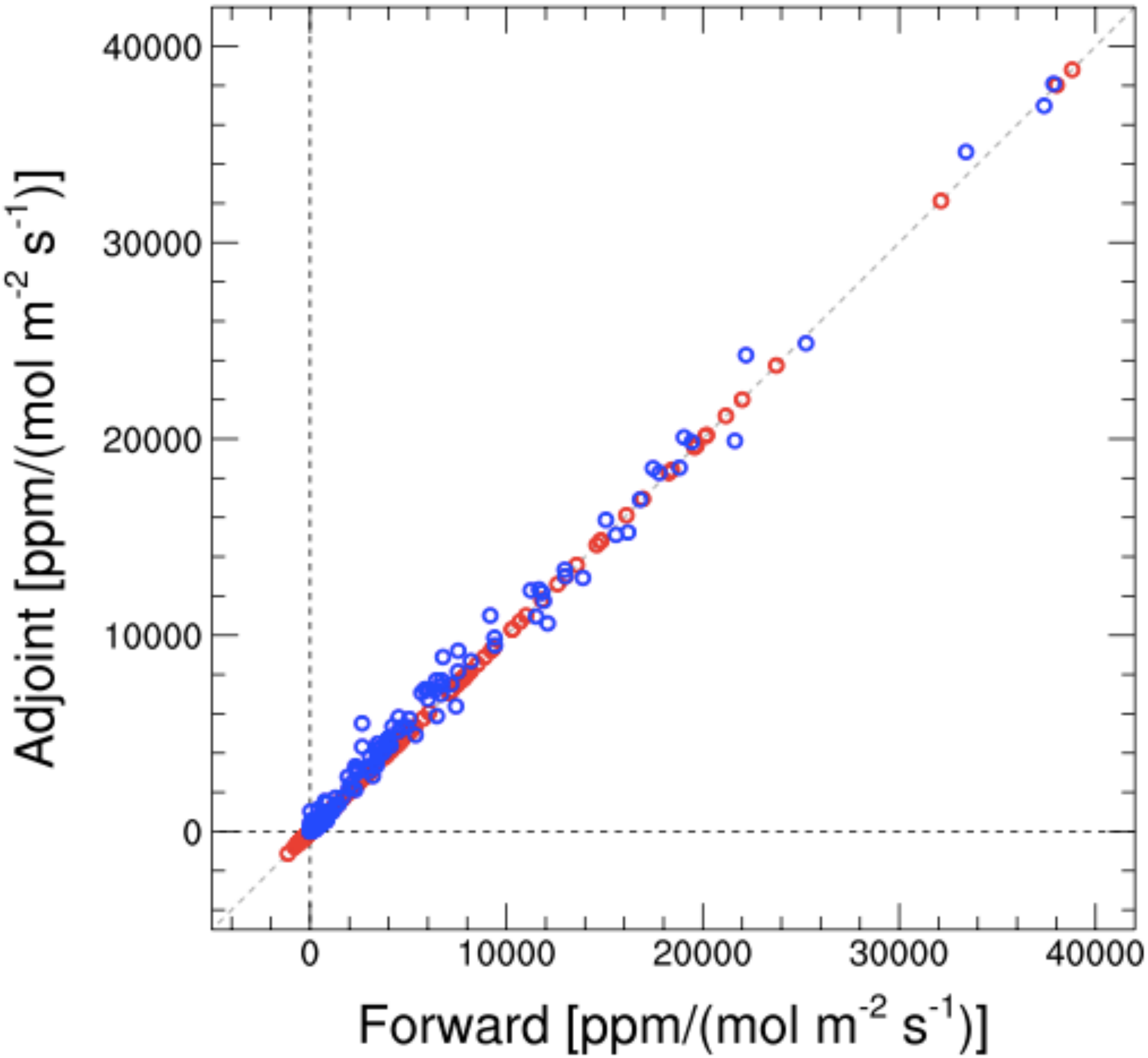
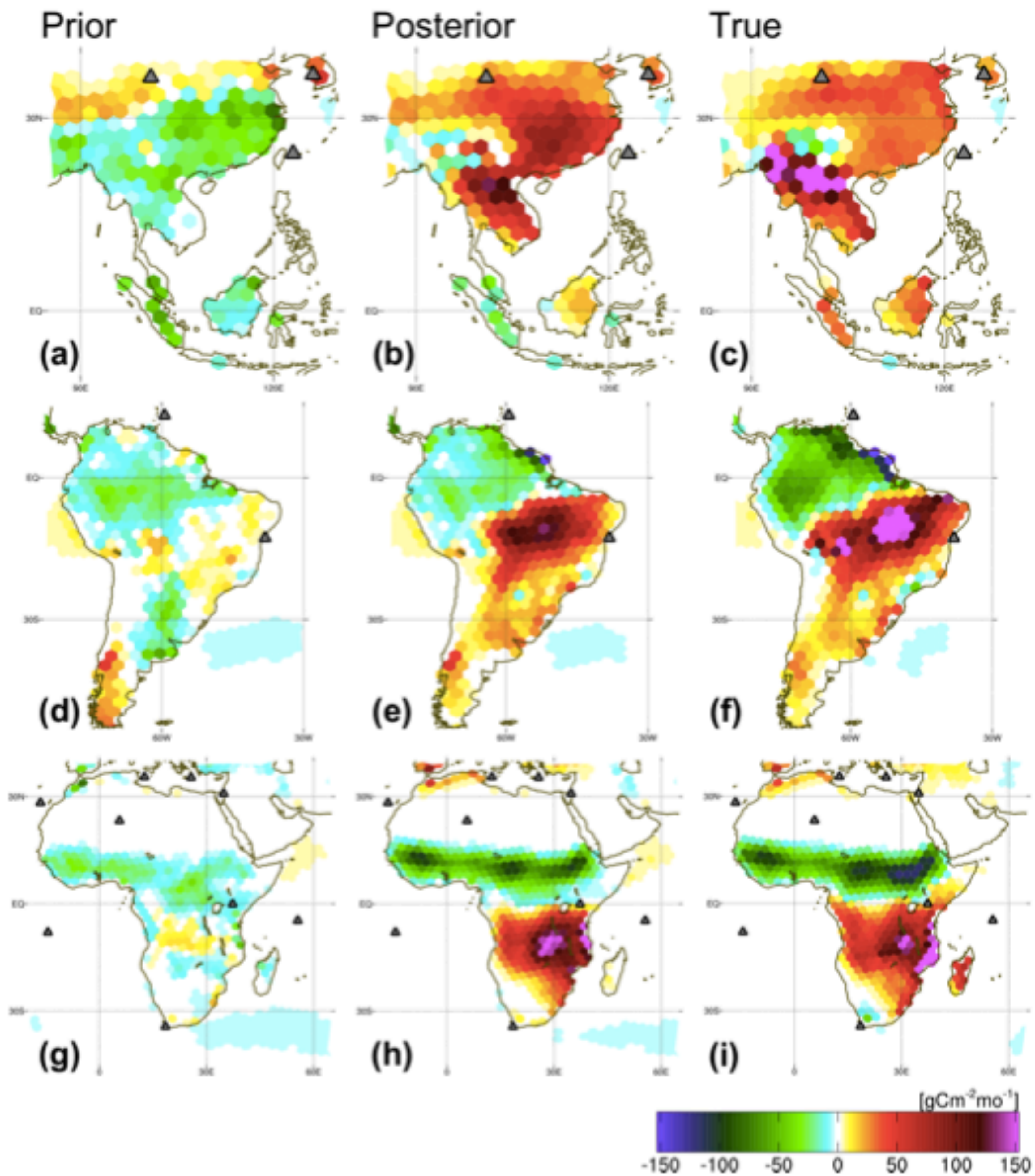


Figure 6. The scatter diagram showing 160 concentration values simulated by the forward model at the observation points versus their corresponding adjoint footprint values at the flux point X. The red open circles denote values from the linear model setup (the forward model without the flux limiter and the discrete adjoint model), while the blue open circles denote values from the nonlinear model setup (the forward model with the flux limiter and the continuous adjoint model).


Prior
Posterior
True
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)
[gCm-2mo-1]
-150
-100
-50
0
50
100
150

POpULar optimisation

$$J(\mathbf{x}) = \frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{B}^{-1}\mathbf{x} + \frac{1}{2}\left(\mathbf{Mx} - \mathbf{y}^{\mathrm{dobs}}\right)^{\mathrm{T}}\mathbf{R}^{-1}\left(\mathbf{Mx} - \mathbf{y}^{\mathrm{dobs}}\right)$$

$$\mathbf{g} = \mathbf{B}^{-1}\mathbf{x} + \mathbf{M}^{\mathrm{T}}\mathbf{R}^{-1}\left(\mathbf{Mx} - \mathbf{y}^{\mathrm{dobs}}\right).$$

B = covariance matrix of the prior fluxes, e.g. space and time correlations in the prior fluxes: often too large to store and invert

Efficient pre-conditioning: x' = $B^{-1/2}$x

E.g. Meirink, 2008: clever decomposition of B (in our *pyshell*)

Going to multi tracer (e.g. CO-CO_2) provides serious challenges

POpULar

- Quasi Newton minimiser
- Limited-memory version of the L-BFGS
- Comes in lineair (CONGRAD-like) and non-lineair version (M1QN3-like)
- Does not require eigenvector decomposition B
- Implement in ***Pyshell***?
- For details: Geosci. Model Dev. Discuss., doi:10.5194/gmd-2016-232, 2016

Since this is the 25th meeting, looking back at one of the first TM5 meetings in 2004

TM5 Documentation contest

- Special offer: the first user who is able to commit a substantial new part to the documentation will receive a nice bottle of something during the next meeting ...
- The jury has decided to reject MC Krol as winner for the contest started in June, since he did not provide the address of the website to the users ...
- Start of new contest: one week after receiving the minutes of this TM meeting
- Success!