Reduced-Basis method for 4DVar data assimilation

S. Boyaval
Laboratoire d’hydraulique Saint-Venant, EDF’lab Chatou
Ecole des Ponts ParisTech (– EDF R&D – CEREMA)
& Inria Paris, MATHERIALS (France)

collab. with M. Kärcher, M. Grepl, K. Veroy (RWTH)
Outline

Problem setting: 4Dvar parametrized

Reduced-Basis: error estimate + greedy projection

Numerical tests: error effectivity and decay rate
A Gaussian linear parametric SPDE model

Advection-diffusion of a quantity $c$ with Gaussian source

$$(\partial_t + u \cdot \nabla - \nu \Delta) c = f + \dot{B}$$

is parametrized by 2 kinds of input:

- boundary conditions like $c(t = 0)$, $\dot{B} := \left( \dot{B}(t) \right)_{t \in (0, T)}$
- process parameters like $\nu$

Having in mind a two-step Data Assimilation (DA) procedure

1. given $\nu$, fit $c(t = 0)$, $\dot{B}$ to “data about $c(t; \nu), t \in [0, T]$”
2. given $\{ c(t; \nu), t \in [0, T); \nu \in \Lambda \}$ “optimize” $\nu$

the Reduced-Basis method can decrease computational costs: $c(t = 0)$, $\dot{B}$ have to be fit for many values of the parameter $\nu$. 

Reduced-Basis for 4DVar
Problem setting: 4Dvar parametrized
A **Gaussian linear parametric** SPDE model

Advection-diffusion of a quantity $c$ with Gaussian source

$$(\partial_t + u \cdot \nabla - \nu \Delta)c = f + \dot{B}$$

is parametrized by 2 kinds of input:

- boundary conditions like $c(t = 0), \dot{B} := (\dot{B}(t))_{t \in (0, T)}$
- process *parameters* like $\nu$

Having in mind a *two-step* Data Assimilation (DA) procedure

1. given $\nu$, fit $c(t = 0), \dot{B}$ to “data about $c(t; \nu), t \in [0, T]$”
2. given $\{c(t; \nu), t \in [0, T]; \nu \in \Lambda\}$ “optimize” $\nu$

the *Reduced-Basis* method can decrease computational costs: $c(t = 0), \dot{B}$ have to be fit *for many values of the parameter* $\nu$. 
**DA by smoothing in a discrete setting**

In practice, one considers calibrating the input of a space-discrete model at discrete times $t^n \in [0, N\Delta t]$

$$(m + \Delta t \ a(\nu))c^n = m \ c^{n-1} + \Delta t \ f^n + \sqrt{\Delta t} \ g_j w_j^n$$

which we rewrite in standard DA notations

$$x_n = M(\nu)x_{n-1} + f_n + \eta_n \quad \eta_n \sim \mathcal{N}(0, Q_n)$$

One DA approach using $z_n = H x_n + \epsilon_n, \epsilon_n \sim \mathcal{N}(0, R_n)$ maximizes the posterior probability law

$$x_0, \ldots, x_N|z_0 = z_0^d, \ldots, z_N = z_N^d \sim \mathcal{N}(x^s, P^s)$$

using a background as prior $mc^{-1} + \Delta t \ f_0$ (=:smoothing)

$$x_0 \sim \mathcal{N}(x_0^f := Mx_{-1} + f_0, P_0^f := Q_0)$$
DA by **smoothing** in a discrete setting

In practice, one considers calibrating the input of a space-discrete model at discrete times $t^n \in [0, N\Delta t]$

$$(m + \Delta t \ a(\nu))c^n = m \ c^{n-1} + \Delta t \ f^n + \sqrt{\Delta t} \ g_j w_j^n$$

which we rewrite in standard DA notations

$$x_n = M(\nu)x_{n-1} + f_n + \eta_n \quad \eta_n \sim \mathcal{N}(0, Q_n)$$

One DA approach using $z_n = H \ x_n + \epsilon_n, \ \epsilon_n \sim \mathcal{N}(0, R_n)$ maximizes the *posterior* probability law

$$x_0, \ldots, x_N|z_0 = z_0^d, \ldots, z_N = z_N^d \sim \mathcal{N}(x^s, P^s)$$

using a *background* as prior $mc^{-1} + \Delta t \ f_0$ (**smoothing**)

$$x_0 \sim \mathcal{N}(x_0^f := Mx_{-1} + f_0, P_0^f := Q_0)$$
MAP of $\mathcal{N}(x^s, P^s)$: quadratic minimization

$$p(x_0 \ldots x_N | z_0 \ldots z_N) \propto \prod_{k=0}^{N} p(z_k | x_k) p(x_k | x_{k-1})$$

MAP minimizes

$$J(x_0 \ldots x_N) := \sum_{n=0}^{N} (Hx_n - z^d_n)^T R_n^{-1} (Hx_n - z^d_n)$$

$$+ \sum_{n=1}^{N} (x_n - Mx_{n-1} - f_{n-1})^T Q_n^{-1} (x_n - Mx_{n-1} - f_{n-1})$$

$$+ (x_0 - x^f_0)^T Q_0^{-1} (x_0 - x^f_0)$$

(1)

the so-called weak 4DVar computational problem: a large (invertible) linear system parametrized by $\nu$ in $M(\nu)$. 

Reduced-Basis for 4DVar

Problem setting: 4Dvar parametrized
4DVar with parameter: cost decreased by RB

4DVar is computationally expensive, especially if $N \gg 1$.

Computing $x^s(\nu)$ for many $\nu$ is very expensive.

A Reduced Basis (RB) approximation $\tilde{x}^s(\nu) \approx x^s(\nu)$ decreases the computational cost of $x^s(\nu)$ after a learning stage has exploited enough variations in $\nu$.

A good estimator for $\tilde{x}^s(\nu) - x^s(\nu)$ is crucial to a fast learning stage.

Note: in practice, the 4DVar saddle-point system is often reduced but without error control.
Outline

Problem setting: 4Dvar parametrized

Reduced-Basis: error estimate + greedy projection

Numerical tests: error effectivity and decay rate
Reduced Basis \( \approx \) “hyper-Galerkin”, \( \Rightarrow \) error estimate!

When \( x_n = Mx_{n-1} + f_n + \eta_n \) results from PDE discretization

\[
(m + \Delta t \, a)c^n = mc^{n-1} + \Delta t \, f^n + \sqrt{\Delta t} \, g_j w_j^n
\]

proj. \( x \approx \hat{x} = Xy \) onto Reduced Basis \( \text{rank}(X) \ll N \times \#(d.o.f.) \)

\[
(\hat{m} + \Delta t \, \hat{a})\hat{c}^n = \hat{m}\hat{c}^{n-1} + \Delta t \, \hat{f}^n + \sqrt{\Delta t} \, \hat{g}_j w_j^n
\]

is computationally cheaper . . . once \( X \) has been identified!

Using residuals: \( (m + \Delta t \, a)e_c^n = me_c^{n-1} + \Delta t \, r_c^n + \Delta t \, e_B^n \), greedy algorithms construct \( X \) incrementally, inspecting a sample of \( \|e_c^N(\nu)\|^2 := \|c^N - X\hat{c}^N\|_{a_0}^2 \) through estimates:

\[
(\beta_m + \frac{\Delta t}{4} \beta_a)\|e_c^n\|^2 \leq \beta_m \|e_c^{n-1}\|^2 + \frac{\Delta t}{2} \beta_a^{-1} \|r_c^n\|_{a_0}^2 + \frac{\Delta t}{2} C_m \|e_B^n\|_m^2
\]
Reduced Basis \( \approx \) “hyper-Galerkin”, \( \Rightarrow \) error estimate!

When \( x_n = Mx_{n-1} + f_n + \eta_n \) results from PDE discretization

\[
(m + \Delta t \ a)c^n = mc^{n-1} + \Delta t \ f^n + \sqrt{\Delta t} \ g_j w_j^n
\]

proj. \( x \approx \hat{x} = Xy \) onto Reduced Basis \( \text{rank}(X) \ll N \times \#(d.o.f.) \)

\[
(\hat{m} + \Delta t \ \hat{a})\hat{c}^n = \hat{m}\hat{c}^{n-1} + \Delta t \ \hat{f}^n + \sqrt{\Delta t} \ \hat{g}_j w_j^n
\]

is computationally cheaper . . . once \( X \) has been identified!

Using residuals: \( (m + \Delta t \ a)e_c^n = me_c^{n-1} + \Delta t \ r_c^n + \Delta t \ e_B^n \),

*greedy algorithms* construct \( X \) incrementally, inspecting a sample of \( \|e_c^N(\nu)\|^2 := \|c^N - X\hat{c}^N\|_a^2 \) through estimates:

\[
(\beta_m + \frac{\Delta t}{4} \beta_a)\|e_c^n\|^2 \leq \beta_m\|e_c^{n-1}\|^2 + \frac{\Delta t}{2} \beta_a^{-1}\|r_c^n\|_{a'}^2 + \frac{\Delta t}{2} C_m\|e_B^n\|_m^2
\]
Reduced Basis $\approx$ “hyper-Galerkin”, $\Rightarrow$ error estimate!

When $x_n = Mx_{n-1} + f_n + \eta_n$ results from PDE discretization

$$(m + \Delta t \ a)c^n = mc^{n-1} + \Delta t f^n + \sqrt{\Delta t} \ gjw_j^n$$

proj. $x \approx \hat{x} = Xy$ onto Reduced Basis $\text{rank}(X) \ll N \times \#(d.o.f.)$

$$(\hat{m} + \Delta t \ \hat{a})\hat{c}^n = \hat{m}\hat{c}^{n-1} + \Delta t \ \hat{f}^n + \sqrt{\Delta t} \ \hat{g}jw_j^n$$

is computationally cheaper . . . once $X$ has been identified!

Using residuals: $(m + \Delta t \ a)e_c^n = me_c^{n-1} + \Delta t \ r_c^n + \Delta t \ e_B^n$, greedy algorithms construct $X$ incrementally, inspecting a sample of $\|e_c^N(\nu)\|^2 := \|c^N - X\hat{c}^N\|^2_{a_0}$ through estimates:

$$\beta_a \sum_{n=1}^{N} \|e_c^n\|^2 \leq \beta_m \|e_c^0\|^2 + 2\beta_a^{-1} \sum_{n=1}^{N} \|r_c^n\|^2_{a_0} + 2Cm \sum_{n=1}^{N} \|e_B^n\|^2_m := \Delta$$
Reduced-Basis for 4DVar

Reduced-Basis: error estimate + greedy projection

Reduced Basis construction for parabolic PDEs

Standard RB uses a $POD$-greedy algorithm:

1. $X = \text{span}\{\zeta_1\}$ using
   
   $\zeta_1$ principal component of $c^1(\nu_1), \ldots, c^N(\nu_1)$ at $\nu_1$

2. While $\max_{\nu \in \Lambda_{\text{train}}} \Delta(\nu) > \varepsilon$, $X = X \cup \{\zeta\}$ using
   
   POD modes $\zeta$ of $c^1(\bar{\nu}), \ldots, c^N(\bar{\nu})$ at $\bar{\nu} \in \arg\max_{\nu \in \Lambda} \Delta(\nu)$

Key to the reduction are:

- the “linear” dimension of $\{c^1(\nu), \ldots, c^N(\nu); \nu \in \Lambda\}$
- the convergence rate of the greedy algorithm
- the accuracy in error estimate

Let us specialize to 4DVar
Reduced-Basis for 4DVar

Reduced-Basis: error estimate + greedy projection

Reduced Basis construction for parabolic PDEs

Standard RB uses a *POD-greedy algorithm*:

1. $X = \text{span}\{\zeta_1\}$ using
   $\zeta_1$ principal component of $c^1(\nu_1), \ldots, c^N(\nu_1)$ at $\nu_1$

2. While $\max_{\nu \in \Lambda_{\text{train}}} \Delta(\nu) > \varepsilon$, $X = X \cup \{\zeta\}$ using
   POD modes $\zeta$ of $c^1(\bar{\nu}), \ldots, c^N(\bar{\nu})$ at $\bar{\nu} \in \arg\max_{\nu \in \Lambda} \Delta(\nu)$

Key to the reduction are:

- the “linear” dimension of $\{c^1(\nu), \ldots, c^N(\nu); \nu \in \Lambda\}$
- the convergence rate of the greedy algorithm
- the accuracy in error estimate

Let us *specialize* to 4DVar
Reduced Basis construction for 4DVar: estimate

Duality: rewrite $J$ with $\sum_{n=1}^{N} (p^n)^T Q_n^{-1} p^n$ and

$$+ \left( (m + \Delta t \, a) c^n - mc^{n-1} - \Delta t \, f^n - \sqrt{\Delta t} \, g_j w_j^n \right)^T p^n$$

so 4Dvar rewrites as a system for $c^n, p^n$ with $p^N = 0$ and

$$(m + \Delta t \, a) p^{n-1} = mp^n + H^T R_n^{-1} (z_n^d - Hc^n)$$

which can be treated by RB like the (forward) eq. for $c^n$

$$\sum_{n=1}^{N} \|e_p^n\|^2 \leq \beta_a^{-1} \left( 2\beta_a^{-1} \sum_{n=1}^{N} \|r_p^n\|_{a_0}^2 + 2C_m C_{HTR^{-1}} H \sum_{n=1}^{N} \|e_c^n\|^2 \right)$$
Reduced Basis construction for 4DVar: greedy

\[
\| e^0_c \|^2 + \sum_{n=1}^{N} \| e^n_B \|^2 \lesssim \beta_a^{-2} \left( \sum_{n=1}^{N} \| r^n_p \|^2_{a_0} + \sum_{n=1}^{N} \| r^n_c \|^2_{a'_0} \right) := \Delta'
\]

allows 4DVar RB to use a new POD-greedy given \( \epsilon > 0 \):

1. \( X = \text{Span}\{\zeta_1, \zeta_2\} \)

   \( \zeta_1 \) principal component of \( c^1(\nu_1), \ldots, c^N(\nu_1) \)

   \( \zeta_2 \) principal component of \( p^1(\nu_1), \ldots, p^N(\nu_1) \)

2. While \( \max_{\nu \in \Lambda_{\text{train}}} \Delta'(\nu) > \epsilon \), \( X = X + \text{Span}\{\zeta, \zeta'\} \)

   \( \zeta \) principal component of \( c^1(\bar{\nu}), \ldots, c^N(\bar{\nu}) \)

   \( \zeta' \) principal component of \( p^1(\bar{\nu}), \ldots, p^N(\bar{\nu}) \)

using for \( \bar{\nu} \in \arg\max\{\Delta'(\nu), \nu \in \Lambda_{\text{train}}\} \)
Outline

Problem setting: 4Dvar parametrized

Reduced-Basis: error estimate + greedy projection

Numerical tests: error effectivity and decay rate
Advection-diffusion in TG vortices

Kärcher M.; Boyaval, S.; Grepl, M. A. & Veroy, K.
Reduced basis approximation and a posteriori error bounds for 4D-Var data assimilation, Optimization and Engineering 2018

advection by: \((\sin(\pi x_1) \cos(\pi x_2), -\cos(\pi x_1) \sin(\pi x_2))\)

\(\nu \in [.02, .1]\) \(\mathbb{P}_1\) \(\Delta t = .04\) 200 time steps \(R \equiv .025\)
Advection-diffusion in TG vortices

\( \nu = 0.1 \quad \nu^{\text{true}} = 0.03 \quad \nu = 0.02 \)

\( k = 20 \)

\( k = 40 \)
Advection-diffusion in TG vortices

\( \nu = .1 \quad \nu^{\text{true}} = .03 \quad \nu = .02 \)

\( k = 80 \)

\( k = 160 \)
Error estimate

Strong 4DVar $Q = 0$ (left) and weak $Q = .1$ (right)
Reduced-Basis for 4DVar
Numerical tests: error effectivity and decay rate

Optimum: $\nu^* = 0.034$ (strong), 0.022 (weak)

<table>
<thead>
<tr>
<th>N</th>
<th>$e_{J,N}^\text{max}$ (strong)</th>
<th>$e_{\nu,N}$ (strong)</th>
<th>$e_{J,N}^\text{max}$ (weak)</th>
<th>$e_{\nu,N}$ (weak)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>3.12e-01</td>
<td>4.18e-01</td>
<td>2.44e-01</td>
<td>6.02e-02</td>
</tr>
<tr>
<td>20</td>
<td>7.36e-03</td>
<td>1.30e-01</td>
<td>1.70e-02</td>
<td>9.33e-03</td>
</tr>
<tr>
<td>30</td>
<td>8.22e-04</td>
<td>1.42e-03</td>
<td>3.51e-03</td>
<td>1.70e-04</td>
</tr>
<tr>
<td>50</td>
<td>1.14e-05</td>
<td>2.98e-05</td>
<td>2.05e-04</td>
<td>3.53e-05</td>
</tr>
<tr>
<td>60</td>
<td>4.36e-06</td>
<td>1.27e-05</td>
<td>9.70e-05</td>
<td>3.90e-05</td>
</tr>
<tr>
<td>70</td>
<td>3.92e-07</td>
<td>4.18e-06</td>
<td>3.58e-05</td>
<td>1.93e-05</td>
</tr>
<tr>
<td>80</td>
<td>8.76e-08</td>
<td>9.71e-08</td>
<td>1.05e-05</td>
<td>4.12e-06</td>
</tr>
<tr>
<td>90</td>
<td>-</td>
<td>-</td>
<td>4.17e-06</td>
<td>2.51e-06</td>
</tr>
<tr>
<td>100</td>
<td>-</td>
<td>-</td>
<td>1.94e-06</td>
<td>3.09e-06</td>
</tr>
</tbody>
</table>
Conclusion & Perspectives

- RB can be specialized to 4DVar with (LTI) parabolic PDEs
- Other models / DA procedures?

Thanks for listening