Reduced Basis-Ensemble Kalman Filter method for PDEs

Internship Defense

Hassan Ballout

CSMI Master's program,
University of Strasbourg

August, 2023

Context

- General context: Building simulation → Importance and challenges.
- Specific context: RB-EnKF method, combination of:
  - Ensemble Kalman Filter (EnKF) method
  - Reduced Basis (RB) method

Context

- General context: Building simulation $\rightarrow$ Importance and challenges.
- Specific context: RB-EnKF method, combination of:
  - Ensemble Kalman Filter (EnKF) method
  - Reduced Basis (RB) method

Objectives

- Implement the RB-EnKF method in the Feel++ software.
- Validate the implementation.
- Document the project: Website [1].

Objectives

- Implement the RB-EnKF method in the Feel++ software.
- Validate the implementation.
- Document the project: Website [1].

Objectives

- Implement the RB-EnKF method in the Feel++ software.
- Validate the implementation.
- Document the project: Website [1].

Taylor-Green Vortex Inverse Problem

The governing parametric PDE (pPDE) can be described as follows:
Find $c(\cdot, \cdot; \mu) : \Omega \times (0, T] \to \mathbb{R}$ such that:

$$\begin{cases} \partial_t c - \mu \Delta c + \beta \cdot \nabla c = 0 & \text{in } \Omega \times (0, T] \\ \nabla c(x, t; \mu) \cdot n = 0 & \text{on } \Gamma_N \times (0, T], \\ c(x, t; \mu) = 0 & \text{on } \Gamma_D \times (0, T], \\ c(x, 0; \mu) = c_0(x; \mu) & \text{in } \Omega. \end{cases} \tag{1}$$

Where β is the taylor-green vortex velocity field
Inverse Problem: Estimate μ^* from noisy observations $\mathcal{G}(\mu^*) = \mathcal{L}c(., .; \mu^*)$.

Taylor-Green Vortex Problem

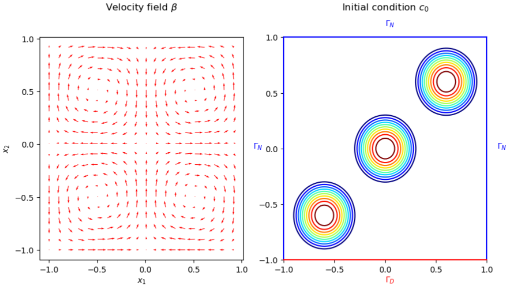
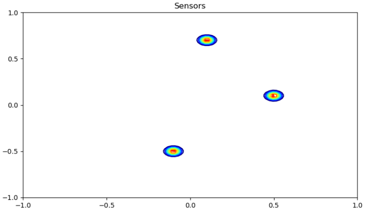


Figure 1: Illustration of the velocity field β, the initial condition $c_0(\cdot;\mu)$, and the boundaries.

Measurement Process

for a given parameter μ, the observation operator $\mathcal{G}$ is defined as:
$\mathcal{G}(\mu) = \mathcal{L}c(\cdot,\cdot;\mu) = (L1, L2, \ldots, L_{128})$, where c is the solution of the pPDE.
and $L_k = \int_{\mathcal{I}} \int_{\Omega} \nu_j \cdot \eta_i \cdot c(x,t;\mu)\, dx\, dt = \int_{\mathcal{I}} \nu_j (\int_{\Omega} \eta_i \cdot c(x,t;\mu)\, dx)\, dt$,
where $k = 3j + i$, $i = 1, \ldots, 3$, $j = 1, \ldots, 40$.



Iterative ensemble method for inverse problems

Observations artificial generation: $y = \mathcal{G}_h(\mu^*) +$ **noise. (FEM + Postprocess)**

- First ensemble $\mathcal{E}_0 = \{\mu_0^{(j)}\}_{j=1}^{J}$ generated from a uniform distribution on $P = [1/50, 1/10]$.
- For an ensemble size J, we apply FEM J times to solve the pPDE and compute $\mathcal{G}_h(\mu_i^{(j)})$ for each parameter $\mu_i^{(j)} \in \mathcal{E}_i$.
- We compute the sample covariances P_n and Q_n used to compute the Kalman gain $K_n = Q_n(P_n + \Sigma)^{-1}$. Where Σ is the observation noise covariance matrix.
- Update the ensemble:

$$\mu_{n+1}^{(j)} = \mu_n^{(j)} + K_n\,(y_n^j - \mathcal{G}(\mu_n^{(j)})), \quad y_n^j \sim N(y, \Sigma)$$

In practice

- For EnKF: Python class "IterativeEnKM".
- For FEM: Feel++ cfpdes toolbox.

In practice

- For EnKF: Python class "IterativeEnKM".
- For FEM: Feel++ cfpdes toolbox.

Results FEM

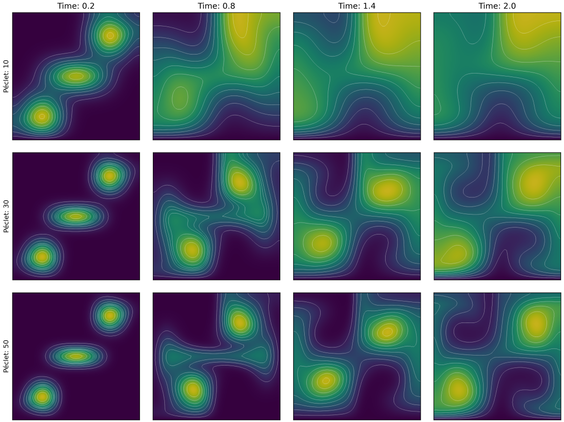


Figure 2: Solution of the PDE for different values of μ and t.

Results Full Order EnKM

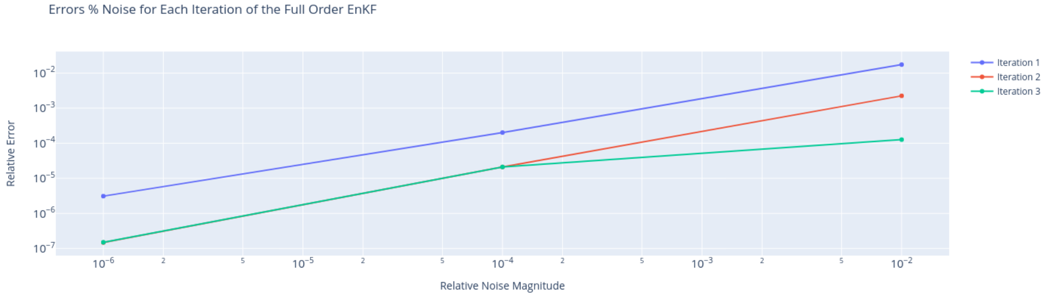


Figure 3: Noise impact on the error

Comments

- The error is decreasing with the observations noise variance$\rightarrow$ Measurements quality matters.
- The algorithm in its current form is computationally intensive$\rightarrow$ 300 minutes for the experiment above.
- Solution: Reduced basis method!

Comments

- The error is decreasing with the observations noise variance$\rightarrow$ Measurements quality matters.
- The algorithm in its current form is computationally intensive$\rightarrow$ 300 minutes for the experiment above.
- Solution: Reduced basis method!

Comments

- The error is decreasing with the observations noise variance$\rightarrow$ Measurements quality matters.
- The algorithm in its current form is computationally intensive$\rightarrow$ 300 minutes for the experiment above.
- Solution: Reduced basis method!

Comments

- The error is decreasing with the observations noise variance $\rightarrow$ Measurements quality matters.
- The algorithm in its current form is computationally intensive $\rightarrow$ 300 minutes for the experiment above.
- Solution: Reduced basis method!

Reduced Basis Approximation

For a simple elliptic problem we define the solution manifold: $\mathcal{M} = \{u(\mu) \,|\, \mu \in P\} \subset X$. We aim to approximate $\mathcal{M}$ by a low-dimensional subspace $\mathcal{X}_N$ spanned by a set of $N \ll N_h$ well-chosen basis functions.

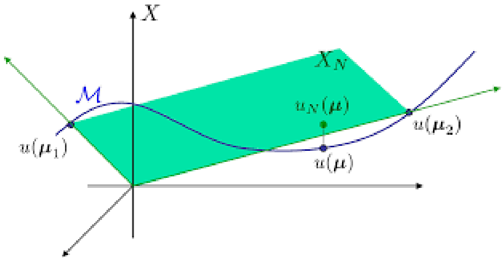


Figure 4: Solution manifold $\mathcal{M}$ and reduced basis subspace $\mathcal{X}_N$.

Offline-Online

- **Offline**:
  - One time, computationally intensive, parameter independent.
  - Compute the reduced basis.
  - Compute parameter-independent and N_h-dependent quantities.
- **Online**: For each $\mu \in \mathcal{P}$:
  - Compute the parameter-dependent quantities.
  - Compute the output of interest.
- Heavy offline + cheap online $\Longrightarrow$ **Efficient RB model**.

Offline-Online

- **Offline**:
  - One time, computationally intensive, parameter independent.
  - Compute the reduced basis.
  - Compute parameter-independent and N_h-dependent quantities.
- **Online**: For each $\mu \in \mathcal{P}$:
  - Compute the parameter-dependent quantities.
  - Compute the output of interest.
- Heavy offline + cheap online $\Longrightarrow$ **Efficient RB model.**

Offline-Online

- **Offline**:
  - One time, computationally intensive, parameter independent.
  - Compute the reduced basis.
  - Compute parameter-independent and N_h-dependent quantities.
- **Online**: For each $\mu \in \mathcal{P}$:
  - Compute the parameter-dependent quantities.
  - Compute the output of interest.
- Heavy offline + cheap online $\Longrightarrow$ **Efficient RB model**.

In practice: Feel++ MOR

- **Model description**: parameter space, affine decomposition, output, boundary conditions...
- **Offline basis generation**: 3 outputs → 3 basis.
- **Online:** MORModels in python

In practice: Feel++ MOR

- **Model description**: parameter space, affine decomposition, output, boundary conditions...
- **Offline basis generation**: 3 outputs $\rightarrow$ 3 basis.
- **Online:** MORModels in python

In practice: Feel++ MOR

- **Model description**: parameter space, affine decomposition, output, boundary conditions...
- **Offline basis generation**: 3 outputs $\rightarrow$ 3 basis.
- **Online:** MORModels in python

Gain in computational time

For a given parameter μ:

- **Full order output using cfpdes**: ≈ 40 s.
- **Reduced basis output** : ≈ 0.04 s. (online cost)

The acceleration achieved is over 1000.

Drawback: Approximation error.

Problem statement

- **Problem**: The full order EnKM is computationally expensive.
- **Solution**:
  - Simple substitution of FEM by RB $\rightarrow$ **Biased RB-EnKM**.
  - Take into account the approximation error $\rightarrow$ **Adjusted RB-EnKM**.

Problem statement

- **Problem**: The full order EnKM is computationally expensive.
- **Solution**:
  - Simple substitution of FEM by RB $\rightarrow$ **Biased RB-EnKM**.
  - Take into account the approximation error $\rightarrow$ **Adjusted RB-EnKM**.

Adjusted RB-EnKM offline

Offline:

Let P_{TRAIN} be a set of S parameters $\{\mu_0^{(s)}\}_{s=1}^S$ sampled from the distribution $\Pi_0(\mu)$. Based on the associated full order and reduced solutions $u_h(\mu^{(s)})$ and $u_\epsilon(\mu^{(s)})$, respectively, we define the training biases:

$\delta_\epsilon(\mu^{(s)}) = Lu_h(\mu^{(s)}) - Lu_\epsilon(\mu^{(s)})$

and the associated empirical moments:

$\Gamma_\epsilon = \frac{1}{S}\sum_{s=1}^S \delta_\epsilon(\mu^{(s)})\delta_\epsilon(\mu^{(s)})^T - \bar{\delta}_\epsilon\bar{\delta}_\epsilon^T$

with $\bar{\delta}_\epsilon = \frac{1}{S}\sum_{s=1}^S \delta_\epsilon(\mu^{(s)})$.

Results Biased RB-EnKM

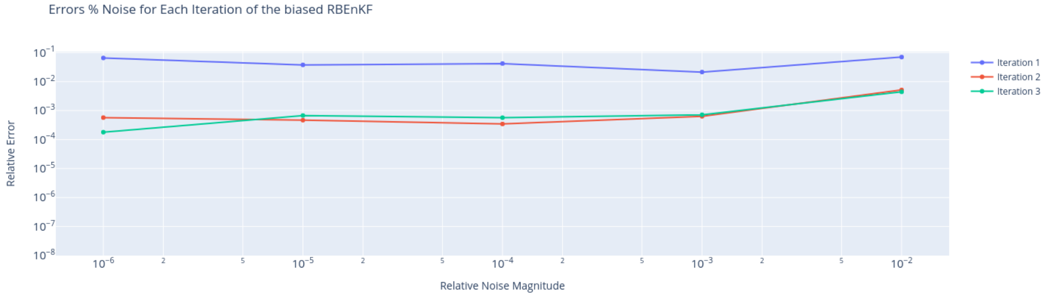


Figure 5: Noise impact on the error

Results Adjusted RB-EnKM

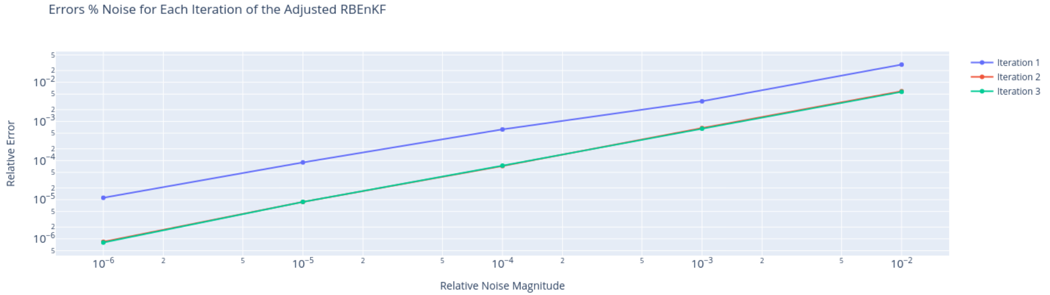


Figure 6: Noise impact on the error

Conclusion

- We have achieved the objectives of the internship.
- Some optimizations could be done to the RB-EnKF class.

Feedback

- Very helpful and friendly atmosphere.
- Research and development experience.

[1] H. Ballout, "Ktirio rb-enkf," https://master-csmi.github.io/2023-m1-ktirio-rbenkf/ktirio-rbenkf/index.html.

[2] J. Hesthaven, G. Rozza, and B. Stamm, *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*, 01 2016.

[3] S. Veys, "A computational framework for the reduced basis method: Applications to non-linear multi-physical problems," Ph.D. dissertation, Université de Grenoble, 2014. [Online]. Available: https://theses.hal.science/tel-01079415

[4] M. Asch, M. Bocquet, and M. Nodet, *Data assimilation: methods, algorithms, and applications*, ser. Fundamentals of Algorithms. SIAM, 2016. [Online]. Available: https://inria.hal.science/hal-01402885

[5] F. A. B. Silva, C. Pagliantini, M. Grepl, and K. Veroy, "A reduced basis ensemble kalman method," 2022. [Online]. Available: https://arxiv.org/abs/2210.02279

Thank You