Data Assimilation for Numerical Weather Prediction

[NWP] Project

Ahmed Attia

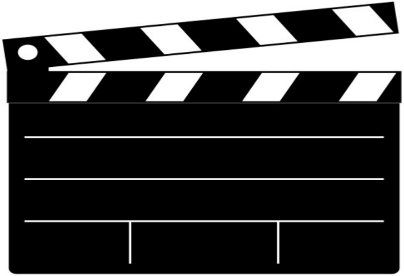
Statistical and Applied Mathematical Science Institute (SAMSI)
19 TW Alexander Dr, Durham, NC 27703
attia@ {vt.edu || samsi.info}

Department of Mathematics
North Carolina State University
amattia2@ncsu.edu

SAMS/NCSU UG-Workshop
May 15, 2017

Motivation: Advection-Diffusion

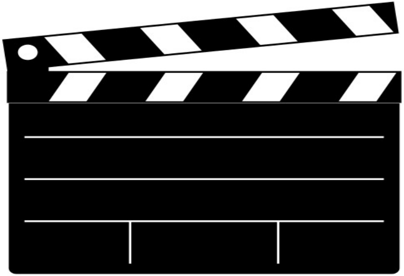
- Consider the concentration of a contaminant u in the domain $\Omega \in \mathbb{R}^2$:



- **Simulation:** given the initial condition $\mathbf{x}_0$, a forward discretized model $\mathcal{F}$, integrate/solve the PDEs forward in time!
- **Forward problem:** given model state $\mathbf{x}$, predict model observations $\mathbf{b} = \mathcal{H}(\mathbf{x})$

Motivation: Advection-Diffusion

- Consider the concentration of a contaminant u in the domain $\Omega \in \mathbb{R}^2$:



- **Simulation:** given the initial condition $\mathbf{x}_0$, a forward discretized model $\mathcal{F}$, integrate/solve the PDEs forward in time!
- **Forward problem:** given model state $\mathbf{x}$, predict model observations $\mathbf{b} = \mathcal{H}(\mathbf{x})$
- **Inverse problem: given noisy, and sparse observation $\mathbf{y}$, and "possibly" uncertain model state $\mathbf{x}^{\rm b}$, recover/estimate the unknown model state $\mathbf{x}^{\rm true}$**
- **Design of experiments:**
  e.g. : sensor placement for optimal reconstruction of parameter

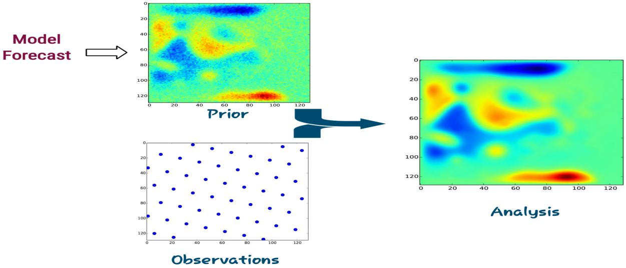
Motivation

"All models are wrong but some are useful"

Box, G. E. P. (1979), "Robustness in the strategy of scientific model building", in Launer, R. L.; Wilkinson, G. N., Robustness in Statistics, Academic Press, pp. 201 – 236.

Motivation

- **Inverse problems and Data Assimilation (DA)**:



$$\underbrace{\textbf{Model} + \textbf{Prior} + \textbf{Observations}}_{\text{with associated \underline{uncertainties}}} \rightarrow \underbrace{\textbf{Best description of the state}}_{\textit{Variational+Ensemble}}$$

- **Applications include**: atmospheric forecasting, power flow, oil reservoir, volcano simulation, etc.

DA: Statistical Inverse Problems

- **Statistical formulation**:
  - **The prior** $\mathcal{P}^{\mathrm{b}}(\mathbf{x})$: encapsulates our knowledge about the system state prior to incorporating additional information.
  - **The likelihood** $\mathsf{P}(\mathbf{y}|\mathbf{x})$: describes the mismatch between what is observed and what the model predicts to be observed.
  - **The posterior** $\mathsf{P}(\mathbf{x}|\mathbf{y})$: probability distribution of the system state conditioned by the collected observations. **This is the probabilistic solution of the inverse problem!**
- Bayes' theorem $\longrightarrow$ $\mathcal{P}^{\mathrm{a}}(\mathbf{x}) = \mathsf{P}(\mathbf{x}|\mathbf{y}) = \frac{\mathsf{P}(\mathbf{y}|\mathbf{x})\mathcal{P}^{\mathrm{b}}(\mathbf{x})}{\mathsf{P}(\mathbf{y})} \propto \mathsf{P}(\mathbf{y}|\mathbf{x})\mathcal{P}^{\mathrm{b}}(\mathbf{x})\,.$
- Simplifying assumptions are imposed on the error distribution (e.g. background error, observation errors, etc.).
- "*Typically*", errors are assumed to be Gaussian (*Easy, tractable, ...*).

The Gaussian Framework

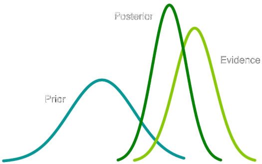
- **The Gaussian framework**: errors are modeled as Gaussian random variables:

$$\mathbf{x}^{\rm b} - \mathbf{x}^{\rm true} \sim \mathcal{N}(0, \mathbf{B}), \quad \mathbf{y} - \mathcal{H}(\mathbf{x}^{\rm true}) \sim \mathcal{N}(0, \mathbf{R}) \,,$$

$$\mathbf{x} \in \mathbb{R}^{\rm N_{state}}, \, \mathbf{y} \in \mathbb{R}^{\rm N_{obs}}, \, {\rm N_{obs}} \ll {\rm N_{state}} \,.$$

- For linear dynamics $\mathcal{F}$, and linear observation operator $\mathcal{H}$, **the posterior is Gaussian.**

- **So what?**



- The posterior PDF represents improved knowledge about $\mathbf{x}$
- The MAP (posterior mode/mean) can be taken as best estimate (analysis) of the unknown truth $\mathbf{x}^{\rm true}$
- The posterior variance/covariance can be taken to express the uncertainty associated with the analysis.

The Gaussian Framework: Limitations

Remember the Gaussian PDF $\mathcal{N}(\overline{x}, \Sigma)$?

$$\mathrm{P}(\mathbf{x}) \propto e^{-\frac{1}{2}(\mathbf{x}-\overline{x})^T \, \Sigma^{-1} \, (\mathbf{x}-\overline{x})}$$

The Gaussian Framework: Limitations

- **Consider atmospheric forecasting:**
  1. Assume we are interested in 3 prognostic/physical variables; e.g. humidity, pressure, vertical and wind-speed, at points of a grid of size 1000×1000 in the XY plane. **The discrete state is of size** 3×10^6.
  2. **The uncertainty, e.g. covariance matrix is of size** 9×10^{12}.
  3. Storing (36 TB), and manipulating (e.g. inverting) such matrix is infeasible!
- *Monte-Carlo (ensemble-based) approach is followed in practice,*
  i.e. probability distributions are approximated by samples/ensembles!
- **Popular/Practical algorithms**:
  + E.g.: EnKF, MLEF, IEnKF, RIP, PF, EnKS, ...
  + **By far, the most popular is EnkF,**
  + **Many flavors of EnKF exist.**

A Standard EnKF algorithm

Given an ensemble of $\mathrm{N_{ens}}$ states ($\mathbf{x}^{\mathrm{a}}_{k-1}(e)$, $e = 1, \ldots, \mathrm{N_{ens}}$) representing the analysis probability distribution at time t_{k-1}.

- **Forecast:** each member of the ensemble is propagated to t_k using the dynamical model to obtain the "forecast" ensemble:

$$\mathbf{x}^{\mathrm{b}}_k(e) = \mathcal{M}_{t_{k-1} \rightarrow t_k}(\mathbf{x}^{\mathrm{a}}_{k-1}(e)) + \eta_k(e), \quad e = 1, \ldots, \mathrm{N_{ens}}.$$

- the ensemble mean and covariance approximate the moments of the prior distribution at the next time point t_k:

$$\begin{aligned} \overline{\mathbf{x}}^{\mathrm{b}}_k &= \frac{1}{\mathrm{N_{ens}}} \sum_{e=1}^{\mathrm{N_{ens}}} \mathbf{x}^{\mathrm{b}}_k(e)\,, \\ \mathbf{X}^{\mathrm{b}}_k &= [\mathbf{x}^{\mathrm{b}}_k(1) - \overline{\mathbf{x}}^{\mathrm{b}}_k, \ldots, \mathbf{x}^{\mathrm{b}}_k(\mathrm{N_{ens}}) - \overline{\mathbf{x}}^{\mathrm{b}}_k]\,, \\ \mathbf{B}_k &= \left(\frac{1}{\mathrm{N_{ens}} - 1} \left(\mathbf{X}^{\mathrm{b}}_k \left(\mathbf{X}^{\mathrm{b}}_k \right)^T \right) \right) \circ \rho. \end{aligned}$$

- *To reduce sampling error due to the small ensemble size, **localization** is performed by taking the point-wise product of the ensemble covariance and a decorrelation matrix ρ.*
- *To avoid ensemble collapse, **inflation** is applied!*

A Standard EnKF algorithm

- **Analysis:** each member of the forecast (ensemble of forecast states $\{\mathbf{x}_k^{\mathrm{b}}(e)\}_{e=1,\ldots,\mathrm{N_{ens}}}$) is analyzed/updated separately using the Kalman filter formulas

$$
\begin{aligned}
\mathbf{x}_k^{\mathrm{a}}(e) &= \mathbf{x}_k^{\mathrm{b}}(e) + \mathbf{K}_k \left([\mathbf{y}_k + \zeta_k(e)] - \mathcal{H}_k(\mathbf{x}_k^{\mathrm{b}}(e)) \right), \\
\mathbf{K}_k &= \mathbf{B}_k \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{B}_k \mathbf{H}_k^T + \mathbf{R}_k \right)^{-1}.
\end{aligned}
$$

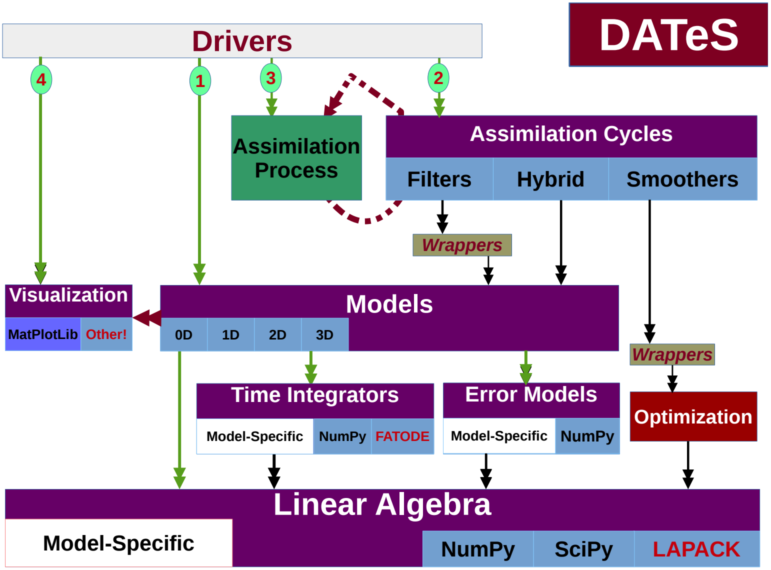
- We will learn, and implement another flavor of EnKF, namely LETKF!
- For that, we will use **DATeS**, an extensible Python-based **D**ata **A**ssimilation **Te**sting **S**uite.

DATeS: Data Assimilation Testing Suite

- Our vision at the Computational Science Laboratory (CSL) Virginia Tech, is to provide an "extensible open-source high-level language DA package" that enables DA researchers to collaborate effectively and avoid reinventing the wheel.
- **DATeS:**
  1. is intended to be a work-in-progress testing environment for DA,
  2. it separates the different building blocks so that they can be integrated with new and also legacy codes as easy as possible,
  3. it enables researchers to focus on implementing their own ideas/algorithms without worrying much about other components of the DA system.

DATeS Website:
`http://people.cs.vt.edu/~attia/DATeS/` or
`https://sibiu.cs.vt.edu/dates/index.html`

DATeS: Data Assimilation Testing Suite



NWP Project: Goal & Plan

Goal: *Learn, and implement the Local Ensemble Transform Kalman Filter (LETKF), and test it with a Quasi-Geostrophic model (see-surface elevation).*

Proposed Plan:

- **Monday & Tuesday:** **read the paper:** *Harlim, John, and Brian R. Hunt. "Local ensemble transform kalman filter: An efficient scheme for assimilating atmospheric data."*
- **Tuesday:** general Python hands-on tutorial (for everyone)
- **Tuesday:** DATes hands-on tutorial, and discuss the LETKF paper
- **Wednesday & Thursday:** implementing the LETKF filter, visualize the results and write a short report/presentation