Data assimilation for engineers

Introduction

Olivier THUAL
Toulouse INP-ENSEEIHT/MFEE

Course:
Data Assimilation (ASID)

Teaching Unit:
Numerical Methods for Scientific Computing in Aerodynamics

Where are we? MFEE/MSN and MFEE/MSN-BD

Teaching Unit: Numerical Methods for Scientific Computing

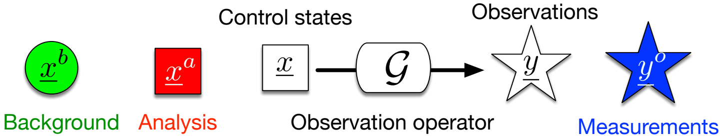
Three courses in this Teaching Unit (5 ECTS):

- Numerical methods: simulation of incompressible flows (35%)
- Numerical methods: simulation of compressible flows (35%)
- Data Assimilation (30%)

What are the learning outcomes?

At the end of the lecture, students should be able to:

- Describe the basic formalism of data assimilation
- Explain the specification of a cost fonction on several examples
- Choose several methods to minimize the cost fonction



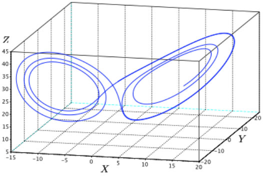
At the end of the numerical projects, student should be able to:

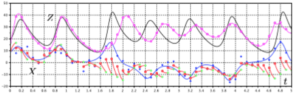
- Write computer programs for several methods with several examples
- Compare and comment the performmances of several methods
- Achieve convincing forecast for chaotic dynamical systems

Course notes

Olivier THUAL

Data assimilation for engineers





Toulouse INP - ENSEEIHT

"Fluid Mechanics, Energetics and Environment" Department

Year 2019-2020, November 29, 2020

Contents

1 From weather forecast to engineer applications **7**
- 1 How is weather forecasted . . . 8
- 2 Other examples of data assimilation applications . . . 10
- 3 What time is it? . . . 13

2 Generic cost function **19**
- 1 Looking for the minimum of a cost function . . . 20
- 2 Linearization methods . . . 22
- 3 How will the bore propagate? . . . 24

3 Time dependent models **31**
- 1 The 4D-Var data assimilation . . . 32
- 2 The Kalman filter data assimilation . . . 36
- 3 Will the water overflow? . . . 39

4 Application projects **47**
- 1 Data assimilation for the Lorenz model . . . 48
- 2 Ensemble methods for two simple examples . . . 52
- 3 Data assimilation for the advection-diffusion model . . . 55

Bibliographie **59**

What is the program of the course?

CM 1	○ Course presentation ① From weather forecast to engineer applliccation **Homework A1: Exo 1.1 "What time is it?"** ② Generic cost function **Homework A2: Exo 2.1 "How will the bore propagate?"**
CM2	③ Time dependent models ④ Application projects **Homework B: Section 4.1 "Lorenz model"**
TDM 1	④ Answers to questions: "Lorenz model" project
TDM 2	④ Answers to questions: "Lorenz model" project
Exam	○ **Presentation of Lorenz project by the groups**
Exam	○ **Presentation of Lorenz project by the groups**

How will the course be evaluated?

Two reports to put on Moodle before deadlines

1. Homework report A:Two simple examples (Exos 1.1 and 2.1)
2. Homework report B: Data assimilation for the Lorenz model (Section 4.1)

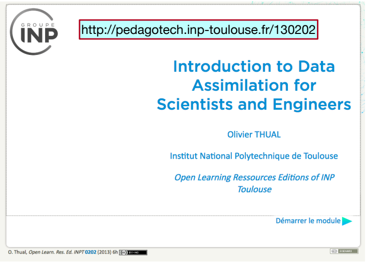
Homework report A: Two simple examples (individual 40%)

- Follow exercices 1.1 and 2.1
- About 6 pages of results
- Sources of the modified programs

Homework report B: Data assimilation - Lorenz model (trinomial 60%)

- Reproduce Section 4.1 and go beyond with Chapter 3
- About 15 pages of results
- Sources of the developed programs

Homework A: Two simple exercices (1/2)


http://pedagotech.inp-toulouse.fr/130202

Introduction to Data Assimilation for Scientists and Engineers

Olivier THUAL

Institut National Polytechnique de Toulouse

Open Learning Ressources Editions of INP Toulouse

Démarrer le module

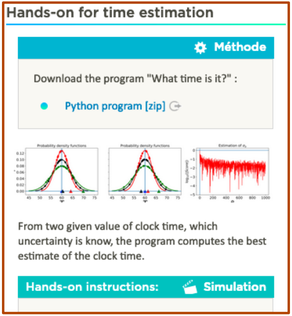
O. Thual, *Open Learn. Res. Ed. INPT* **0202** (2013) 6h

Exos 1.1 and 2.1

1. What time is it?
2. How will the bore propagate?

Link to the ressource:

Available from Moodle or web


Hands-on for time estimation

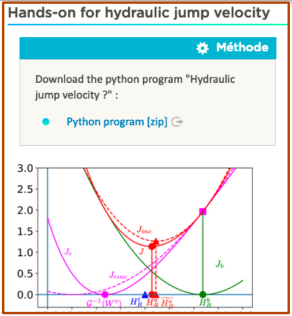
Méthode

Download the program "What time is it?" :

- Python program [zip]

From two given value of clock time, which uncertainty is know, the program computes the best estimate of the clock time.

Hands-on instructions: **Simulation**


Hands-on for hydraulic jump velocity

Méthode

Download the python program "Hydraulic jump velocity ?" :

- Python program [zip]

Homework A: Two simple exercices (2/2)

Hands-on instructions: **Simulation**

1. Read the content of the program
2. Launch the program with the default parameters
3. Describe the program algorithm briefly
4. Change parameters and describe results
5. Replace $\mathcal{G}$ and $\mathbf{G} = \mathcal{G}'(x_b)$ by another function

Pedagogical resources

Moodle N7

All documents on ASID Moodle course :
`http://moodle-n7.inp-toulouse.fr/mod/data/view.php?id=759`

Oneline courses

- O. Thual, Introduction to Data Assimilation for Scientists and Engineers, *Open Learn. Res. Ed. INPT*, **0202** (2013) 6h
  `http://pedagotech.inp-toulouse.fr/130202`
- O. Pannekoucke, Introduction to data assimilation, *Open Learn. Res. Ed. INPT*, **0831** (2013) 6h
  `http://pedagotech.inp-toulouse.fr/130831`
- *Open Learn. Res. Ed. INPT*, **0831** (2013) 6h S. Gratton and Ph. Toint, Numerical methods for Data Assimilation, *Open Learn. Res. Ed. INPT*, **0826** (2013) 6h
  `http://pedagotech.inp-toulouse.fr/130826`

Outline of the slides

1. Data assimilation for weater forecast (from Chapter 1)

General presentation of data assimilation on geophysical examples: meterology, oceanography or hydrology. General formalism.

2. How will the bore propagate? (from Chapter 2)

Minimization of the cost function on a very simple example. Incremental and ensemble methods.
Generalization of these two methods to any dimensions.

3. The Lorenz model (from Chapter 4)

Comparing four data assimilation methods on a chaotic dynamical system.

4. Will the water overflow? (from Chapter 3)

Observation operator including a temporal model. Application to a simple model.

Data assimilation for engineers

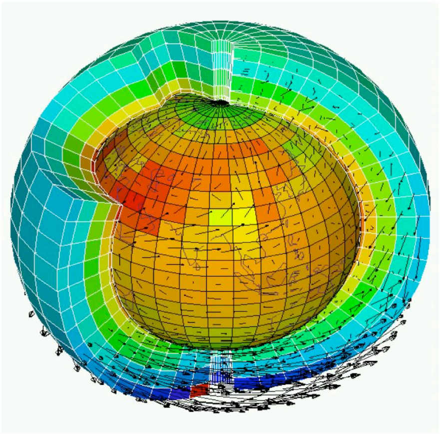
Chapter 1: From weather forecast to engineer applications

Olivier THUAL
Toulouse INP-ENSEEIHT/MFEE

Course:
Data Assimilation (ASID)

Teaching Unit:
Numerical Methods for Scientific Computing in Aerodynamics

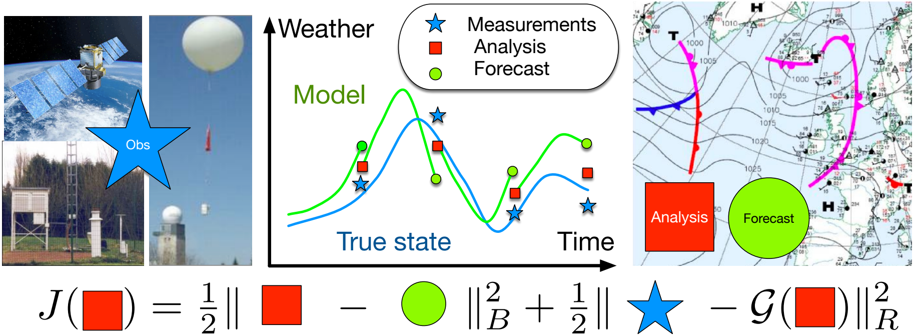
How is weather forecasted?



Atmospheric model:

Fluid mechanics equations for winds, pressure, temperature and humidity.

Data assimilation chain : a minimization problem

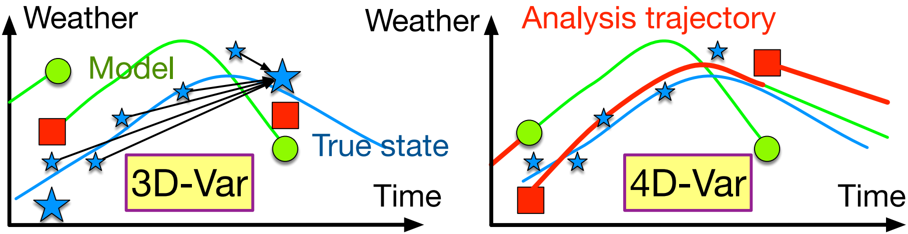


Looking for the present weather : the analysis

A state close to both the previous forecast and the field measurments

Two century of data assimilation

- XVIIIth: planet orbit computations by Gauss and least square method by Legendre.
- XXth: concept of maximum likelyhood by Fisher, Kalman filter for the APOLLO program and objective analysis of meteorological fields.
- End of XXth: 3D-Var data assimilation for weather forecast model.

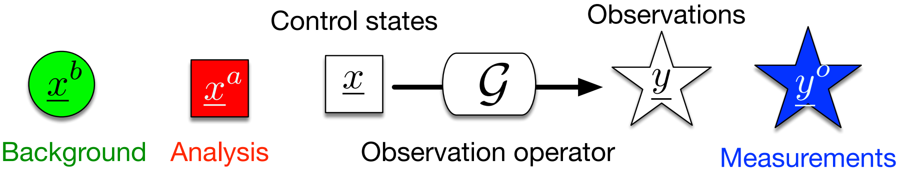


- XXIth: gain of 20% forecast quality at Météo-France with 4D-Var

The standard formalism of data assimilation

The analysis minimizes a cost function

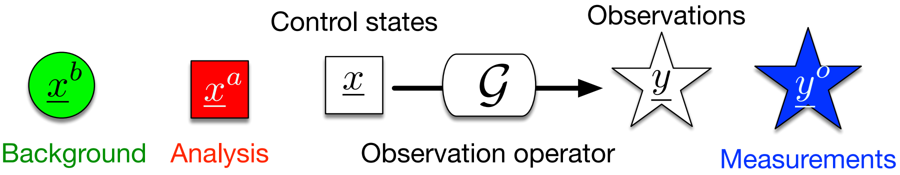
$$J(\underline{x}) = \frac{1}{2}\left(\underline{x} - \underline{x}^b\right)^T \underline{\underline{B}}^{-1}\left(\underline{x} - \underline{x}^b\right) + \frac{1}{2}\left[\underline{y}^o - \mathcal{G}(\underline{x})\right]^T \underline{\underline{R}}^{-1}\left[\underline{y}^o - \mathcal{G}(\underline{x})\right]$$



The metrics depends on the uncertainties

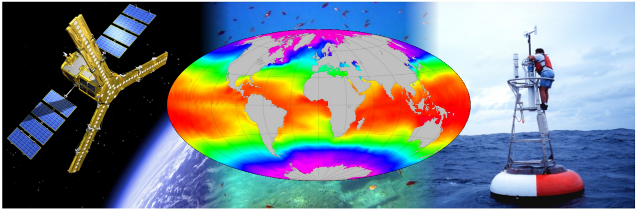
- The $N \times N$ "background error covariance matrix" $\underline{\underline{B}}$
- The $M \times M$ "observation error covariance matrix" $\underline{\underline{R}}$

Data assimilation for meteorology



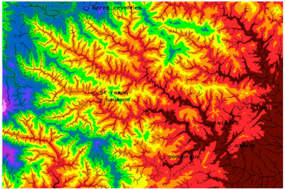
Control space	Observation operator	Observation space
• 3D fields: temperature, pressure, humidity, winds • Order ten millions of grid points	Evolution model of the primitive equations of the atmosphere	• Satellite data: surface temperatures, radiances, cloud cover... • Order one million of observations

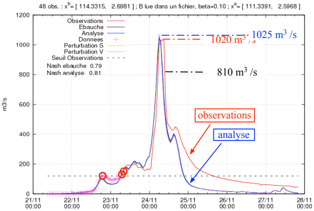
Data assimilation for oceanography



Control space	Observation operator	Observation space
• 3D fields: temperature, pressure, salinity, currents • 2D fields: sea surface level • Order ten millions of grid points	Evolution model of the primitive equations of the ocean	• Satellite data: sea surface temperature, altimetry • In-situ data: temperature, salinity... • Order thousands of observations

Data assimilation for hydrology

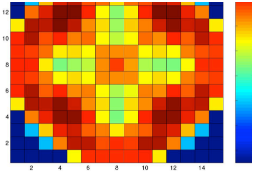


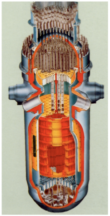


Control space	Observation operator	Observation space
• 1D fields: water height, velocity, temperature • Model parameters: friction, soil water content... • Order thousands of grid points	Shallow water (Saint-Venant) equations	• Satelitte data: altimetry • In-situ data: water height, piezometric height • Order hundred observations

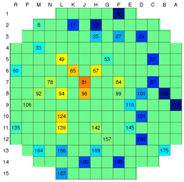
Data assimilation for nuclear cores

Neutronic flux



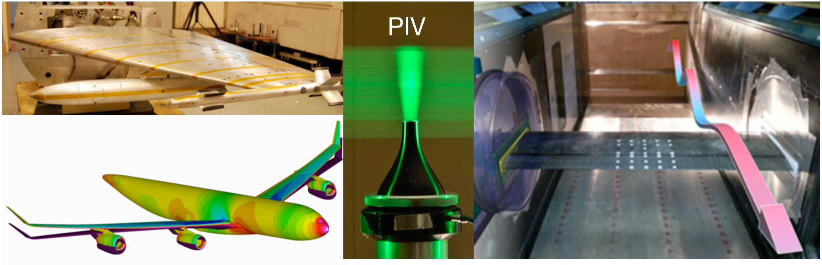


Activity measurements



Control space	Observation operator	Observation space
• 3D fields: neutronic flux, temperature, chemical concentration • Parameters: boundary reflections, diffusion coefficients • Order ten millions of grid points	Equilibrium model of the neutronic flux interacting with the heat transport fluid	• Neutronic and thermo-hydraulic sensors • Thousand of observations

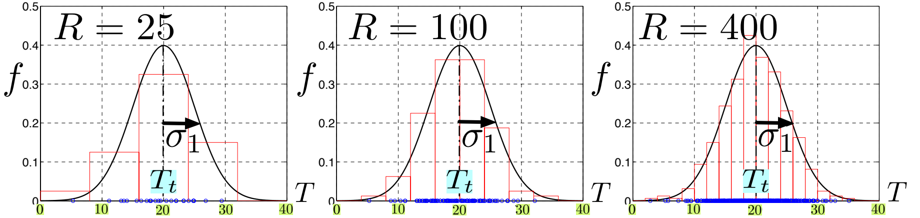
Data assimilation for aerodynamics



Control space	Observation operator	Observation space
• 3D fields: velocity, pressure • Model parameters: wing friction • Order millions of grid points	Compressible Navier-Stokes equations	• Aerodynamic flume measurements • PIV, pressure • Order hundred observations

Gaussian random variable

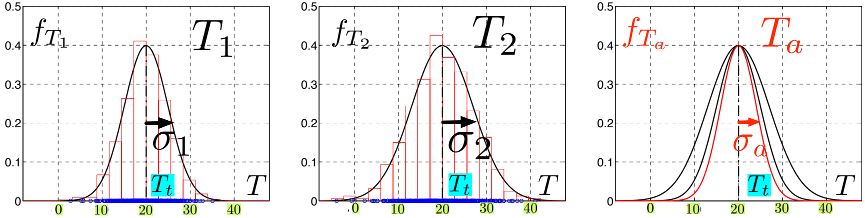
$$f_{T_1}(x) = \frac{1}{\sqrt{2\pi}\,\sigma_1} \exp\left[-\frac{(T - T_t)^2}{2\,\sigma_1^2}\right] \implies \langle \Phi(x) \rangle = \int_{-\infty}^{\infty} \Phi(x)\, f_{T_1}(x)\, dx$$



Mean: $$T_t = \langle T_1 \rangle = \int_{\mathbb{R}} T\, f_{T_1}(T)\, dT \sim \frac{1}{R} \sum_{r=1}^{R} T_1^{(r)}$$

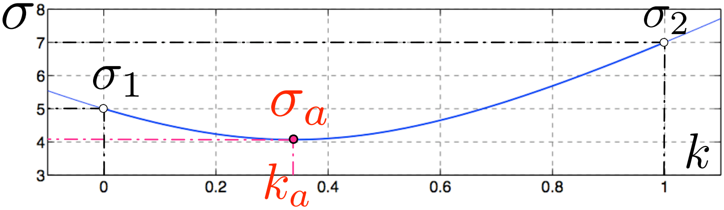
Variance: $$\sigma_1^2 = \left\langle T_1'^2 \right\rangle = \int_{\mathbb{R}} (T - T_t)^2\, f_{T_1}(T)\, dT$$

Analysis with two uncorrelated variables



New random variable $T_a = (1-k)\,T_1 + k\,T_2$ **with** $< T_1' \; T_2' >= 0$

Best choice: $T_a = \dfrac{C_1 \; T_1 + C_2 \; T_2}{C_1 + C_2}$ with $C_1 = \dfrac{1}{\sigma_1^2}$, $C_2 = \dfrac{1}{\sigma_2^2}$



$$C_a = \frac{1}{\sigma_a^2}$$

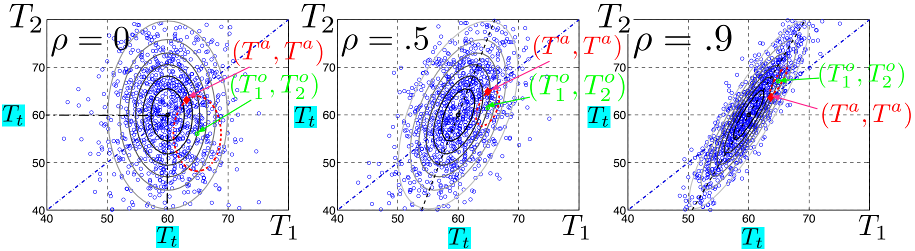
$$C_a = C_1 + C_2$$

Analysis with correlated variables

Analysis when $< T_1' \, T_2' >= \rho \, \sigma_1 \, \sigma_2$

The analysis $T_a = (1-k) T_1^o + k T_2^o$ minimizes the cost function:

$$J(T) = \frac{1}{2} \left(T_1^o - T, \quad T_2^o - T \right) \begin{pmatrix} \sigma_1^2 & \rho \, \sigma_1 \, \sigma_2 \\ \rho \, \sigma_1 \, \sigma_2 & \sigma_2^2 \end{pmatrix}^{-1} \begin{pmatrix} T_1^o - T \\ T_2^o - T \end{pmatrix}$$

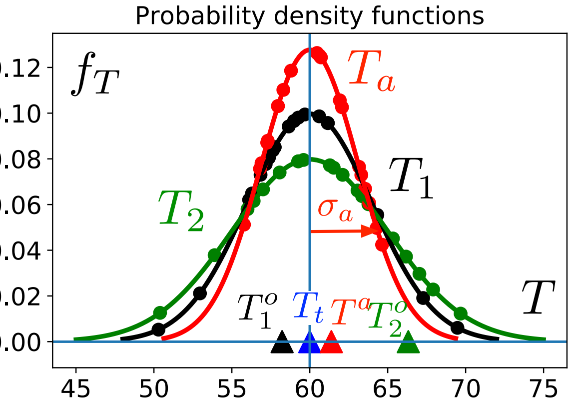


Link with the general formalism of data assimilation

$\underline{x} = (T) \in \boldsymbol{R}$, $\underline{y}^o = (T_1^o, T_2^o)^T \in \boldsymbol{R}^2$ and $\mathcal{G}(\underline{x}) = (T, T)^T \in \boldsymbol{R}^2$.
There is no background and the $2\times$ 2 matrix is $\underline{\underline{R}}$.

Homework A1: "What time is is?", Exo 1.1

http://pedagotech.inp-toulouse.fr/130202



1. Read program.
2. Launch program
3. Desactivate figures
4. Plot σ_a function of R
5. Count improvements

Data assimilation for engineers

Chapter 2: Generic cost function

Olivier THUAL
Toulouse INP-ENSEEIHT/MFEE

Course:
Data Assimilation (ASID)

Teaching Unit:
Numerical Methods for Scientific Computing in Aerodynamics

Basic linear algebra

Vectors are $1 \times N$ or $1 \times M$ matrices

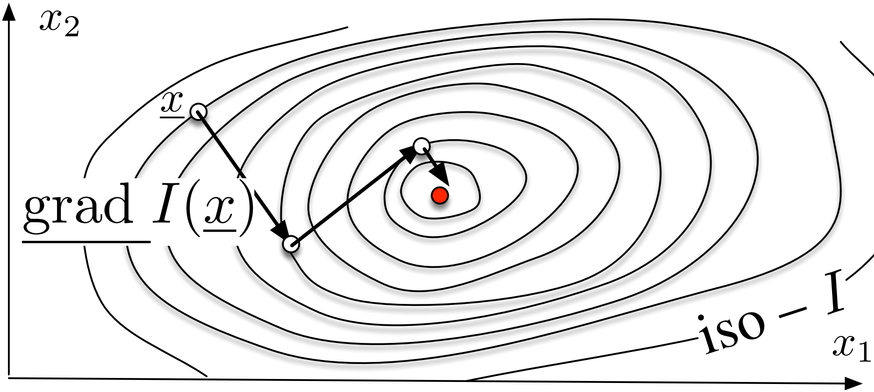
$$\underline{x} = \begin{pmatrix} x_1 \\ \dots \\ x_j \\ \dots \\ x_N \end{pmatrix} , \quad \underline{y} = \begin{pmatrix} y_1 \\ \dots \\ y_i \\ \dots \\ y_M \end{pmatrix} , \quad \begin{cases} \underline{x}^T = (x_1, \dots, x_j, \dots, x_N) \\ \\ \underline{y}^T = (y_1, \dots, y_i, \dots, y_M) \end{cases}$$

Example of a $M \times N$ matrix considered as a linear operator:

$$\underline{y} = \underline{\underline{H}}\, \underline{x} \quad \Longleftrightarrow \quad \begin{pmatrix} y_1 \\ \dots \\ y_i \\ \dots \\ y_M \end{pmatrix} = \begin{pmatrix} H_{11} & \dots & H_{1j} & \dots & H_{1N} \\ \dots & \dots & \dots & \dots & \dots \\ H_{i1} & \dots & H_{ij} & \dots & H_{iN} \\ \dots & \dots & \dots & \dots & \dots \\ H_{M1} & \dots & H_{Mj} & \dots & H_{MN} \end{pmatrix} \begin{pmatrix} x_1 \\ \dots \\ x_j \\ \dots \\ x_N \end{pmatrix}$$

Gradient of a scalar function

$$\underline{\text{grad}}\ I(\underline{x}) = \left(\frac{\partial I}{\partial x_1},\ \frac{\partial I}{\partial x_2},\ \dots \frac{\partial I}{\partial x_j},\ \dots . \frac{\partial I}{\partial x_N} \right)^T$$



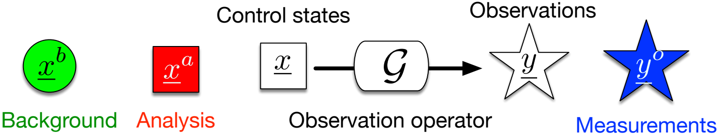
Examples of gradient computations

$\mathbf{I}(\underline{\mathbf{x}})$	$\underline{\text{grad}}\ \mathbf{I}(\underline{\mathbf{x}})$
$\underline{u}^T\,\underline{x}$	$\underline{u}$
$\underline{u}^T\,\underline{\underline{M}}\,\underline{x}$	$\underline{\underline{M}}^T\,\underline{u}$
$\underline{x}^T\,\underline{\underline{M}}\,\underline{x}$	$(\underline{\underline{M}} + \underline{\underline{M}}^T)\,\underline{x}$
$\underline{x}^T\,\underline{\underline{H}}^T\,\underline{\underline{S}}\,\underline{\underline{H}}\,\underline{x}$	$\underline{\underline{H}}\,(\underline{\underline{S}} + \underline{\underline{S}}^T)\,\underline{\underline{H}}^T\,\underline{x}$

Generic cost function for data assimilation

Searching the analysis $\underline{x}^a$ that minimize the cost function

$$J(\underline{x}) = \frac{1}{2}\left(\underline{x} - \underline{x}^b\right)^T \underline{\underline{B}}^{-1}\left(\underline{x} - \underline{x}^b\right) + \frac{1}{2}\left[\underline{y}^o - \mathcal{G}(\underline{x})\right]^T \underline{\underline{R}}^{-1}\left[\underline{y}^o - \mathcal{G}(\underline{x})\right]$$



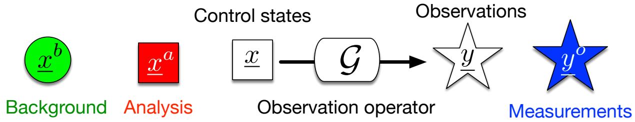
- Control space $\underline{x} \in \boldsymbol{R}^N$ and Observation space $\underline{y} \in \boldsymbol{R}^M$
- Background $\underline{x}^b$ known with errors $\underline{\epsilon}^b$
- Measurement $\underline{y}^o$ known with errors $\underline{\epsilon}^o$
- Covariance background error matrix $\underline{\underline{B}}$ with $B_{ij} = \left\langle \epsilon_i^b \, \epsilon_j^b \right\rangle$
- Covariance observation error matrix $\underline{\underline{R}}$ with $R_{ij} = \left\langle \epsilon_i^o \, \epsilon_j^o \right\rangle$

Bayesian approach of data assimilation

Density probability function for $\underline{x}^b$ and $\underline{y}^o$ independantly

$$f^b(\underline{x}) = K^b\, e^{-J_b(\underline{x})} \quad \text{with} \quad J_b(\underline{x}) = \frac{1}{2}\left(\underline{x} - \underline{x}^b\right)^T \underline{\underline{B}}^{-1}\left(\underline{x} - \underline{x}^b\right)$$

$$f^o(\underline{y}) = K^o\, e^{-J_r(\underline{x})} \quad \text{with} \quad J_r(\underline{y}) = \frac{1}{2}\left(\underline{y}^o - \underline{y}\right)^T \underline{\underline{R}}^{-1}\left(\underline{y}^o - \underline{y}\right)$$



Density probability function for $\underline{x}$ knowing $\underline{y}^o$

$$f^{b/o}(\underline{x}) = f^b(\underline{x})\, f^o[\mathcal{G}(\underline{x})] = K^o\, K^b\, e^{-J(\underline{x})} \quad \text{with}$$

$$J(\underline{x}) = J_b(\underline{x}) + J_r\left[\mathcal{G}(\underline{x})\right] = J_b(\underline{x}) + \frac{1}{2}\left[\underline{y}^o - \mathcal{G}(\underline{x})\right]^T \underline{\underline{R}}^{-1}\left[\underline{y}^o - \mathcal{G}(\underline{x})\right]$$

Incremental cost function through a linearization of $\mathcal{G}$

$$J(\underline{x}) = \frac{1}{2}\left(\underline{x} - \underline{x}^b\right)^T \underline{\underline{B}}^{-1}\left(\underline{x} - \underline{x}^b\right) + \frac{1}{2}\left[\underline{y}^o - \mathcal{G}(\underline{x})\right]^T \underline{\underline{R}}^{-1}\left[\underline{y}^o - \mathcal{G}(\underline{x})\right]$$

Linearization of $\mathcal{G}$ around the background $\underline{x}^b$:

$$\mathcal{G}(\underline{x}^b + \underline{\delta x}) \;\approx\; \mathcal{G}(\underline{x}^b) + \underline{\underline{G}}\;\underline{\delta x}$$

Incremental cost function

$$J_{inc}(\underline{x}^b + \underline{\delta x}) = \frac{1}{2}\underline{\delta x}^T\;\underline{\underline{B}}^{-1}\;\underline{\delta x} + \frac{1}{2}\left(\underline{d} - \underline{\underline{G}}\,\underline{\delta x}\right)^T\;\underline{\underline{R}}^{-1}\;\left(\underline{d} - \underline{\underline{G}}\,\underline{\delta x}\right)$$

where $\underline{d} = \underline{y}^o - \mathcal{G}(\underline{x}^b)$ is the innovation vector

Minimum of the incremental cost function

Knowing the innovation vector $\underline{d} = \underline{y}^o - \mathcal{G}(\underline{x}^)$:

$$J_{inc}(\underline{x}^b + \underline{\delta x}) = \frac{1}{2}\underline{\delta x}^T \; \underline{\underline{B}}^{-1} \; \underline{\delta x} + \frac{1}{2}\left(\underline{d} - \underline{\underline{G}}\,\underline{\delta x}\right)^T \; \underline{\underline{R}}^{-1} \; \left(\underline{d} - \underline{\underline{G}}\,\underline{\delta x}\right)$$

Gradient of the incremental cost function

$$\text{gr}\underline{\text{ad}} \; J_{inc}(\underline{x}^b + \underline{\delta x}) = \underline{\underline{B}}^{-1}\,\underline{\delta x} - \underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\left[\underline{d} - \underline{\underline{G}}\,\delta x\right]$$

The minimum $\widetilde{\underline{x}}^a$ is found through $\text{gr}\underline{\text{ad}} \; J_{inc}(\underline{x}^a) = \underline{0}$:

$$\widetilde{\underline{x}}^a = \underline{x}^b + \underline{\underline{K}}\;\underline{d}$$

where the gain matrix is: $\underline{\underline{K}} = \left(\underline{\underline{B}}^{-1} + \underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}}\right)^{-1}\;\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}$

and is also equal to (SMW identity): $\underline{\underline{K}} = \underline{\underline{B}}\,\underline{\underline{G}}^T\;\left(\underline{\underline{G}}\,\underline{\underline{B}}\,\underline{\underline{G}}^T + \underline{\underline{R}}\right)^{-1}$

Sherman-Morrison-Woodbury identity

Two expressions of the gain matrix:

$$\underline{\underline{K}} = (\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}} + \underline{\underline{B}}^{-1})^{-1}\,\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1} = \underline{\underline{B}}\,\underline{\underline{G}}^T\,(\underline{\underline{G}}\,\underline{\underline{B}}\,\underline{\underline{G}}^T + \underline{\underline{R}})^{-1}$$

- If $M < N$: inverse the $M \times M$ matrix $(\underline{\underline{G}}\,\underline{\underline{B}}\,\underline{\underline{G}}^T + \underline{\underline{R}})$
- If $N < M$: inverse the $N \times N$ matrix $(\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}} + \underline{\underline{B}}^{-1})$

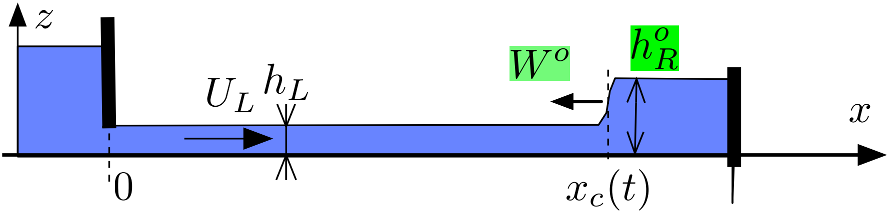
$$\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,(\underline{\underline{G}}\,\underline{\underline{B}}\,\underline{\underline{G}}^T + \underline{\underline{R}}) = \underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}}\,\underline{\underline{B}}\,\underline{\underline{G}}^T + \underline{\underline{G}}^T = (\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}}\,\underline{\underline{B}} + \underline{\underline{I}})\,\underline{\underline{G}}^T$$

$$= (\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}}\,\underline{\underline{B}} + \underline{\underline{B}}^{-1}\,\underline{\underline{B}})\,\underline{\underline{G}}^T = (\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}} + \underline{\underline{B}}^{-1})\,\underline{\underline{B}}\,\underline{\underline{G}}^T$$

$$\left(\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}} + \underline{\underline{B}}^{-1}\right)^{-1}\,\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1} = \underline{\underline{B}}\,\underline{\underline{G}}^T\,(\underline{\underline{G}}\,\underline{\underline{B}}\,\underline{\underline{G}}^T + \underline{\underline{R}})^{-1}$$

How will the bore propagate?

A very simple geophysical model



Mass conservation:

$$h_L(U_L - W) = h_R(U_R - W) \quad \Longrightarrow W = \frac{-h_L\, U_L}{h_R - h_L}$$

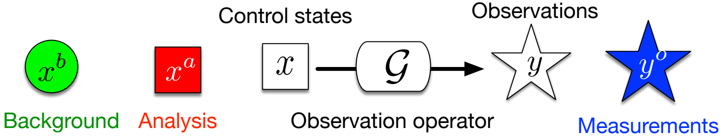
Notations $x \in \mathbf{R}$, $\mathcal{G} : \mathbf{R} \to \mathbf{R}$, $y \in \mathbf{R}$:

$$x = h_R\ , \quad y = W\ , \quad \text{and} \quad y = \mathcal{G}(x) = \frac{-h_L\, U_L}{x - h_L} = \frac{-q}{x - h_L}$$

Best estimate of the height $x = h_R$

Too much information in: $x = h_R$, $\mathcal{G}(x) = \frac{-q}{x-h_L}$, $y = W$

- We know approximatively $x = x^b$ with an incertainty error σ_b
- We know approximatively $y = y^o$ with an incertainty error σ_r
- What is the best estimate x^a of x knowing that $y = \mathcal{G}(x)$?

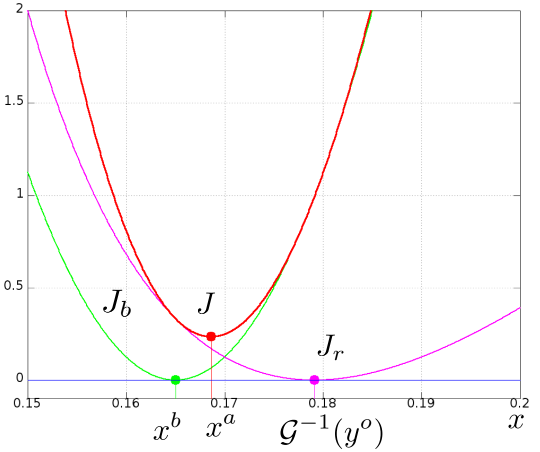


The analysis x^a minimizes the cost function:

$$J(x) = \frac{(x - x^b)^2}{2\,\sigma_b^2} + \frac{[y^o - \mathcal{G}(x)]^2}{2\,\sigma_r^2}$$

Plotting the cost function $J(x)$

$$J(x) = \frac{(x - x^b)^2}{2\,\sigma_b^2} + \frac{[y^o - \mathcal{G}(x)]^2}{2\,\sigma_r^2} = J_b(x) + J_r(x)$$



$$J_b(x) = \frac{(x - x^b)^2}{2\,\sigma_b^2}$$

$$J_r(x) = \frac{[y^o - \mathcal{G}(x)]^2}{2\,\sigma_r^2}$$

with $\quad \mathcal{G}(x) = \dfrac{-q}{x - h_L}$

Incremental cost function

The cost functin to minimize

$$J(x) = \frac{(x - x^b)^2}{2\,\sigma_r^2} + \frac{[y^o - \mathcal{G}(x)]^2}{2\,\sigma_r^2} \qquad \text{with} \qquad \mathcal{G}(x) = \frac{-q}{x - h_L}$$

Linearization of $\mathcal{G}$ around x^b:

$$\mathcal{G}(x) = \mathcal{G}(x^b + \delta x) \sim \mathcal{G}(x^b) + G\,\delta x \qquad \text{with } \delta x = x - x^b$$

One can compute $G = \mathcal{G}'(x^b) = q/(x^b - h_L)^2$

The incremental cost function J_{inc} is an approximation of J:

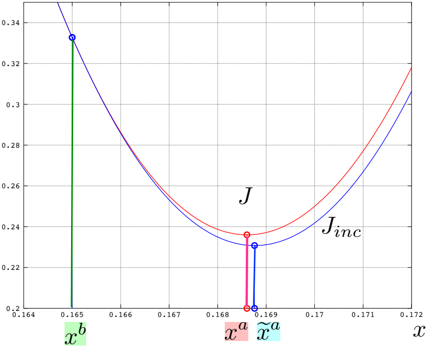
$$J_{inc}(x^b + \delta x) = \frac{(\delta x)^2}{2\,\sigma_b^2} + \frac{(d - G\,\delta x)^2}{2\,\sigma_r^2}$$

where $d = y^o - \mathcal{G}(x^b)$ is the “innovation”

Plotting the incremental cost function J_{inc}

Gradient of the function $J_{inc}(x^b + \delta x) = \frac{(\delta x)^2}{2\,\sigma_b^2} + \frac{(d - G\,\delta x)^2}{2\,\sigma_r^2}$:

$$J'_{inc}(x^b + \delta x) = \frac{\delta x}{\sigma_b^2} + G\,\frac{G\,\delta x - d}{\sigma_r^2}$$



J'_{inc} vanishes for: $\underline{\widetilde{x}}^a = \underline{x}^b + \widetilde{\delta x}^a$

with $\widetilde{\delta x}^a = K\,d$ where:

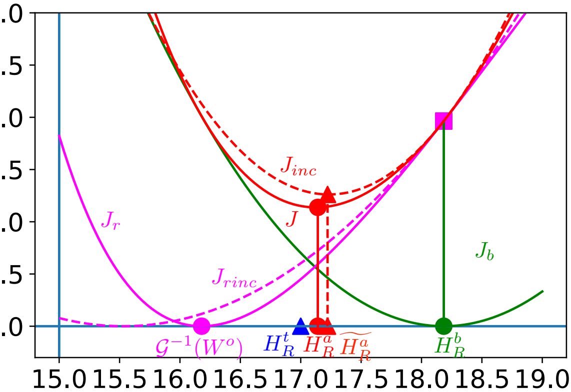
Innovation: $d = y^o - \mathcal{G}(x_b)$

Gain: $K = \left(\frac{1}{\sigma_b^2} + \frac{G^2}{\sigma_r^2}\right)^{-1} \frac{G}{\sigma_r^2}$

or $K = \sigma_b^2\,G\,\left(G^2\,\sigma_b^2 + \sigma_r^2\right)^{-1}$

Homework A2: "How will the bore propagate?", Exo 2.1

http://pedagotech.inp-toulouse.fr/130202



$$\mathcal{G}(x) = \frac{-q}{x - h_L}$$

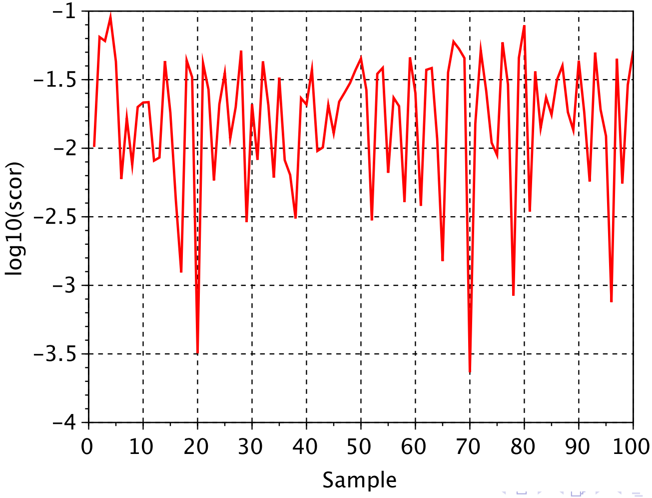
$$G = \mathcal{G}'(x^b) = \frac{q}{(x - h_L)^2}$$

1. Read program.
2. Launch program
3. Compute scores
4. Replace $\mathcal{G}$
5. Count improvements

Exo 2.1: Example of score for $\mathcal{G}(x) = -q/(x - h_L)$

Experimental values

$q = 7,\ h_L = 5,\ h_R = x^t = 17,\ x^b = 18,\ \sigma_b = 1,\ \sigma_r = .03$



Data assimilation for engineers

Chapter 3: Time dependent models

Olivier THUAL
Toulouse INP-ENSEEIHT/MFEE

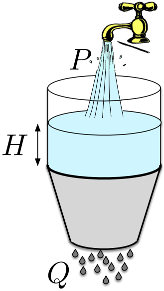
Course:
Data Assimilation (ASID)

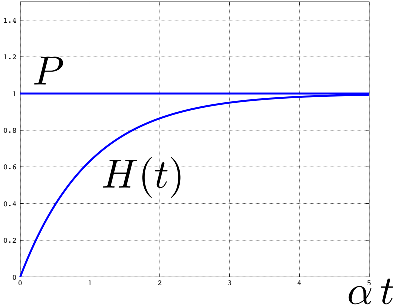
Teaching Unit:
Numerical Methods for Scientific Computing in Aerodynamics

Will the water overflow

Reservoir model

$$\frac{dH(t)}{dt} = -\alpha\, H(t) + P \qquad \text{with} \qquad H(0) = 0$$





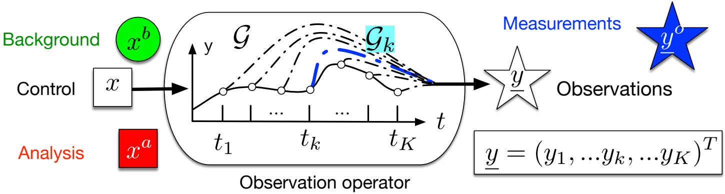
Exact solution

$$H(t) = (1/\alpha)\,[1 - \exp(-\alpha\, t)]$$

Assimilation of α

Measurements of $y_k = H(t_k)$ to determine $x = \alpha$

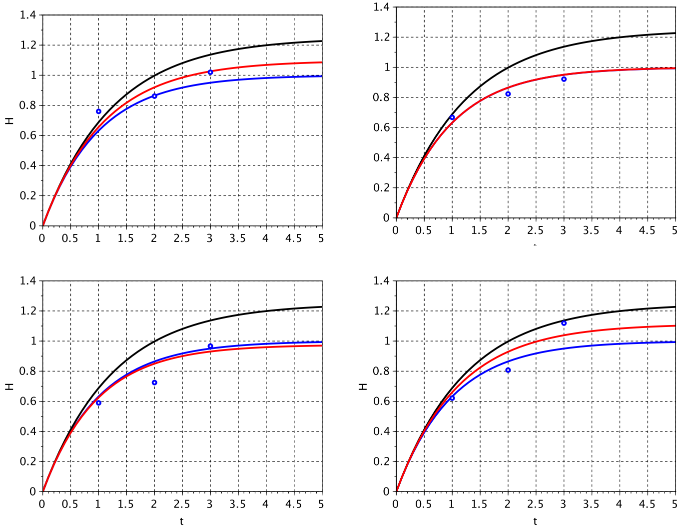
- $\underline{y}^o = (y_1^o, ...y_k^o, ...y_K^o)^T$
- $\mathcal{G} = (\mathcal{G}_1^T, ...\mathcal{G}_k^T, ...\mathcal{G}_K^T)^T$ with $\mathcal{G}_k(\alpha) = (P/\alpha)[1 - \exp(-\alpha\, t_k)]$



Cost function

$$J(\alpha) = \frac{1}{2}\frac{(\alpha - \alpha^b)^2}{\sigma_b^2} + \sum_{k=1}^{K} \frac{\left[y_k^o - \mathcal{G}_k(\alpha)^2\right]}{2\,\sigma_r^2}$$

Incremental method for the assimilation of α

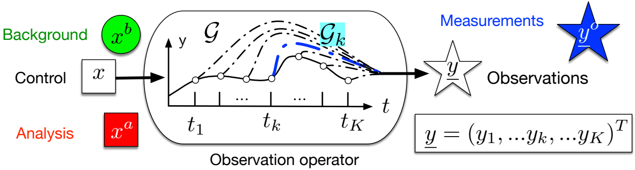


Red: analysis, Blue:true state, Black: background

Assimilation of (α, P)

Measurements of $y_k = H(t_k)$ to determine $\underline{x} = (\alpha, P)$

- $\underline{y}^o = (y_1^o, ... y_k^o, ... y_K^o)^T$
- $\mathcal{G} = (\mathcal{G}_1^T, ... \mathcal{G}_k^T, ... \mathcal{G}_K^T)^T$ with $\mathcal{G}_k(\alpha, P) = (P/\alpha)[1 - \exp(-\alpha\, t_k)]$



Cost function:

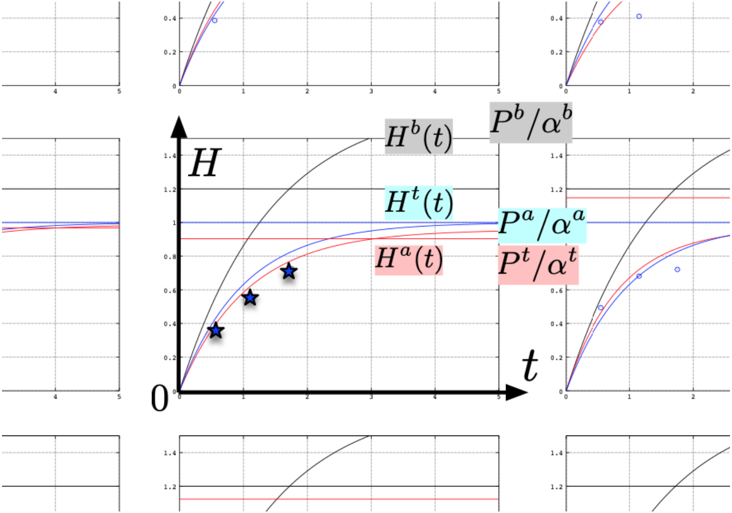
$$J(\alpha, P) = \frac{1}{2}(\alpha - \alpha^b, P - P^b)\, \underline{\underline{B}}^{-1} \begin{pmatrix} \alpha - \alpha^b \\ P - P^b \end{pmatrix}$$

$$+ \sum_{k=1}^{K} \frac{[y_k^o - \mathcal{G}_k(\alpha, P)]^2}{2\,\sigma_r^2}$$

Covariance background error matrix:

$$\underline{\underline{B}} = \begin{pmatrix} \sigma_\alpha^2 & \rho\,\sigma_\alpha\,\sigma_P \\ \rho\,\sigma_\alpha\,\sigma_P & \sigma_P^2 \end{pmatrix}$$

Incremental method for the assimilation of (α, P)



Red: analysis, Blue:true state, Black: background

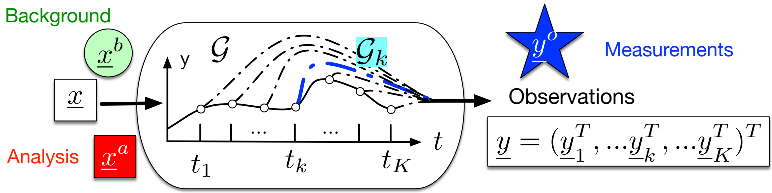
The 4D-Var data assimilation

$\underline{x} \in \mathbf{R}^N$ with measurments $\underline{y}^o \in \mathbf{R}^M$

- $\underline{y} = (\underline{y}_1^T, \dots \underline{y}_k^T, \dots \underline{y}_K^T)^T$ with $\underline{y}_k = \mathcal{G}_k(\underline{x})$
- $\underline{y}^o = \left(\underline{y}_1^{oT}, \dots \underline{y}_k^{oT}, \dots \underline{y}_K^{oT}\right)^T$

Observation operator:

$$\mathcal{G} = (\mathcal{G}_1^T, \dots \mathcal{G}_k^T, \dots \mathcal{G}_K^T)^T$$



$$J(\underline{x}) = \frac{1}{2}\left(\underline{x} - \underline{x}^b\right)^T \underline{\underline{B}}^{-1} \left(\underline{x} - \underline{x}^b\right)$$

$$+ \frac{1}{2} \sum_{k=1}^{K} \left[\underline{y}_k^o - \mathcal{G}_k(\underline{x})\right]^T \underline{\underline{R}}_k^{-1} \left[\underline{y}_k^o - \mathcal{G}_k(\underline{x})\right]$$

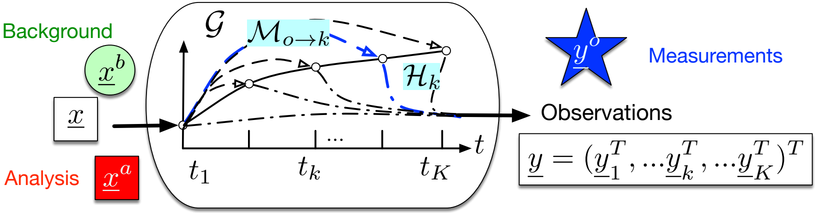
Covariance observation error matrix:

$$\underline{\underline{R}} = \text{diag}\ (\underline{\underline{R}}_1, \dots \underline{\underline{R}}_k, \dots \underline{\underline{R}}_K)$$

Nonlinear Cauchy problem

Dynamical system

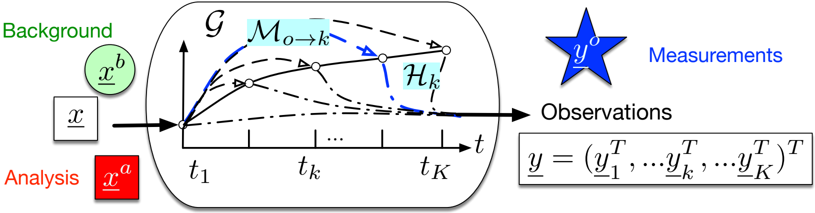
$$\frac{d\underline{X}}{dt} = \mathcal{L}(t, \underline{X}) \quad \text{with} \quad \underline{X}(0) = \underline{x}$$



Control of the initial conditions $\underline{x} = \underline{X}(0)$:

- $\underline{X}(t_k) = \mathcal{M}_{l\to k}\left[\underline{X}(t_l)\right] \quad \Longrightarrow \quad \underline{X}(t_k) = \mathcal{M}_{0\to k}(\underline{x})$.
- $\underline{y}_k = \mathcal{H}_k\, \mathcal{M}_{0\to k}(\underline{x})$ with $\mathcal{H}_k$ nonlinear or linear
- $\mathcal{G} = \left(\mathcal{G}_1^T, \ldots \mathcal{G}_k^T, \ldots \mathcal{G}_K^T\right)^T$ with $\mathcal{G}_k = \mathcal{H}_k\, \mathcal{M}_{0\to k}$

The generic 4D-Var cost function



Assuming $\underline{\underline{R}} = \text{diag}\ \left(\underline{\underline{R}}_1, \dots \underline{\underline{R}}_k, \dots \underline{\underline{R}}_K\right)$

$$J(\underline{x}) = \frac{1}{2}\left(\underline{x} - \underline{x}^b\right)^T \underline{\underline{B}}^{-1}\left(\underline{x} - \underline{x}^b\right)$$

$$+\frac{1}{2}\sum_{k=1}^{K}\left[\underline{y}_k^o - \mathcal{H}_k\,\mathcal{M}_{0\to k}(\underline{x})\right]^T \underline{\underline{R}}_k^{-1}\left[\underline{y}_k^o - \mathcal{H}_k\,\mathcal{M}_{0\to k}(\underline{x})\right].$$

Linear tangent model

Linearization around the background vector $\underline{x}^b$:

$$\mathcal{H}_k\, \mathcal{M}_{0\to k}(\underline{x}^b + \underline{\delta x}) \sim \mathcal{H}_k\, \mathcal{M}_{0\to k}(\underline{x}^b) + \underline{\underline{H}}_k\, \underline{\underline{M}}_{0\to k}\, \underline{\delta x}$$

- $\underline{\underline{H}}_k$ is the linearized of $\mathcal{H}_k$ around $\mathcal{M}_{0\to k}(\underline{x}^b)$
- $\underline{\underline{M}}_{0\to k}$ is the linearized of $\mathcal{M}_{0\to k}$ around $\underline{x}^b$

Linear tangent dynamical system and tangent model $\underline{\underline{M}}_{0\to k}$

$$\frac{\partial \underline{U}}{\partial t} = \underline{\underline{L}}\left[\underline{X}^b(t)\right]\, \underline{U} \quad \text{with} \quad \underline{U}(0) = \underline{\delta x}$$

- $\underline{X}^b(t)$ is the solution of $\frac{d\underline{X}}{dt} = \mathcal{L}(t, \underline{X})$ with $\underline{X}^b(0) = \underline{x}^b$
- $\underline{\underline{L}}$ is such that $\mathcal{L}[\underline{X}^b(t) + \underline{U}] \sim \mathcal{L}[\underline{X}^b(t)] + \underline{\underline{L}}\left[\underline{X}^b(t)\right]\, \underline{U}$
- $\underline{U}(t_k) = \underline{\underline{M}}_{l\to k}\, \underline{U}(t_l)$ with $\underline{\underline{M}}_{l\to k} = \exp\left\{\int_{t_l}^{t_k} \underline{\underline{L}}\left[\underline{X}^b(t)\right]\, dt\right\}$
- $\underline{U}(t_k) = \underline{\underline{M}}_{0\to k}\, \underline{\delta x}$ with $\underline{\underline{M}}_{0\to k} = \exp\left\{\int_{0}^{t_k} \underline{\underline{L}}\left[\underline{X}^b(t)\right]\, dt\right\}$

The incremental 4D-Var function

Incremental cost function with $\underline{\underline{R}} = \text{diag}\ (\underline{\underline{R}}_1, ...\underline{\underline{R}}_k, ...\underline{\underline{R}}_K)$:

$$J_{inc}(\underline{x}^b + \underline{\delta x}) = \frac{1}{2}\underline{\delta x}^T \underline{\underline{B}}^{-1}\underline{\delta x}$$

$$+\frac{1}{2}\sum_{k=1}^{K}\left[\underline{d}_k - \underline{\underline{H}}_k\,\underline{\underline{M}}_{0\to k}\,\underline{\delta x}\right]^T \underline{\underline{R}}_k^{-1}\left[\underline{d}_k - \underline{\underline{H}}_k\,\underline{\underline{M}}_{0\to k}\,\underline{\delta x}\right]$$

where $\underline{d}_k = \underline{y}_k^o - \mathcal{H}_k\,\mathcal{M}_{0\to k}(\underline{x}^b)$ are the innovation subvectors

- $\mathcal{G} = \left(\mathcal{G}_1^T, ...\mathcal{G}_k^T, ...\mathcal{G}_K^T\right)^T$ with $\mathcal{G}_k = \mathcal{H}_k\,\mathcal{M}_{0\to k}$
- Innovation vector: $\underline{d} = \underline{y}^o - \mathcal{G}(\underline{x})$
- $\underline{\underline{G}} = \text{diag}\left(\underline{\underline{H}}_1\,\underline{\underline{M}}_{0\to 1}, ...\underline{\underline{H}}_k\,\underline{\underline{M}}_{0\to k}, ...\underline{\underline{H}}_K\,\underline{\underline{M}}_{0\to K}\right)$

$$\underline{\widetilde{x}}^a = \underline{x}^b + \underline{\underline{K}}\,\underline{d} \quad \text{with} \quad \underline{\underline{K}} = \left(\underline{\underline{B}}^{-1} + \underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}}\right)^{-1}\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}$$

Adjoint model $\underline{\underline{M}}^*_{k\to 0} = \underline{\underline{M}}^T_{0\to k}$

- $\underline{\underline{K}} = \left(\underline{\underline{B}}^{-1} + \underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}\,\underline{\underline{G}}\right)^{-1}\,\underline{\underline{G}}^T\,\underline{\underline{R}}^{-1}$
- $\underline{\underline{G}} = \mathrm{diag}\left(\underline{\underline{H}}_1\,\underline{\underline{M}}_{0\to 1}, \dots \underline{\underline{H}}_k\,\underline{\underline{M}}_{0\to k}, \dots \underline{\underline{H}}_K\,\underline{\underline{M}}_{0\to K}\right)$
- $\underline{\underline{G}}^T = \mathrm{diag}\left(\underline{\underline{M}}^T_{0\to 1}\,\underline{\underline{H}}^T_1, \dots \underline{\underline{M}}^T_{0\to k}\,\underline{\underline{H}}^Y_k, \dots \underline{\underline{M}}^T_{0\to K}\,\underline{\underline{H}}^Y_K\right)$

Definition of the adjoint model $\underline{\underline{M}}^*$

New Cauchy problem: $\dfrac{\partial \underline{\widetilde{X}}}{\partial \tau} = \underline{\underline{L}}^T\left[\underline{X}^b(t_K - \tau)\right]\,\underline{\widetilde{X}}$ with $\underline{\widetilde{X}}(0) = \underline{\widetilde{x}}$

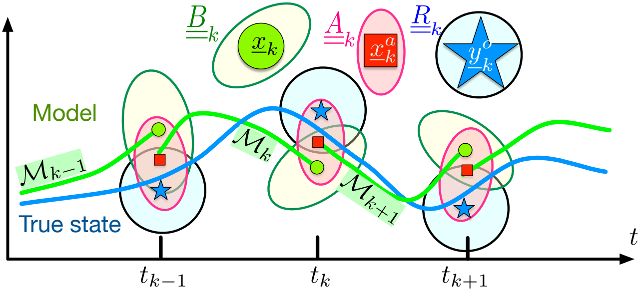
$$\underline{\widetilde{X}}(\tau_l) = \underline{\underline{M}}^*_{k\to l}\,\underline{\widetilde{X}}(\tau_k) \quad \text{with} \quad \underline{\underline{M}}^*_{k\to l} = \exp\left\{\int_{\tau_k}^{\tau_l} \underline{\underline{L}}^T\left[\underline{X}^b(t_K - \tau)\right]\,d\tau\right\}$$

$$\underline{\underline{M}}_{l\to k} = \exp\left\{\int_{\tau_l}^{\tau_k} \underline{\underline{L}}\left[\underline{X}^b(t)\right]\,dt\right\} \Longrightarrow \underline{\underline{M}}^*_{k\to l} = \underline{\underline{M}}^T_{l\to k} \;\text{ et }\; \underline{\underline{M}}^*_{k\to 0} = \underline{\underline{M}}^T_{0\to k}$$

Extended Kalman filter

Initialization:

- $\underline{x}_0 = \underline{x}$
- $\underline{\underline{A}}_0 = \underline{\underline{B}}$.



We denote $\mathcal{M}_k = \mathcal{M}_{k-1\to k}$ and $\underline{\underline{M}}_k = \underline{\underline{M}}_{k-1\to k}$:

$$\begin{aligned} \underline{x}_k &= \mathcal{M}_k\left(\underline{x}^a_{k-1}\right) \quad \text{and} \quad \underline{\underline{B}}_k = \underline{\underline{M}}_k\,\underline{\underline{A}}_{k-1}\,\underline{\underline{M}}_k^T + \underline{\underline{Q}}_k \\ \underline{x}^a_k &= \underline{x}_k + \underline{\underline{K}}_k\,\underline{d}_k \quad \text{with} \\ \underline{d}_k &= \underline{y}^o_k - \mathcal{H}_k\left(\underline{x}_k\right) \quad \text{and} \quad \underline{\underline{K}}_k = \underline{\underline{B}}_k\,\underline{\underline{H}}_k^T\left(\underline{\underline{H}}_k\,\underline{\underline{B}}_k\,\underline{\underline{H}}_k + \underline{\underline{R}}_k\right)^{-1}, \\ \underline{\underline{A}}_k &= \left(\underline{\underline{I}} - \underline{\underline{K}}_k\,\underline{\underline{H}}_k\right)\underline{\underline{B}}_k \end{aligned}$$

where $\underline{\underline{Q}}_k$ is the covariance model error matrix.

Data assimilation for engineers

Chapter 4: Application projects

Olivier THUAL
Toulouse INP-ENSEEIHT/MFEE

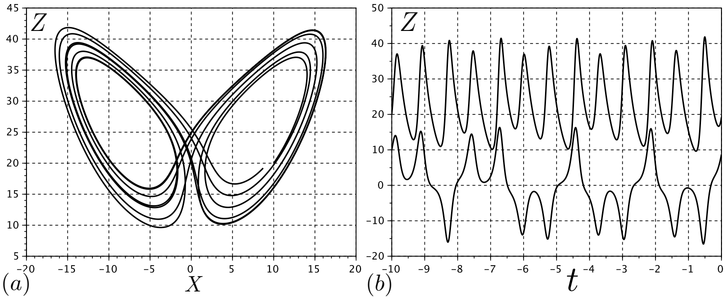
Course:
Data Assimilation (ASID)

Teaching Unit:
Numerical Methods for Scientific Computing in Aerodynamics

Project: data assimilation for the Lorenz model

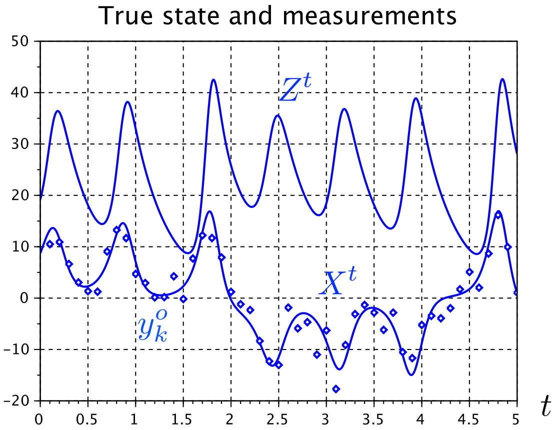
The Lorenz model: $\sigma = 10$, $\rho = 28$ and $\beta = 8/3$

$$\frac{dX}{dt} = -\sigma X + \sigma Y \,, \quad \frac{dY}{dt} = \rho X - Y - X Z \,, \quad \frac{dZ}{dt} = X Y - \beta Z$$



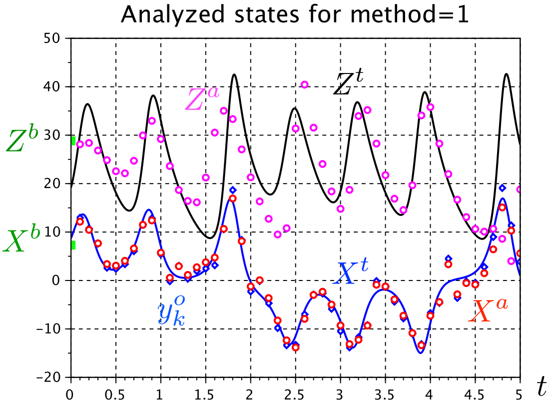
Transitory: $t \in [-10, 0]$ with $(X_s, Y_s, Z_s) = (10, 15, 20)$ for $t = -10$

Twin experiments



- True trajectory: $[X^t(t), Y^t(t), Z^t(t)]$ for $t \in [0, t_f]$ with $t_f = 5$
- Background: $(X^b, Y^b, Z^b) = (X_0, Y_0, Z_0) + (\epsilon_X, \epsilon_Y, \epsilon_Z)$
- Measurements: $y_k^o = X^t(t_k) + \epsilon^o$ for $k = 1, ..., 50$, where $t_k = k\,\tau$

Chain of BLUE (3D-Var) with $\underline{B}$ constant



Let $\underline{\underline{B}} = \sigma_b^2 \, \underline{\underline{I}}$ constant:

$$J_k(\underline{x}) =$$

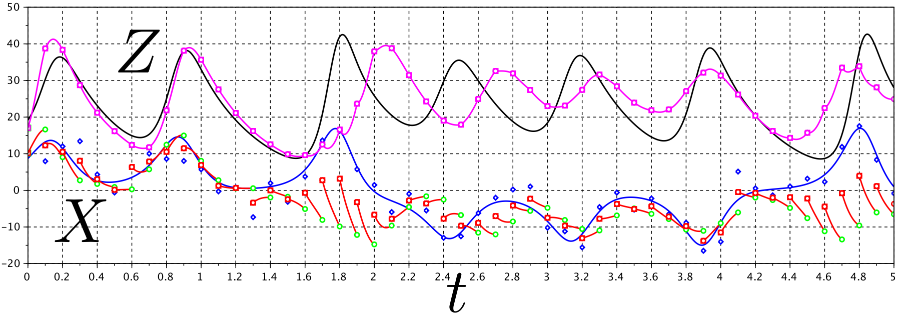
$$\frac{1}{2}(\underline{x} - \underline{x}_k^b)^T \, \underline{\underline{B}}^{-1} \, (\underline{x} - \underline{x}_k^b)^T$$

$$+ \frac{1}{2\,\sigma_r^2}(y_k^o - X)^2$$

$$\underline{x}_k^a = \underline{x}_k^b + \underline{\underline{K}} \, d_k = \left[X_k^b + \frac{\sigma_b^2}{\sigma_b^2 + \sigma_r^2}(y_k^o - X_k^b), \, Y_k^b, \, Z_k^b \right]^T$$

where $d_k = y_k^o - \underline{G} \, \underline{x}_k^b = y_k^o - X_k^b$ is the innovation

Chain of BLUE (3D-Var) with $\underline{\underline{B}} = \underline{\underline{A}}$



$$J(\underline{x}) = \frac{1}{2}(\underline{x} - \underline{x}_k^b)^T \, \underline{\underline{A}}_{k-1}^{-1} \, (\underline{x} - \underline{x}_k^b)^T + \frac{1}{2\,\sigma_r^2}(y_k^o - X)^2$$

$$\underline{\underline{A}}_k = (\underline{\underline{I}} - \underline{\underline{K}}\,\underline{G})\,\underline{\underline{A}}_{k-1} \quad \text{for } k = 1, ..., K \text{ with } \quad \underline{\underline{A}}_0 = \underline{\underline{B}}$$

Kalman filter with no error model

$$J(\underline{x}) = \frac{1}{2}(\underline{x} - \underline{x}_k^b)^T \,\underline{\underline{B}}_k^{-1}\, (\underline{x} - \underline{x}_k^b)^T + \frac{1}{2\,\sigma_r^2}(y_k^o - X)^2 \;,$$

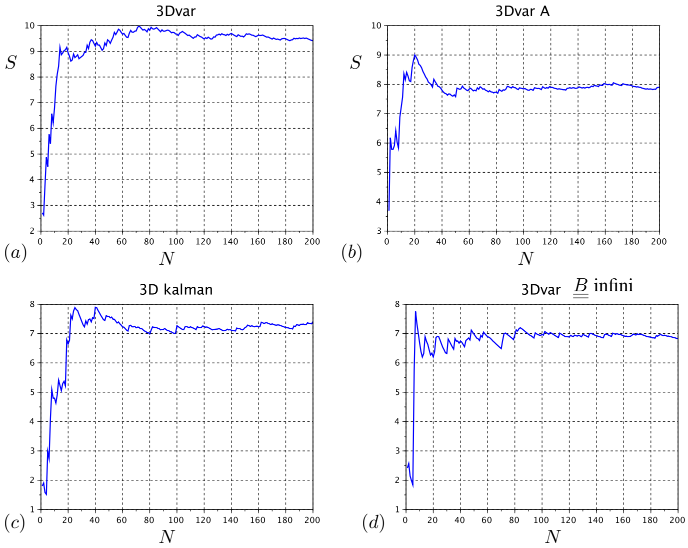
$$\underline{\underline{A}}_{k-1} = (\underline{\underline{I}} - \underline{\underline{K}}\,\underline{G})\,\underline{\underline{B}}_{k-1} \qquad \text{and} \qquad \underline{\underline{B}}_k = \underline{\underline{M}}_k\,\underline{\underline{A}}_{k-1}\,\underline{\underline{M}}_k^T$$

Tangent model $\underline{\underline{M}}_k$:

$$\begin{aligned}
\frac{d}{dt}(dX) &= -\sigma\,dX + \sigma\,dY \\
\frac{d}{dt}(dY) &= \rho\,dX - dY - dX\,Z^b(t) - X^b(t)\,dZ \\
\frac{d}{dt}(dZ) &= dX\,Y^b(t) + X^b(t)\,dY - \beta\,dZ
\end{aligned}$$

$\underline{\underline{M}}_k$ obtained by integrating from t_{k-1} to t_k with the respective initial conditions $(1,0,0)$, $(0,1,0)$ and $(0,0,1)$ for (dX, dY, dZ)

Comparison of the methods (scores)



Hands-on: "Data assimilation for the Lorenz model" (1/2)

Lorenz model

The Lorenz model shows a chaotic dynamics. Write a Python program to simulate the evolution of this model:

- Parameters ($\sigma = 10, \rho = 28, \beta = 10/3$),
- Initial conditions $(x, y, z) = (10, 15, 20)$ for $t = -10$.

Twin experiments

- From the $t = 0$ state of the previous transitory simulation, compute the "true trajectory" for $t \in [0, t_f]$ with $t_f = 5$.
- Consider the values $x(t_i)$ for $t_i = i\,\tau$ with $i = \{1, 2..., K\}$, with $K = 50$, and $\tau = t_f/K$, perturbed by a random gaussian error of standard deviation $\sigma_r = 1$, as the measurement vector $\underline{y}^o$.

Hands-on: "Data assimilation for the Lorenz model" (2/2)

Data assimilation methods

At first, program the methods for Section 1.2 of Chapter 4:

- Chain of BLUE with $\underline{\underline{B}}$ constant
- Chain of BLUE with $\underline{\underline{B}} = \underline{\underline{A}}$
- Kalman filter with no error model

Explore other methods, from Chapter 3 or other sources, such as:

- Incremental 4DVar
- Kalman filter with model error
- Ensemble Kalman filter
- ...