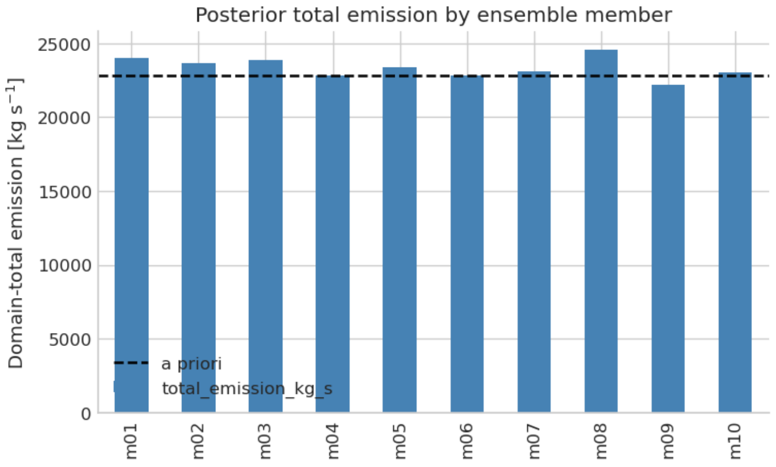
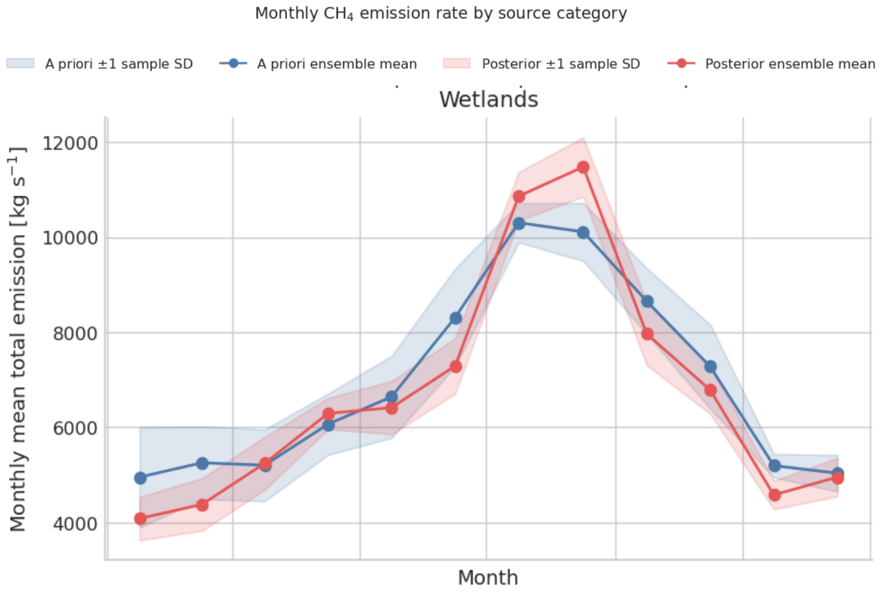
Ensemble 4D-Var

For posterior uncertainty estimates

Arjo Segers, Tyler Wizenberg (TNO)





Ensemble-4DVar

Standard 4D-Var optimisation minimizes cost function:

$$J(x) = -\frac{1}{2}(x - x_b)^T B^{-1}(x - x_b) - \frac{1}{2}(y - Hx)^T R^{-1}(y - Hx)$$

The x_a that minimizes $J(x)$ depends on the background x_b and observations y.

Ensemble-4DVar minimizes N times the cost function for perturbed input:

$$J(x) = -\frac{1}{2}(x - x_b + \delta x_i)^T \boldsymbol{B}^{-1}(x - x_b + \delta x_i) - \frac{1}{2}(y - Hx + \delta y_i)^T \boldsymbol{R}^{-\mathbf{1}}(y - Hx + \delta y_i)$$

with:

$$\delta \boldsymbol{x}_i \sim N(\boldsymbol{o}, \boldsymbol{B}), \qquad \delta y_i \sim N(\mathbf{o}, \mathbf{R}), \qquad i = 1, .., N$$

providing N different solutions $x_{a,i}$

Not to be confused with "4D-Ensemble-Var" !
That describes the background covariance B using an ensemble ...

Branches and documentation

Two code branches:

- TM5-4DVAR: ajs/cams-ch4_ens
- TM5-MP : ajs/cams_ch4_ens

In User-Guide of TM5-4DVAR, chapter "Ensemble 4D-Var"

(not online on gitlab-pages, that holds the user-guide from the master branch)

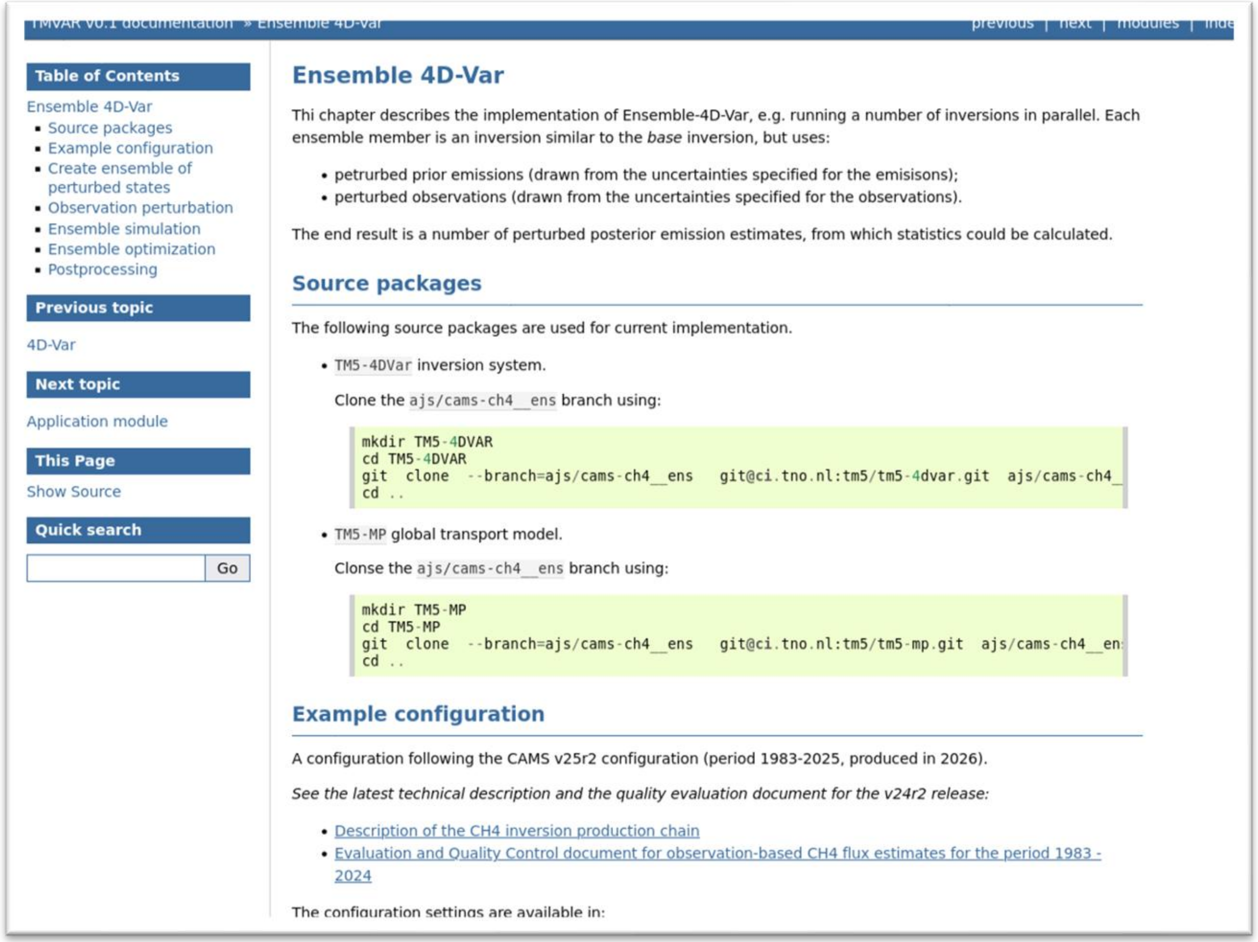


Table of Contents

Previous topic

4D-Var

Next topic

Application module

This Page

Show Source

Quick search

Go

Ensemble 4D-Var

Thi chapter describes the implementation of Ensemble-4D-Var, e.g. running a number of inversions in parallel. Each ensemble member is an inversion similar to the *base* inversion, but uses:

- petrurbed prior emissions (drawn from the uncertainties specified for the emisisons);
- perturbed observations (drawn from the uncertainties specified for the observations).

The end result is a number of perturbed posterior emission estimates, from which statistics could be calculated.

Source packages

The following source packages are used for current implementation.

- `TM5-4DVar` inversion system.

  Clone the `ajs/cams-ch4__ens` branch using:

```
mkdir TM5-4DVAR
cd TM5-4DVAR
git clone --branch=ajs/cams-ch4__ens  git@ci.tno.nl:tm5/tm5-4dvar.git  ajs/cams-ch4_
cd ..
```

- `TM5-MP` global transport model.

  Clonse the `ajs/cams-ch4__ens` branch using:

```
mkdir TM5-MP
cd TM5-MP
git clone --branch=ajs/cams-ch4__ens  git@ci.tno.nl:tm5/tm5-mp.git  ajs/cams-ch4__en
cd ..
```

Example configuration

A configuration following the CAMS v25r2 configuration (period 1983-2025, produced in 2026).

See the latest technical description and the quality evaluation document for the v24r2 release:

- Description of the CH4 inversion production chain
- Evaluation and Quality Control document for observation-based CH4 flux estimates for the period 1983 - 2024

The configuration settings are available in:

State perturbation

State vector contains emission perturbation factors:

state						correlation length scales		
main element	sub element	units	shape	x^b	σ	horizontal (Gaussian)	vertical	temporal (exponential)
emission deviation values (semi-exponential)	wetlands	1	(lon,lat,time)	0	100%	500 km	-	0
	rice	1	(lon,lat,time)	0	100%	500 km	-	0
	biomass burning	1	(lon,lat,time)	0	100%	500 km	-	0
	other	1	(lon,lat,time)	0	50%	500 km	-	9.5 month

Goal: generate *N* state perturbations:

$$\delta \boldsymbol{x}_i \sim N(\boldsymbol{o}, \boldsymbol{B})$$

From the *pre-conditioner* we have available the covariance factorization:

$$\mathbf{B} \;=\; \mathbf{B}^{1/2}\,\mathbf{B}^{T/2}$$

Draw normal distributed *pre-conditioned* state vectors:

$$\mathbf{w}_i \;\sim\; \mathcal{N}(\mathbf{o}, \mathbf{I})$$

then the transformed states have the required statistics:

$$\delta \boldsymbol{x}_i = \boldsymbol{B}^{1/2}\,\boldsymbol{w}_i$$

(Same operation is used in the pre-conditioning)

$$E_{i,j,t}(\gamma_{i,j,t}) \;=\; \begin{cases} E^b_{i,j,t}\,\exp(\gamma_{i,j,t}), & \gamma_{i,j,t} < 0 \\ E^b_{i,j,t}\,(1+\gamma_{i,j,t}), & \gamma_{i,j,t} \geq 0 \end{cases}$$

Expansion from state to emissions.

Observation representation perturbation

Generate *N* observation representation perturbations:

$$\delta y_i \sim N(\mathbf{o}, \mathbf{R})$$

NOTE: not just perturbation of measurements ***y***, but perturbation of meas. minus model ***y*** - ***Hx*** !

Observation representation error **R** has contributions from:

- observation error:
    - measurement error
    - temporal sampling error
- model representation error:
    - spatial sampling error
    - temporal sampling error

Implementation: added array with perturbation factors to "point_input.nc4", applied when computing gradient forcing

```
netcdf point_input {
dimensions:
  id = 1468 ;
  member = 11 ;
  ...
variables:
  ...
  double mixing_ratio(id) ;
  double mixing_ratio_stdv(id) ;
  double mixing_ratio_error(id) ;
  ...
  double perturb(id, member) ;
    perturb:units = "1" ;
    perturb:long_name = "perturbation factor out of N(0,1) per ensemble member" ;

}
```

Ensemble definition

Convention: *"Always run ensemble, first member is unperturbed"*

Special case: micro-ensemble of 1 member)

Definition of ensemble by member names and tracer names:

```
! ensemble names:
ens.members         :  m00 m01 m02 m03 m04 m05 m06 m07 m08 m09 m10
! ensemble tracer names:
ens.tracers         :  CH4 CH4_m01 CH4_m02 CH4_m03 CH4_m04 CH4_m05 CH4_m06 CH4_m07 CH4_m08 CH4_m09 CH4_m10
```

default member and tracer (unperturbed)

Implementation in job-tree

```
tmvar
     .setup
           .emis
           .covar
     .xb


     .model
           .b01
           .b02
           .b03
           ...
     .sensi
           .b01 .b02 .b03   ...
     .var4d
           .iter-0001
                     .fwd
                       .b01  .b02  .b03  ...
                     .dep
                       .b01  .b02  .b03  ...
                     .grd
                       .b01  .b02  .b03  ...
                     .g2gc

                     .opt

                     .xc2x

             :
     .apos
          .b01
          .b02
          .b03
          ...
```

Parallelism in single inversion:

- time parallel blocks (Pandey et al.)
- model simulations with domain decomposition

Implementation in job-tree

```
tmvar
     .setup
           .emis
           .covar
     .xb
     .ens
        .m00 m01 m02 m03 m04 m05 ...
     .model
           .b01
           .b02
           .b03
           ...
     .sensi
           .b01 .b02 .b03  ...
     .var4d
           .iter-0001
                     .fwd
                         .b01  .b02  .b03  ...
                     .dep
                         .b01  .b02  .b03  ...
                     .grd
                         .b01  .b02  .b03  ...
                     .g2gc
                          .m00 m01 m02 m03 m04 m05 ...
                     .opt
                          .m00 m01 m02 m03 m04 m05 ...
                     .xc2x
                          .m00 m01 m02 m03 m04 m05 ...
            :
     .apos
          .b01
          .b02
          .b03
          ...
```

Parallelism in single inversion:

- time parallel blocks (Pandey et al.)
- model simulations with domain decomposition and multiple tracers (1 per ensemble member)
- ensemble member processed in parallel

Ensemble simulation in TM5-MP

Changes in "`chem_param.F90`":

tracer per ensemble member:

```
! ntrace: number of tracers for chemistry
integer, parameter                            ::  ntrace  = 11

integer, parameter                            ::  tracer_name_len = 8
character(len=tracer_name_len), parameter     ::  names(maxtrace) = (/ &
      'CH4     ', &
      'CH4_m01 ', &
      'CH4_m02 ', &
      'CH4_m03 ', &
      'CH4_m04 ', &
      'CH4_m05 ', &
      'CH4_m06 ', &
      'CH4_m07 ', &
      'CH4_m08 ', &
      'CH4_m09 ', &
      'CH4_m10 ' /)
```

This file is now *generated* from the settings *(similar as the "*`dims_grid.F90`*" file ...)*

names of ensemble members:

```
! ensemble size, incl. base:
integer, parameter                  ::  nens = 11

! member names:
character(len=8), parameter         ::  ens_name(nens) = (/ &
      'm00     ', &
      'm01     ', &
      'm02     ', &
      'm03     ', &
      'm04     ', &
      'm05     ', &
      'm06     ', &
      'm07     ', &
      'm08     ', &
      'm09     ', &
      'm10     ' /)
```

mapping from ensemble members to tracers:

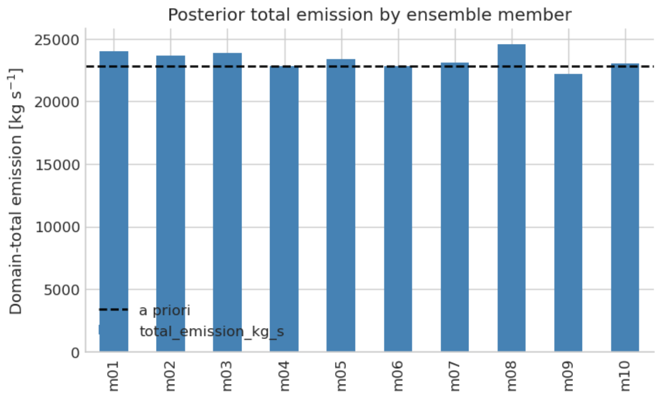
```
! number of tracers per ensemble member:
integer, parameter                  ::  ens_ntrace = 1

! mapping to tracer index:
integer, parameter                  ::  ens_itrace(nens,ens_ntrace) = reshape( (/ &
      1, &
      2, &
      3, &
      4, &
      5, &
      6, &
      7, &
      8, &
      9, &
      10, &
      11 /), (/nens,ens_ntrace/) )
```

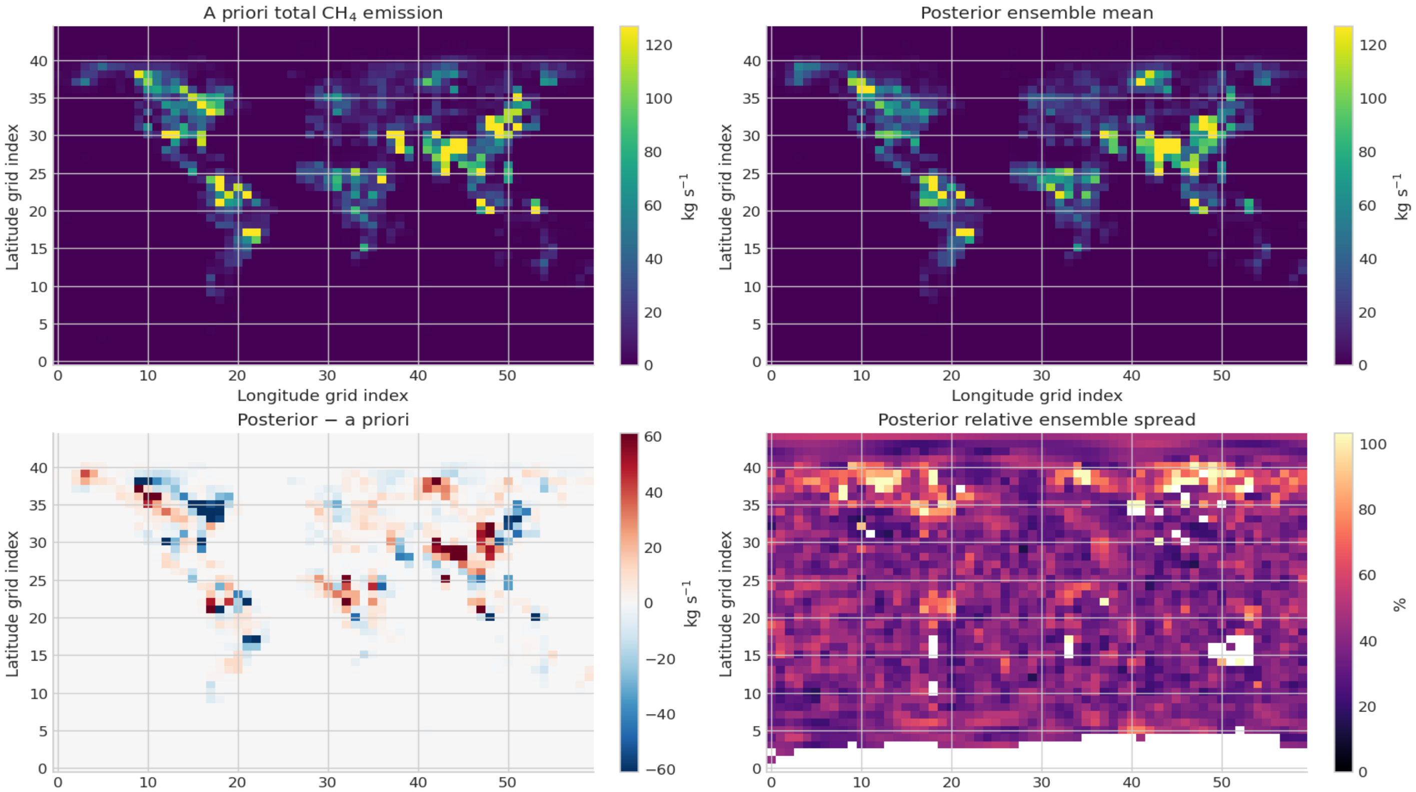
TNO innovation for life

Prior and posterior uncertainties

Tested on coarse resolution (6x4 deg.) and short period (1 year), 1+10 members

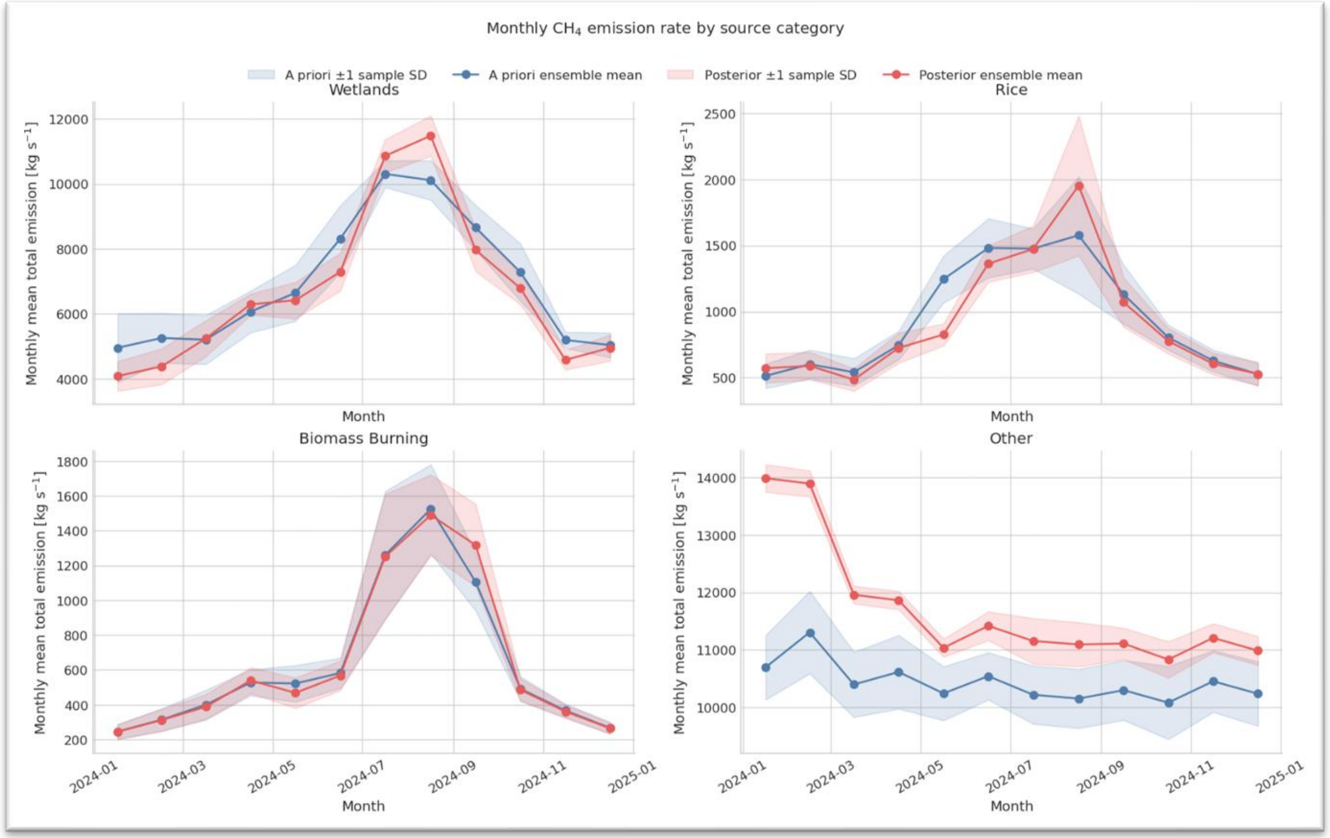
- Basic checks: does prior ensemble have the correct statistics?
    - relative std.dev: ok
    - spatial correlation: ok
    - temporal correlation: ok
- Example of emissions:



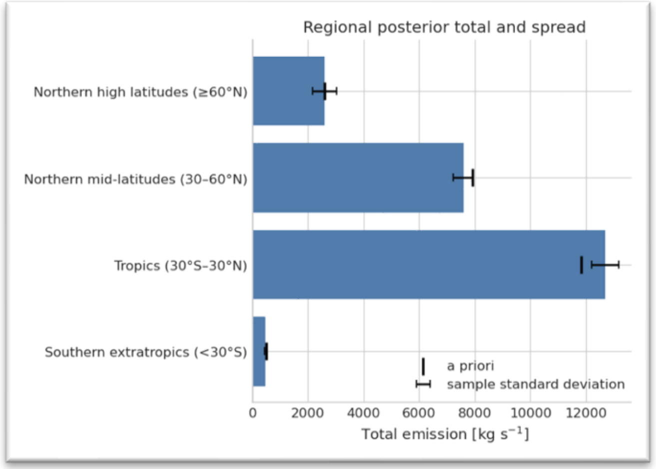


Prior and posterior uncertainties

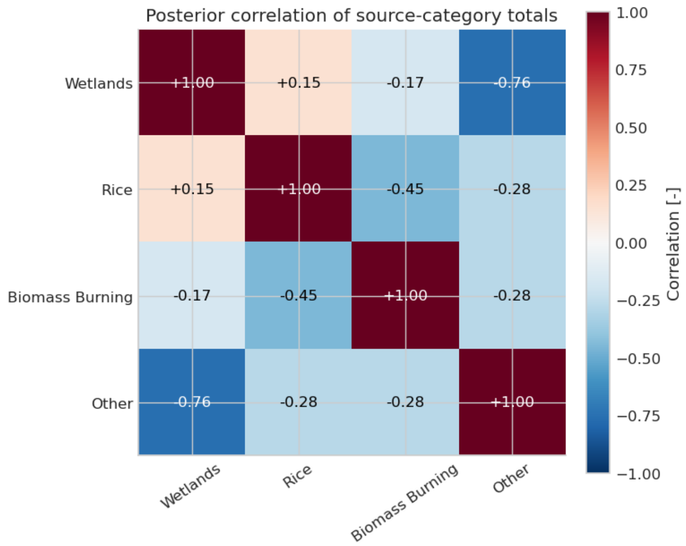
Drawing aggregated statistics from ensembles:



Time series of monthly global emissions per category.



Aggregation per zonal band.



Correlation between optimized categories

innovation for life

Summary and outlook

- Implemented ensemble in TM5-4DVar
  - First results ok (coarse resolution, 1 year)
  - *Minor fixes needed: time-parallel-block processing*

- Practical use in CAMS/CH4 inversions:
  - Too expensive on 1x1 resolution *(Fortran version ...)*
  - Apply on coarser resolution:
    - *sufficient for aggregated emissions*
  - Apply on short period (1 year), use for all years?
    - *uncertainties are consistent in time?*

- Use the prior ensemble to tune uncertainties?
  Observation-minus-simulation statistics should match with uncertainties

- *"Faster model? Let's spent the time on an ensemble!"*