TM5-4DVAR CO_2

S. Houweling, S. Basu, W. Peters,
T. Kaminski

Outline

- Reorganization of 4D-VAR code
- 4D-VAR of CO_2: Posterior uncertainties

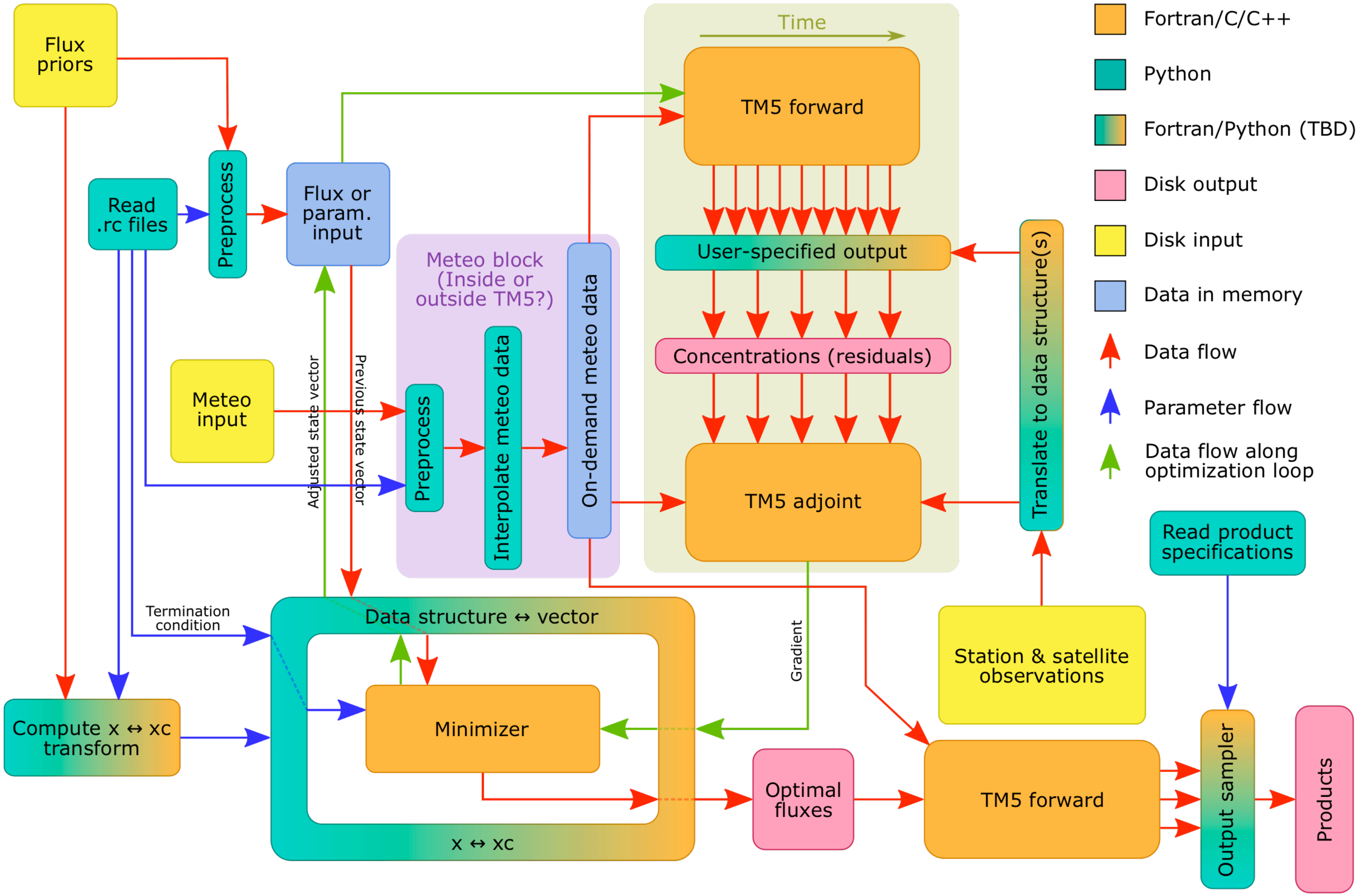
Reorganization 4D-VAR: Why?

- Parts were hard coded with specific task in mind
- Structure has become complicated to read
- It is cumbersome to change something
- Difficult to further develop code together

Reorganization 4D-VAR: How?

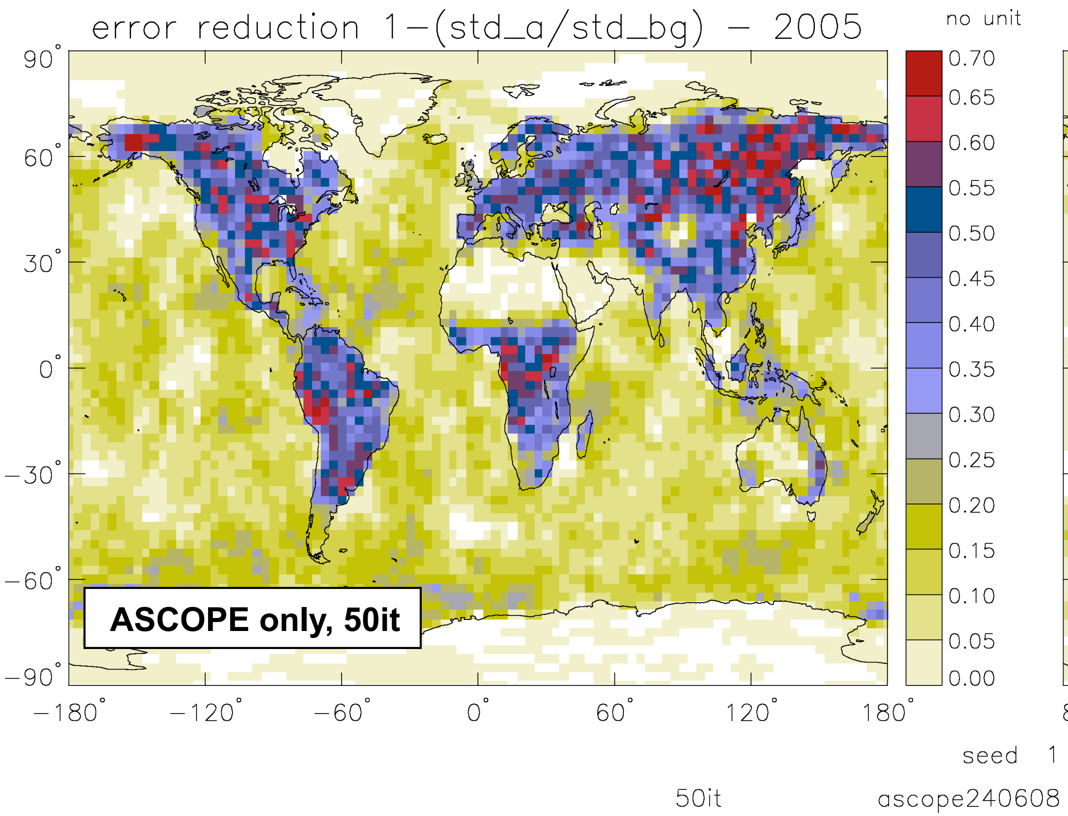
- More modular structure (exchange TM5 version, optimization routine, etc.)
- Computationally intensive parts: Fortran
- Organizational parts: Python (compatible with Wouter's EnKF)

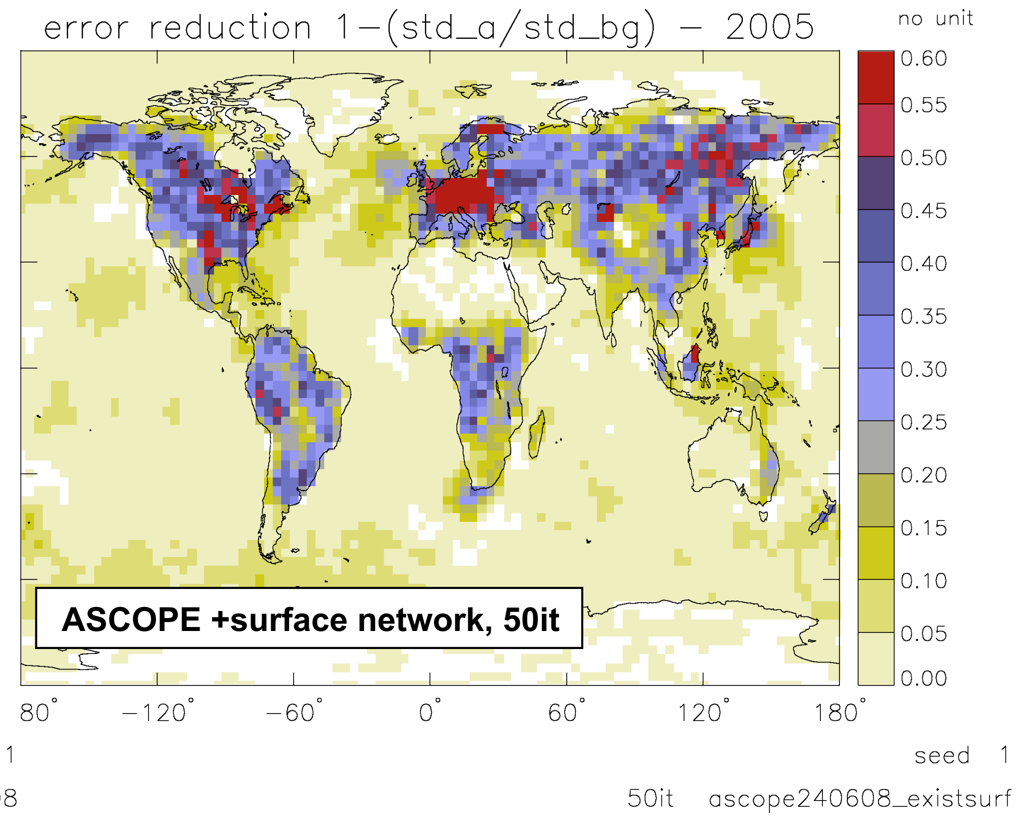
Current status: flow diagram



Posterior uncertainties: A problem?

A-SCOPE performance simulations using LMDZ 4DVAR

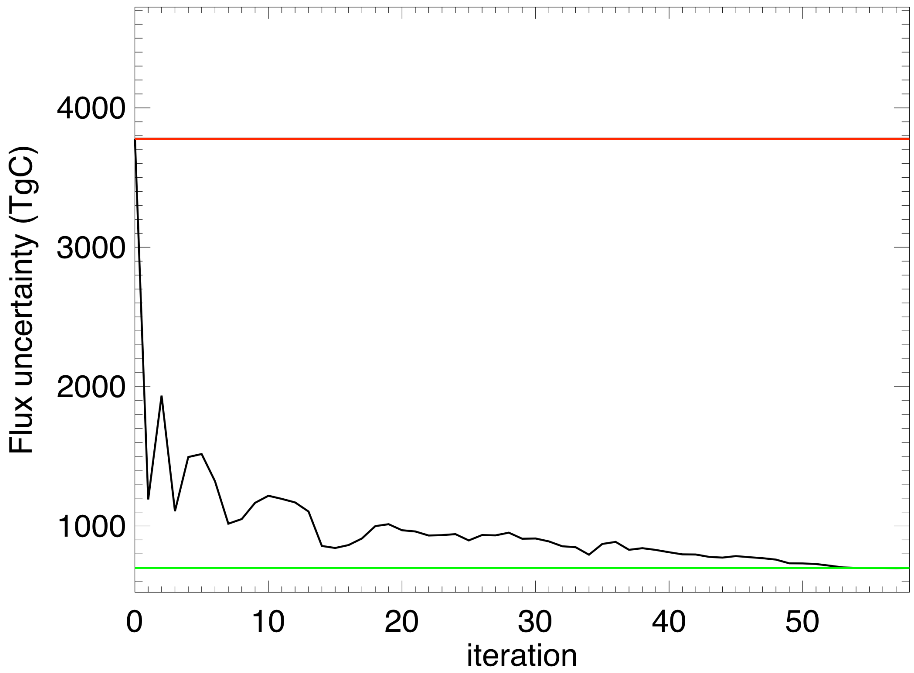




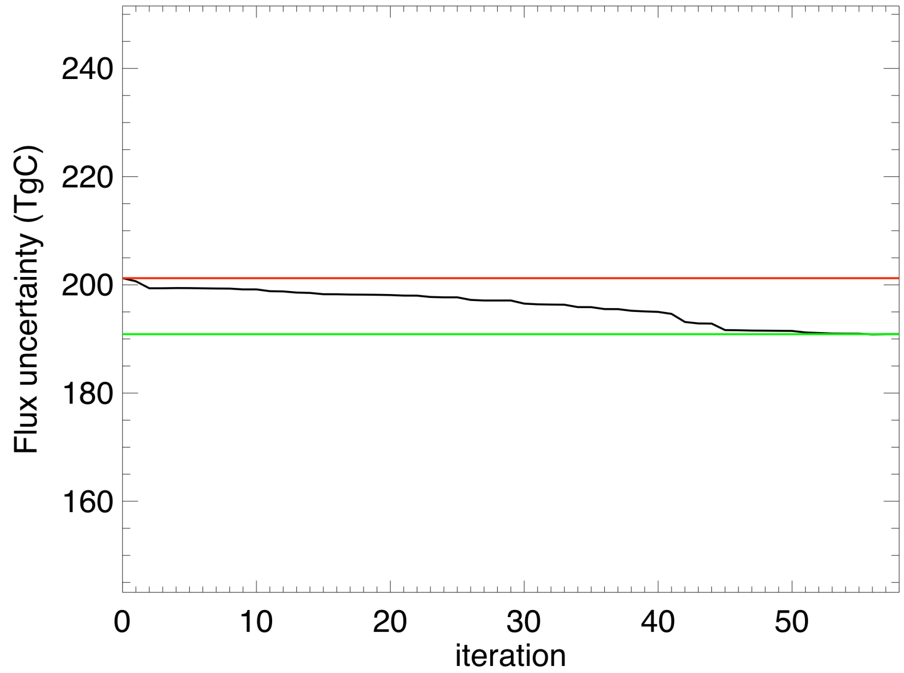
A-SCOPE + surface network doesn't always perform better ...: convergence problem

Uncertainty reduction in TM5

Globe



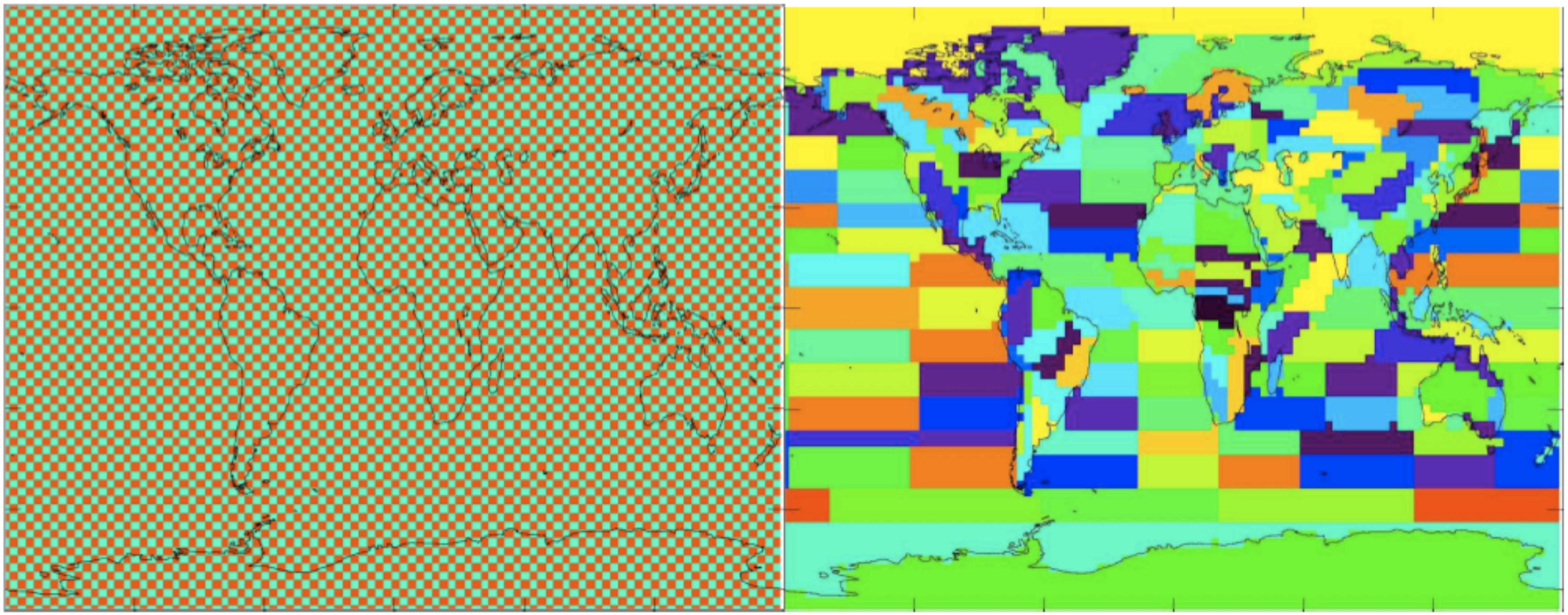
Europe



Similar convergence problem as in LMDZ ….

Solution: Matrix inversion

4D-VAR | Matrix inversion



Matrix inversion: Exact posterior uncertainties, but much lower resolution

Low resolution uncertainties are not a problem, but high resolution desired for fluxes

Option:
4D-VAR selected uncertainties only

Cost function:

$$J(x) = \frac{1}{2}(Tx - d)^T (Tx - d) + \frac{1}{2}(x - p)^T (x - p)$$

Selected posterior uncertainties:

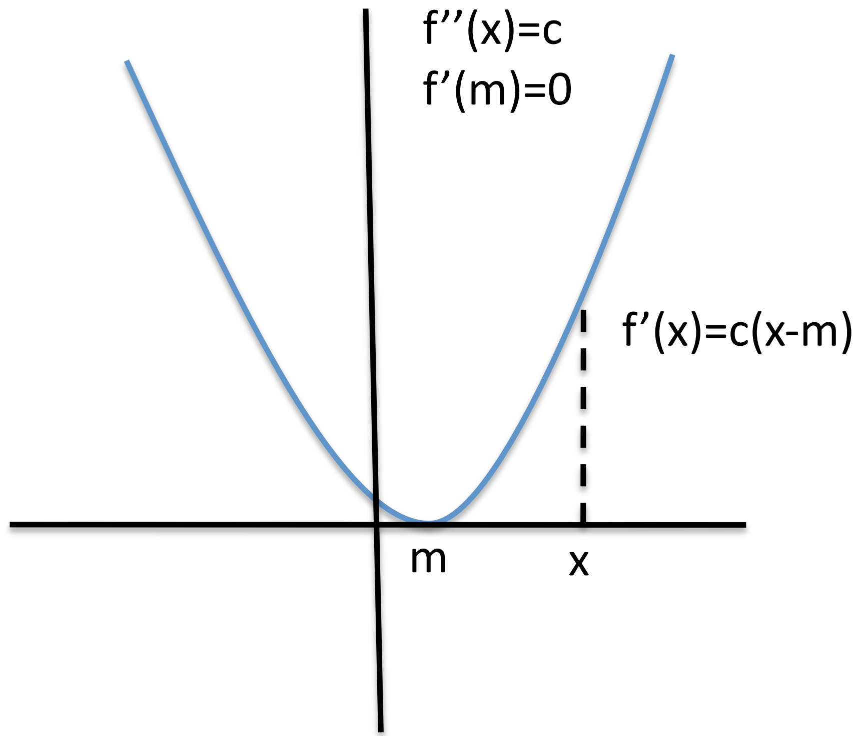
$$w^T C(m) v = w^T H^{-1} v \ .$$

Trick 1

(thanks to Thomas Kaminski)

Multi dimensional parabola:

$$g(x) = H(x - m)$$



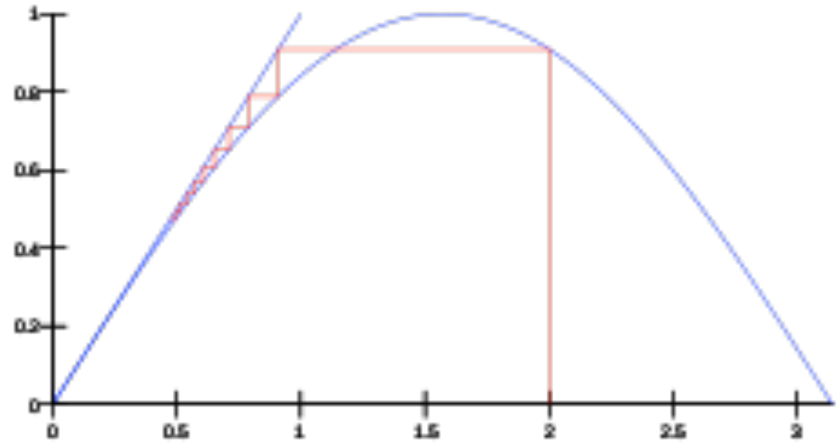
Consequence: You only need adjoint TM5 (i.e. g= f') to evaluate Hv

But, what we need is $H^{-1}v$ … => solve $u=H^{-1}v$

Trick 2

Solve $u=H^{-1}v$ by fixed point iteration:

$$x_{n+1}=f(x_n)$$



In our case:

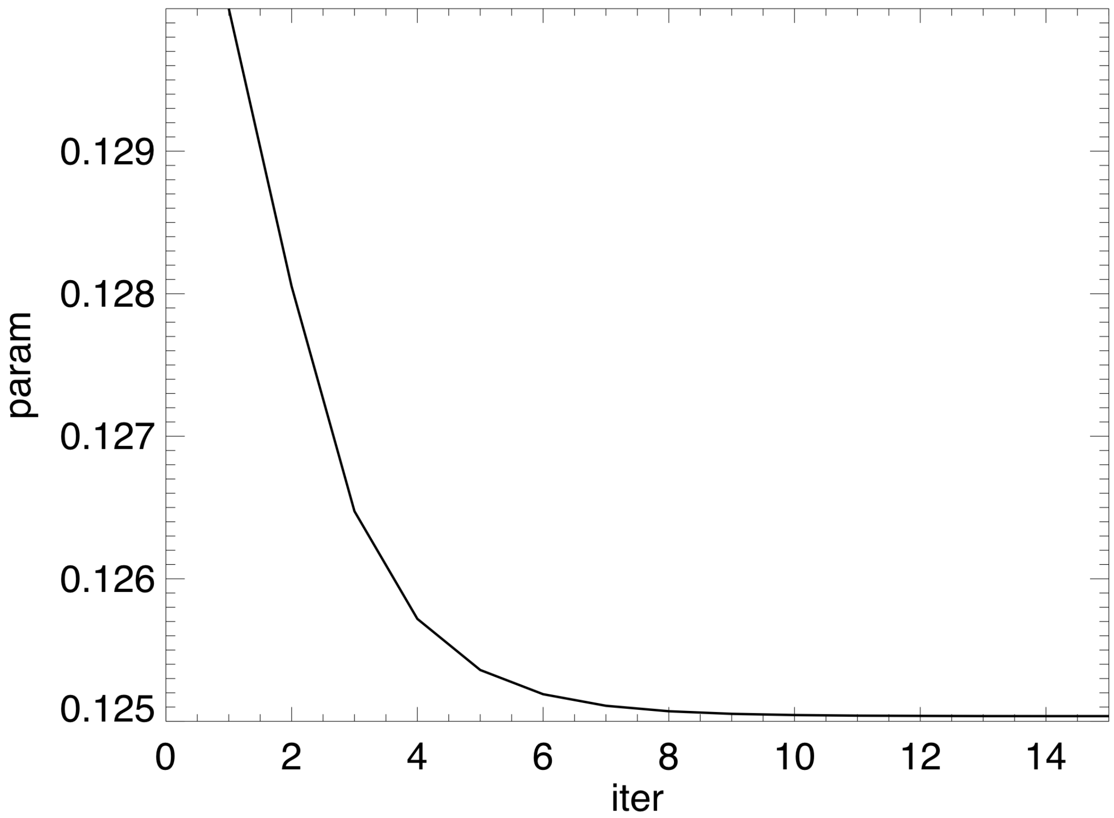
$$u_{n+1} = Qv + (I - QH)u_n$$

Approximation of H^{-1}

TM5 adjoint

Does it converge?

First test using 'few box model':



Next: will it work for a realistic case?