Towards integrated tools for inversion studies: the Community Inversion Framework

Antoine Berchet, Rona Thompson, Tuula Aalto, Peter Bergamaschi, Grégoire Broquet, Dominik Brunner, Frédéric Chevallier, Richard Engelen, Christoph Gerbig, Christine Groot Zwaaftink, Stephan Henne, Sander Houweling, Ute Kartens, Werner L. Kutsch, Ingrid van der Laan-Luijkx, Guillaume Monteil, Paul I. Palmer, Wouter Peters, Philippe Peylin, Isabelle Pison, Christian Rödenbeck, Marielle Saunois, Marko Scholze, Espen Sollum, Aki Tsuruta

LABORATOIRE DES SCIENCES DU CLIMAT & DE L'ENVIRONNEMENT

20 years of successful developments

Past developments

- simple analytical inversions turned into complex systems
- use of a large diversity of products (in-situ, satellites, etc.)
- application to all scales and many species

Achievements

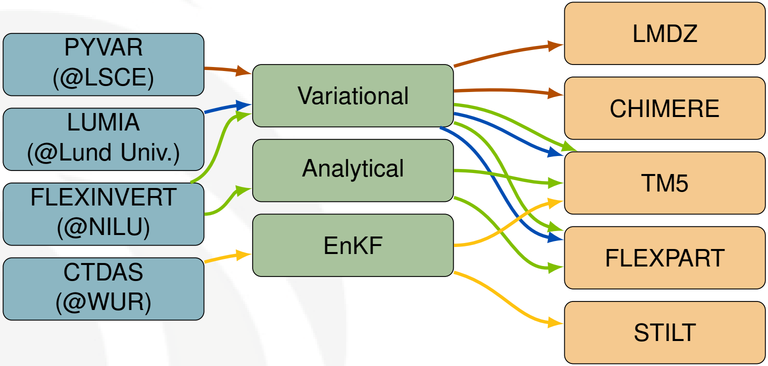
- quasi-operational systems
- recognized constraints on anthropogenic emissions and atmospheric budgets in general

Future needs and directions

- new high-resolution satellites
- request for operational validation of national reporting

⇒ multi species / model / scale / solver systems

Current atmospheric inversion landscape: examples



Pros

- ✓ dynamic community
- ✓ necessary and fruitful diversity

Cons

- ✗ redundant developments
- ✗ loss of efficiency to propagate new developments
- ✗ lack of inter-comparability

Proposed solution: the Community Inversion Framework

Objectives

1. rationalize development efforts
2. foster cross-compatibility and inter-comparability of inversion systems
3. ensure quality control with better traceability and transparency
4. open the way towards operational systems

Means

1. define common standards and protocols in the inversion community
2. provide a unified framework for harmonized good practices, code structure and input/output formats

$\Rightarrow$ define a common level of granularity for functions and objects

Back to basics: classical methods

Analytical solution

$$\begin{cases} \mathbf{x}^a = \mathbf{x}^b + \mathbf{K}(\mathbf{y}^o - \mathbf{H}\mathbf{x}) \\ \mathbf{P}^a = \mathbf{P}^b - \mathbf{K}\mathbf{H}\mathbf{P}^b \end{cases} \quad \text{with } \mathbf{K} = \mathbf{P}^b\mathbf{H}^T(\mathbf{R} + \mathbf{H}\mathbf{P}^b\mathbf{H}^T)^{-1}$$

Variational inversion

$$\begin{cases} J(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}^b)^T(\mathbf{P}^b)^{-1}(\mathbf{x} - \mathbf{x}^b) + \frac{1}{2}(\mathbf{y}^o - \mathcal{H}(\mathbf{x}))^T\mathbf{R}^{-1}(\mathbf{y}^o - \mathcal{H}(\mathbf{x})) \\ \nabla J_{\mathbf{x}} = (\mathbf{P}^b)^{-1}(\mathbf{x} - \mathbf{x}^b) + \mathcal{H}^*(\mathbf{R}^{-1}(\mathbf{y}^o - \mathcal{H}(\mathbf{x})) \\ \quad\quad = \chi + \mathcal{H}^*(\mathbf{R}^{-1}(\mathbf{y}^o - \mathcal{H}(\mathbf{x})) \end{cases}$$

Ensemble Kalman filters

$$\begin{cases} \mathbf{H}\mathbf{P}^b\mathbf{H}^T \simeq \frac{1}{N-1}(\mathcal{H}(\mathbf{x}_1), ..., \mathcal{H}(\mathbf{x}_N)) \cdot (\mathcal{H}(\mathbf{x}_1), ..., \mathcal{H}(\mathbf{x}_N))^T \\ \mathbf{P}^b\mathbf{H}^T \simeq \frac{1}{N-1}(\mathbf{x}_1, ..., \mathbf{x}_N) \cdot (\mathcal{H}(\mathbf{x}_1), ..., \mathcal{H}(\mathbf{x}_N))^T \end{cases}$$

Back to basics: decomposition of operations

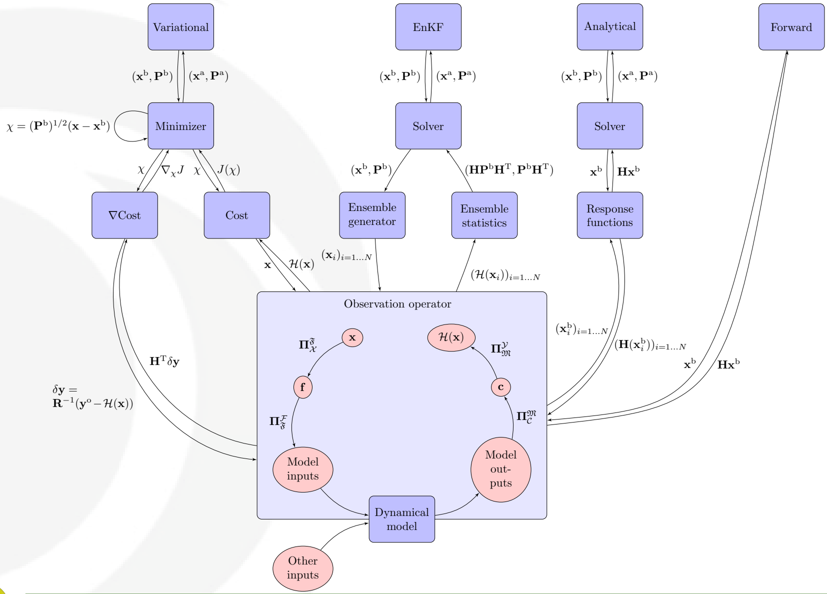
Variational inversion

$$\begin{cases} J(\mathbf{x}) &= \frac{1}{2}(\mathbf{x}-\mathbf{x}^{\mathrm{b}})^{\mathrm{T}}(\mathbf{P}^{\mathrm{b}})^{-1}(\mathbf{x}-\mathbf{x}^{\mathrm{b}}) + \frac{1}{2}(\mathbf{y}^{\mathrm{o}}-\mathcal{H}(\mathbf{x}))^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{y}^{\mathrm{o}}-\mathcal{H}(\mathbf{x})) \\ \nabla J_{\mathbf{x}} &= (\mathbf{P}^{\mathrm{b}})^{-1}(\mathbf{x}-\mathbf{x}^{\mathrm{b}}) + \mathcal{H}^*(\mathbf{R}^{-1}(\mathbf{y}^{\mathrm{o}}-\mathcal{H}(\mathbf{x})) \\ &= \chi + \mathcal{H}^*(\mathbf{R}^{-1}(\mathbf{y}^{\mathrm{o}}-\mathcal{H}(\mathbf{x})) \end{cases}$$

Required operations

- $\begin{cases} \mathcal{X} & \rightarrow & \mathcal{Y} \\ \mathbf{x} & \rightarrow & \mathcal{H}(\mathbf{x}) \end{cases}$
- $\begin{cases} \mathcal{Y}^* & \rightarrow & \mathcal{Y}^* \\ \delta\mathbf{y} & \rightarrow & \mathbf{R}^{-1}\delta\mathbf{y} \end{cases}$
- $\begin{cases} \mathcal{Y}^* & \rightarrow & \mathcal{X}^* \\ \delta\mathbf{y} & \rightarrow & \mathcal{H}^*(\delta\mathbf{y}) \end{cases}$
- $\begin{cases} \mathcal{X}^* & \rightarrow & \mathfrak{A}^* \\ \delta\mathbf{x} & \rightarrow & \delta\chi \equiv (\mathbf{P}^{\mathrm{b}})^{1/2}\delta\mathbf{x} \end{cases}$
- $\begin{cases} \mathfrak{A} & \rightarrow & \mathcal{X} \\ \chi & \rightarrow & \mathbf{x} \equiv (\mathbf{P}^{\mathrm{b}})^{1/2}\chi \end{cases}$

Diagram of the target framework



Example of CIF objects: the observation operator

obsoperator:	- purpose:	maps data from the control space to the observation space and conversely			
	- inheritance:	- controlvect			
		- obsvect			
		- model			
	- metadata	none			
	- data	none			
	- methods:	- obsoper:	$\mathbf{x}$	$\rightarrow$	$\mathcal{H}(\mathbf{x})$
			$\mathcal{H}^*(\mathbf{y}^*)$	$\leftarrow$	$\mathbf{y}^*$
	- calls:	- controlvect.$\mathbf{\Pi}_{\mathcal{X}}^{\mathfrak{F}}$:	$\mathbf{x}$	$\leftrightarrow$	$\mathbf{x}_{\mathrm{model}}$
		- model.$\mathbf{\Pi}_{\mathfrak{F}}^{\mathcal{F}}$:	$\mathbf{x}_{\mathrm{model}}$	$\leftrightarrow$	model inputs
		- model.run:	model inputs	$\leftrightarrow$	model outputs
		- model.$\mathbf{\Pi}_{\mathcal{C}}^{\mathfrak{M}}$:	model outputs	$\leftrightarrow$	$\mathbf{y}_{\mathrm{model}}$
		- obsvect.$\mathbf{\Pi}_{\mathfrak{M}}^{\mathcal{Y}}$:	$\mathbf{y}_{\mathrm{model}}$	$\leftrightarrow$	$\mathbf{y}^{\mathrm{o}}$

Example of CIF objects: the model class

model:	- purpose:	drives the dynamical model, prepares inputs and extracts outputs to CIF-compatible structures			
	- inheritance:	none			
	- metadata	- resolution - computation mode - sub-periods if any - chemistry if any - model-specific configuration - path to fixed inputs (e.g., meteo data)			
	- data	none			
	- methods:	- run:	model inputs	$\rightarrow$	model outputs
		- $\boldsymbol{\Pi}_{\mathcal{F}}^{\mathcal{C}}$:	$\mathbf{x}_{\mathrm{model}}$	$\leftrightarrow$	model inputs
		- $\boldsymbol{\Pi}_{\mathcal{C}}^{\mathfrak{M}}$:	model output	$\leftrightarrow$	$\mathbf{y}_{\mathrm{model}}$
	- calls:	none			

Community development

Timeline and method

- framework fully defined with templates by mid-2019
- first common bricks implemented by 2020
- Python environment
- documentation and free access through a centralized portal

Take-home message

Identified needs

- improve overall efficiency as a community to keep up with technical and theoretical developments
- enhance transparency, inter-comparability and inter-operability toward operational use and policy-making support

Community-scale convergence and collaboration

- harmonized practices and formats
- common coding environment for inversion bricks

Thanks for your attention