Model reduction method for solving parameter-dependent dynamical systems

Marie Billaud-Friess[⋆,†]
Joint work with : A. Nouy[⋆]

Work supported by GDR MOMAS

[†] marie.billaud-friess@ec-nantes.fr
[⋆]GeM, Ecole Centrale de Nantes

October 6th, 2015

Outline

1. **Motivations**
2. **Low-rank approximation methods**
3. **Proposed method**
4. **Numerical experiments**

General context

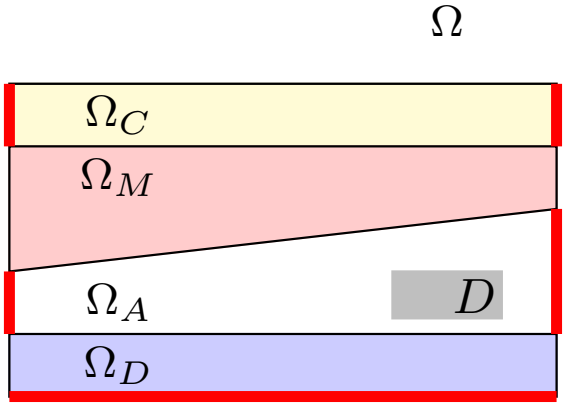
Parameter and time dependent problem

Find $\boldsymbol{u}(t, \boldsymbol{\xi}) : \boldsymbol{\xi} \times [0, T] \to \mathbb{R}^d$ such that

$$\mathcal{F}(\boldsymbol{u}(t, \boldsymbol{\xi}), t, \boldsymbol{\xi}) = 0$$

where $\boldsymbol{\xi} \in \boldsymbol{\Xi} \subset \mathbb{R}^p$ are parameters or random variables in measure space $(\boldsymbol{\Xi}, \mu)$.

Example : *Groundwater flows : benchmark* COUPLEX *with uncertainties* [1]



$$\omega \left(\frac{\partial C}{\partial t} + \lambda(\boldsymbol{\xi}) C \right) - \nabla \cdot (d_e(\boldsymbol{\xi}) \nabla C) + \boldsymbol{u}(\boldsymbol{\xi}) \cdot \nabla C = 0$$

with $\boldsymbol{u}(\boldsymbol{\xi}) = -K_i(\boldsymbol{\xi}) \nabla H$ s.t.

$$-\nabla \cdot (K(\boldsymbol{\xi}) \nabla H) = 0, \quad K(\boldsymbol{\xi}) = \sum_i \mathbf{1}_{\Omega_i} K_i(\boldsymbol{\xi})$$

Finite discretization $\rightsquigarrow$ Algebraic system with d very large

1. https://www.ljll.math.upmc.fr/cances/gdrmomas/ex_qualifications.html

Need of model reduction

Issue : Multiple parameter simulations can be expensive and intractable

- Variable of interest : $s(\boldsymbol{\xi}) = \ell(\boldsymbol{u}(\boldsymbol{\xi}))$
- Quantity of interest : statistical moments, failure probability, sensitivity indices

$\rightsquigarrow$ **Derive a surrogate problem cheaper to solve**

Strategy : Model order reduction methods

- Proper generalized decomposition,
- Reduced basis with POD-Greedy strategy,
- Dynamical low-rank approach,
- Dynamical Orthogonal approximation ...

$\rightsquigarrow$ **Low-rank approximation methods**

$$\boldsymbol{u} \approx \sum_{i=1}^{r} \boldsymbol{v}_i \otimes \alpha_i, \text{ with } r << d \text{ small.}$$

Problem setting

Parameter-dependent dynamical system

Find $\boldsymbol{u} : \boldsymbol{\xi} \times [0,T] \to X$ such that

$$\left\{ \begin{array}{rcll} \boldsymbol{u}'(t,\boldsymbol{\xi}) & = & \boldsymbol{f}(\boldsymbol{u}(t,\boldsymbol{\xi}),t,\boldsymbol{\xi}) & t \in (0,T), \\ \boldsymbol{u}(0,\boldsymbol{\xi}) & = & \boldsymbol{u}^0(\boldsymbol{\xi}). & \end{array} \right. \tag{1}$$

- $\boldsymbol{\xi} \in \boldsymbol{\Xi} \subset \mathbb{R}^p$ a (random) parameter vector ,
- $X = \mathbb{R}^d$ equiped with the L^2 scalar product $\langle \cdot, \cdot \rangle_X$,
- the *state* $\boldsymbol{u}$ is differentiable,
- $\boldsymbol{u}^0 : \boldsymbol{\xi} \to X$ is the initial datum,
- $\boldsymbol{f} : X \times [0,T] \times \boldsymbol{\Xi} \to X$ the *flux* that is Lipschitz continuous with respect to $\boldsymbol{u}$.

Low-rank approximation methods : a sub-space point of view

① Ideal rank-r approximation for parametric dynamical system (1)

$$\min_{\boldsymbol{v}(t)\in\mathcal{M}_r} \|\boldsymbol{v}(t) - \boldsymbol{u}(t,\cdot)\|_{\Xi,q}, \quad t>0$$

with $\mathcal{M}_r = \left\{\boldsymbol{v} = \sum_{i=1}^r \boldsymbol{v}_i \otimes \alpha_i, \boldsymbol{v}_i \in X, \alpha_i \in \mathrm{L}^p_\mu(\Xi)\right\}$ and $r \in \mathbb{N}$ such that $r \ll d$.

② Subspace point of view

$$\min_{\substack{X_r \subset X \\ \dim X_r = r}} \quad \min_{\boldsymbol{v}(t,\cdot)\in \mathrm{L}^p_\mu \otimes X_r} \|\boldsymbol{v}(t,\cdot) - \boldsymbol{u}(t,\cdot)\|_{\Xi,p}, \quad t>0$$

$\boldsymbol{u}$ unknown a priori $\rightsquigarrow$ Practical algorithms for computing sub-optimal approx. $\boldsymbol{u}_r$

L^2-norm : Dynamical low-rank approximation [Koch07]

L^∞-norm : Reduced basis approach

1. Greedy construction of $X_r \subset X$ using "snapshots" of the solution
2. Low-rank approximation $\boldsymbol{u}_r \in X_r$ by (Petrov-)Galerkin projection of (1) in X_r

Low-rank approximation with RB methods I

Low-rank approximation **with time dependent reduced space $X_r(t)$**

$$\boldsymbol{u}(t,\boldsymbol{\xi}) \approx \boldsymbol{u}_r(t,\boldsymbol{\xi}) = \sum_{i=1}^{r} \boldsymbol{v}_i(t)\alpha_i(t,\boldsymbol{\xi})$$

Method 1 : Weak-greedy algorithm

For each $t > 0$, while $\max_{\boldsymbol{\xi}\in\boldsymbol{\Xi}_{train}} \Delta_r(t,\boldsymbol{\xi}) > \varepsilon$ or $r < r_{max}$ do.

1. Find $\boldsymbol{\xi}_{r+1}(t) = \max_{\boldsymbol{\xi}\in\boldsymbol{\Xi}_{train}} \Delta_r(t,\boldsymbol{\xi})$.
2. Compute $\boldsymbol{u}(t,\boldsymbol{\xi}_{r+1}(t))$.
3. Set $X_{r+1}(t) = X_r(t) + \text{span}\{\boldsymbol{u}(t,\boldsymbol{\xi}_{r+1}(t))\}$.

✓ Δ_r surrogate of the exact projection error s.t.

$$c_r\Delta_r(t,\cdot) \leq \|\boldsymbol{u}(t,\cdot) - \boldsymbol{P}_r(t)\boldsymbol{u}(t,\cdot)\|_X \leq C_r\Delta_r(t,\cdot), \quad c_r, C_R > 0$$

✓ At each instant t, the weak-greedy algorithm converges [Bin11, Buff12]

✗ Costly strategy since $\boldsymbol{\xi}_r(t)$ is time dependent $\rightsquigarrow$ Many calls to (1).

Low-rank approximation with RB methods II

Low-rank approximation **with time independent reduced space X_r**

$$\boldsymbol{u}(t,\boldsymbol{\xi}) \approx \boldsymbol{u}_r(t,\boldsymbol{\xi}) = \sum_{i=1}^{r} \boldsymbol{v}_i \alpha_i(t,\boldsymbol{\xi})$$

Strategies : Snapshot method, POD based method, POD-greedy method

Method 2 : POD-greedy algorithm [Haas11]

While $\max_{\boldsymbol{\xi}\in\Xi_{train}} \|\Delta_r(\cdot,\boldsymbol{\xi})\|_{(0,T),2} > \varepsilon$ or $r < r_{max}$ do.

1. Find $\boldsymbol{\xi}_{r+1} = \max_{\boldsymbol{\xi}\in\Xi_{train}} \|\Delta_r(\cdot,\boldsymbol{\xi})\|_{(0,T),2}$.
2. Compute $t \mapsto \boldsymbol{u}(t,\boldsymbol{\xi}_{r+1})$ and $S(\cdot,\boldsymbol{\xi}_r) : t \mapsto \boldsymbol{u}(t,\boldsymbol{\xi}_{r+1}) - \boldsymbol{P}_r\boldsymbol{u}(t,\boldsymbol{\xi}_{r+1})$.
3. Set $X_{(r+1)\ell} = X_{r\ell} + \text{span}\{POD\left(S(\cdot,\boldsymbol{\xi}_{r+1}),\ell\right)\}$.

- ✓ Global error estimate $\|\Delta_r(\cdot,\boldsymbol{\xi})\|_{(0,T),2}$
  - ⇝ Time independent $\boldsymbol{\xi}_r$ with at most r calls to (1)
  - ⇝ Parameter $\boldsymbol{\xi}_r$ selected for given objective (e.g. GO norms)
- ✓ Quasi-optimal convergence rates for Algorithm 2 [Haas13]
- ✗ The approximation $\boldsymbol{u}_r$ is not an interpolation of $\boldsymbol{u}$ at points $\{\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_r\}$
- ✗ Large r when $\boldsymbol{u}$ has a rich spectral content (e.g. u solution of the advection equation)

Proposed approach

Goal : Design a time-dependent low-rank approximation method for solving parametric dynamical systems

$$\boldsymbol{u}(t,\boldsymbol{\xi}) \approx \boldsymbol{u}_r(t,\boldsymbol{\xi}) = \sum_{i=1}^{r} \boldsymbol{v}_i(t)\alpha_i(t,\boldsymbol{\xi})$$

Specifications : A method combining the best features of method 1 & 2

1. Time dependent reduced space $X_r(t) \subset X$ by snapshots of the solution selected with a greedy algorithm
2. Low-rank approximation $\boldsymbol{u}_r(t)$ by solving the reduced dynamical system obtained by (Petrov-)Galerkin projection of (1) on $X_r(t)$
3. Rigorous a-posteriori error estimation $\|\boldsymbol{u}_r(t,\cdot) - \boldsymbol{u}(t,\cdot)\|_X \leq \Delta_r(t,\cdot)$

Constructing the reduced space

Method 3 : T-greedy algorithm

While $\max_{\boldsymbol{\xi}\in\Xi_{train}} \|\Delta_r(\cdot,\boldsymbol{\xi})\|_{(0,T),2} > \varepsilon$ or $r < r_{max}$ do.

1. Find $\boldsymbol{\xi}_{r+1} = \arg\max_{\boldsymbol{\xi}\in\Xi_{train}} \|\Delta_r(t,\boldsymbol{\xi})\|_{(0,T),2}$.
2. Compute $t \mapsto \boldsymbol{u}(t,\boldsymbol{\xi}_{r+1})$ and $S(\cdot,\boldsymbol{\xi}_r) : t \mapsto \boldsymbol{u}(t,\boldsymbol{\xi}_{r+1}) - \boldsymbol{P}_r\boldsymbol{u}(t,\boldsymbol{\xi}_{r+1})$.
3. For each time $t > 0$ proceed as follows.
   - Set $\boldsymbol{v}_{r+1}(t) = \dfrac{S_{r+1}(t)}{\|S_{r+1}(t)\|_X}$.
   - Construct $X_{r+1}(t) = X_r(t) + \mathrm{span}\{\boldsymbol{v}_{r+1}(t)\}$.

Proposition

Assuming that $\dim X_r(t) = r$ for $t > 0$, then

1. the applications $\boldsymbol{v}_i : t \mapsto \boldsymbol{v}_i(t), i = 1,\dots,r$ exists and are differentiable,
2. the resulting family of functions $\{\boldsymbol{v}_i(t)\}_{i=1}^r$ is orthonormal for each $t > 0$.

Projection strategy

Notations : Here $\boldsymbol{\xi} \in \boldsymbol{\Xi}$ is fixed

- $\boldsymbol{V}_r(t) \in \mathbb{R}^{d\times r} = [\boldsymbol{v}_1(t), \ldots, \boldsymbol{v}_r(t)]$ s.t. $\boldsymbol{V}_r^T(t)\boldsymbol{V}_r(t) = \boldsymbol{I}_r$, $\boldsymbol{P}_r(t) = \boldsymbol{V}_r(t)\boldsymbol{V}_r^T(t)$,
- $\boldsymbol{\alpha}_r \in \mathbb{R}^r, \boldsymbol{u}_r \in \mathbb{R}^d$ s.t. $\boldsymbol{u}_r(t,\boldsymbol{\xi}) = \boldsymbol{V}_r(t)\boldsymbol{\alpha}_r(t,\boldsymbol{\xi}) \Leftrightarrow \boldsymbol{\alpha}_r(t,\boldsymbol{\xi}) = \boldsymbol{V}_r(t)^T\boldsymbol{u}_r(t,\boldsymbol{\xi})$.

① The Galerkin projection

$$\boldsymbol{V}_r(t)^T\boldsymbol{u}_r'(t,\boldsymbol{\xi}) = \boldsymbol{V}_r(t)^T\boldsymbol{f}(\boldsymbol{u}_r(t),\boldsymbol{\xi},t)$$

② The low-rank approximation u_r

$$\boldsymbol{u}_r'(t,\boldsymbol{\xi}) = \boldsymbol{P}_r(t)\boldsymbol{f}(\boldsymbol{u}_r(t,\boldsymbol{\xi}),\boldsymbol{\xi},t) + (\boldsymbol{I}_d - \boldsymbol{P}_r(t))\boldsymbol{V}_r'(t)\boldsymbol{\alpha}_r(t,\boldsymbol{\xi}) \quad \boldsymbol{u}_r(0) = \boldsymbol{P}_r(0)\boldsymbol{u}^0$$

Remark : Dependence in time of $X_r(t)$ taken into account by means of $\boldsymbol{V}_r'(t)$

③ The reduced dynamical system

$$\boldsymbol{\alpha}_r'(t,\boldsymbol{\xi}) = \boldsymbol{f}_r(\boldsymbol{\alpha}_r(t,\boldsymbol{\xi}),\boldsymbol{\xi},t) \quad \boldsymbol{\alpha}_r^0 = \boldsymbol{V}_r^T(0)\boldsymbol{u}^0 \ a$$

$$\text{with } \boldsymbol{f}_r(\boldsymbol{\alpha}_r(t,\boldsymbol{\xi}),\boldsymbol{\xi},t) = \boldsymbol{V}_r(t)^T\boldsymbol{f}(\boldsymbol{u}_r(t,\boldsymbol{\xi}),\boldsymbol{\xi},t) - \boldsymbol{V}_r(t)^T\boldsymbol{V}_r'(t)\boldsymbol{\alpha}_r(t,\boldsymbol{\xi})$$

---

a. For parameter independent initial condition $\boldsymbol{\alpha}_r^0 = (1,0,\ldots,0)$.

Error analysis

Linear case

Summary : For $\boldsymbol{f}(\boldsymbol{u}(t),t) = \boldsymbol{A}\boldsymbol{u}(t)$, the approximation error $\boldsymbol{e}(t) = \boldsymbol{u}_r(t) - \boldsymbol{u}(t)$ for $t > 0$ is

$$\begin{cases} \boldsymbol{e}'(t) &= \underbrace{\boldsymbol{A}\boldsymbol{e}(t)}_{\text{flux error}} + \underbrace{\boldsymbol{Q}_r(t)\boldsymbol{A}\boldsymbol{u}_r(t)}_{\text{flux projection}} - \underbrace{\boldsymbol{Q}_r(t)\boldsymbol{V}_r'(t)\boldsymbol{\alpha}_r(t)}_{\text{basis dynamics}} \\ \boldsymbol{e}(0) &= \boldsymbol{Q}_r(0)\boldsymbol{u}^0. \end{cases}$$

where $\boldsymbol{Q}_r(t) = \boldsymbol{P}_r(t) - \boldsymbol{I}_d \in \mathbb{R}^{d\times d}$

Proposition

The approximation error $\boldsymbol{e}$ satisfies $\|\boldsymbol{e}(t)\|_X \leq \Delta_r(t), \forall t \in [0,T]$, where

$$\Delta_r(t) := \left[\sup_{s\in[0,t]} \|\boldsymbol{r}(s)\|_X \frac{1 - e^{-\|\boldsymbol{A}\|_X t}}{\|\boldsymbol{A}\|_X} + \|\boldsymbol{e}(0)\|_X \right] e^{t\|\boldsymbol{A}\|_X}$$

with $\|\boldsymbol{A}\|_X = \lambda_{max}(\boldsymbol{A}\boldsymbol{A}^T)$ and $\boldsymbol{r}(s) = \boldsymbol{Q}_r(t)(\boldsymbol{A}\boldsymbol{u}_r(t) - \boldsymbol{V}_r'(t)\boldsymbol{\alpha}_r(t))$

Remark :

- Error estimate valid for time independent reduced space [Haas11]
- Coarse error estimate valid only for linear cases

Error analysis

General case

Summary : The approximation error $\boldsymbol{e}(t) = \boldsymbol{u}_r(t) - \boldsymbol{u}(t)$ for $t > 0$ is

$$\begin{cases} \boldsymbol{e}'(t) &= \underbrace{(\boldsymbol{f}(\boldsymbol{u}_r(t),t) - \boldsymbol{f}(\boldsymbol{u}(t),t))}_{\text{flux error}} + \underbrace{\boldsymbol{Q}_r(t)\boldsymbol{f}(\boldsymbol{u}_r(t),t)}_{\text{flux projection}} - \underbrace{\boldsymbol{Q}_r(t)\boldsymbol{V}_r'(t)\boldsymbol{\alpha}_r(t)}_{\text{basis dynamics}} \\ \boldsymbol{e}(0) &= \boldsymbol{Q}_r(0)\boldsymbol{u}^0. \end{cases}$$

Proposition (general case)

The approximation error $\boldsymbol{e}$ satisfies $\|\boldsymbol{e}(t)\|_X \leq \Delta_r(t), \quad \forall t \in [0,T]$, where

$$\begin{aligned} \Delta_r(t) &:= \int_0^t \gamma(s) e^{\int_s^t \beta(\tau)d\tau ds} + \|\boldsymbol{e}(0)\|_X e^{\int_0^t \beta(\tau)d\tau ds}, \\ \gamma(t) &= \|\boldsymbol{Q}_r(t)(\boldsymbol{f}(\boldsymbol{u}_r(t),t) - \boldsymbol{V}_r'(t)\boldsymbol{\alpha}_r(t))\|_X, \\ \beta(t) &= L_X[\boldsymbol{f}](\boldsymbol{u}_r(t)). \end{aligned}$$

In practice : A first order linearization of $\boldsymbol{f}$ is used [Wir14]

$$L_X[\boldsymbol{f}](\boldsymbol{y}_r(t)) \leq L_X[\boldsymbol{J}(\boldsymbol{u}_r(t))] + o(\|e\|_X)$$

with $\boldsymbol{J}(\boldsymbol{u}_r) \in \mathbb{R}^{d\times d}$ the Jacobian matrix of $\boldsymbol{f}$ at $\boldsymbol{u}_r$.

Test case 1

Deterministic dvection equation with periodic boundary conditions

$$\frac{\partial}{\partial t}u(x,t) + a\frac{\partial}{\partial x}u(x,t) = 0, \quad u^0(x) = \frac{1}{\sqrt{2\pi}}e^{-\frac{(x+0.06)^2}{0.05^2}}$$

- Spatial domain $\Omega = (-1,1)$ and time interval $I = (0,1)$.
- The advection velocity is $a = 1.325$.

Discretization :

1. Finite difference upwind scheme on a regular mesh $\Omega = \{x_i\}_{i=0}^{d+1}$ where $x_i = -1 + i\delta x$ and $\delta x = \frac{2}{d+1}$ with $d = 200$ nodes.

$$\frac{d}{dt}\boldsymbol{u}(t) = \boldsymbol{A}\boldsymbol{u}(t), \quad \boldsymbol{u}(0) = (u^0)_{i=1}^d.$$

2. Time integration with explicit Euler scheme on uniform discretization $I = \{t_k\}_{k=1}^K$, with $t^k = k\delta t$ and $\delta t = \delta x/2$

$$\boldsymbol{u}^{k+1} = (\boldsymbol{I}_d + \delta t\boldsymbol{A})\boldsymbol{u}^k.$$

Comparison : Confrontation of two projection methods with

1. fixed reduced space $X_r = \text{span}\{POD_\ell(\boldsymbol{u}^0, \ldots, \boldsymbol{u}^K)\}$
2. time dependent reduced space $X_r^k = \text{span}\{\boldsymbol{u}^k\}$

Method n°1

$\dim(X_r) = \ell$	E_2	E_∞
1	0.99862	1
5	0.99592	1
10	0.95843	0.41232
20	0.070717	0.010705
50	1.6532e-10	2.9288e-11
100	4.9305e-13	4.2383e-14

Tab.1 : Relative error

$$E_p = \|\boldsymbol{u}_r - \boldsymbol{u}\|_{(0,T),p} / \|\boldsymbol{u}\|_{(0,T),p}$$

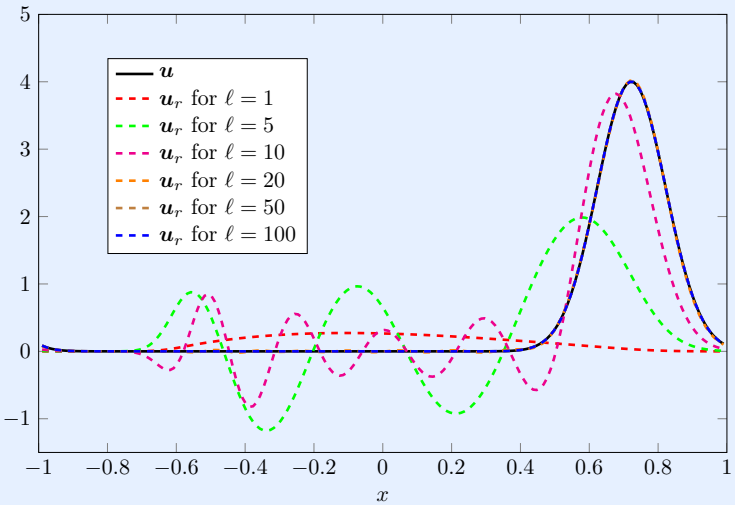


Fig.1 : Approximation compared to the exact solution at $T = 1$

1. For large r, fine precision $\rightsquigarrow$ limits due to POD
2. With $r = 1$, $E_2 =$ 1.9235e-15, $E_\infty =$3.9074 e-15 $\rightsquigarrow$ interpolant method

Test case 2

Advection-diffusion eq. with homogeneous Dirichlet boundary conditions

$$\frac{\partial}{\partial t}u(x,t,\xi) + a(\xi)\frac{\partial}{\partial x}u(x,t,\xi) - \mu(\xi)\frac{\partial}{\partial x^2}u(x,t,\xi) = 0, \quad u^0(x) = x(2-x)^2 e^{2x}.$$

- Spatial domain $\Omega = (0,2)$ and time interval $I = (0,1)$,
- $\mu(\xi) = 0.5(2 + cos(\pi\xi)^2)$ and $a(\xi) = 0.1\sin(\pi\xi)$ with $\xi \sim \mathcal{N}(0,1)$.

Discretization :

1. Finite difference scheme on a regular mesh with $d = 200$ nodes

$$\frac{d}{dt}\boldsymbol{u}(t) = \boldsymbol{A}(\xi)\boldsymbol{u}(\xi,t), \quad \boldsymbol{u}(0) = \boldsymbol{u}^0$$

2. Time integration with implicit Euler scheme with $K = 200$

$$(\boldsymbol{I}_X - \delta t \boldsymbol{A}(\xi))\boldsymbol{u}^{k+1}(\xi) = \boldsymbol{u}^k(\xi)$$

Results for T-greedy algorithm $r = 10$

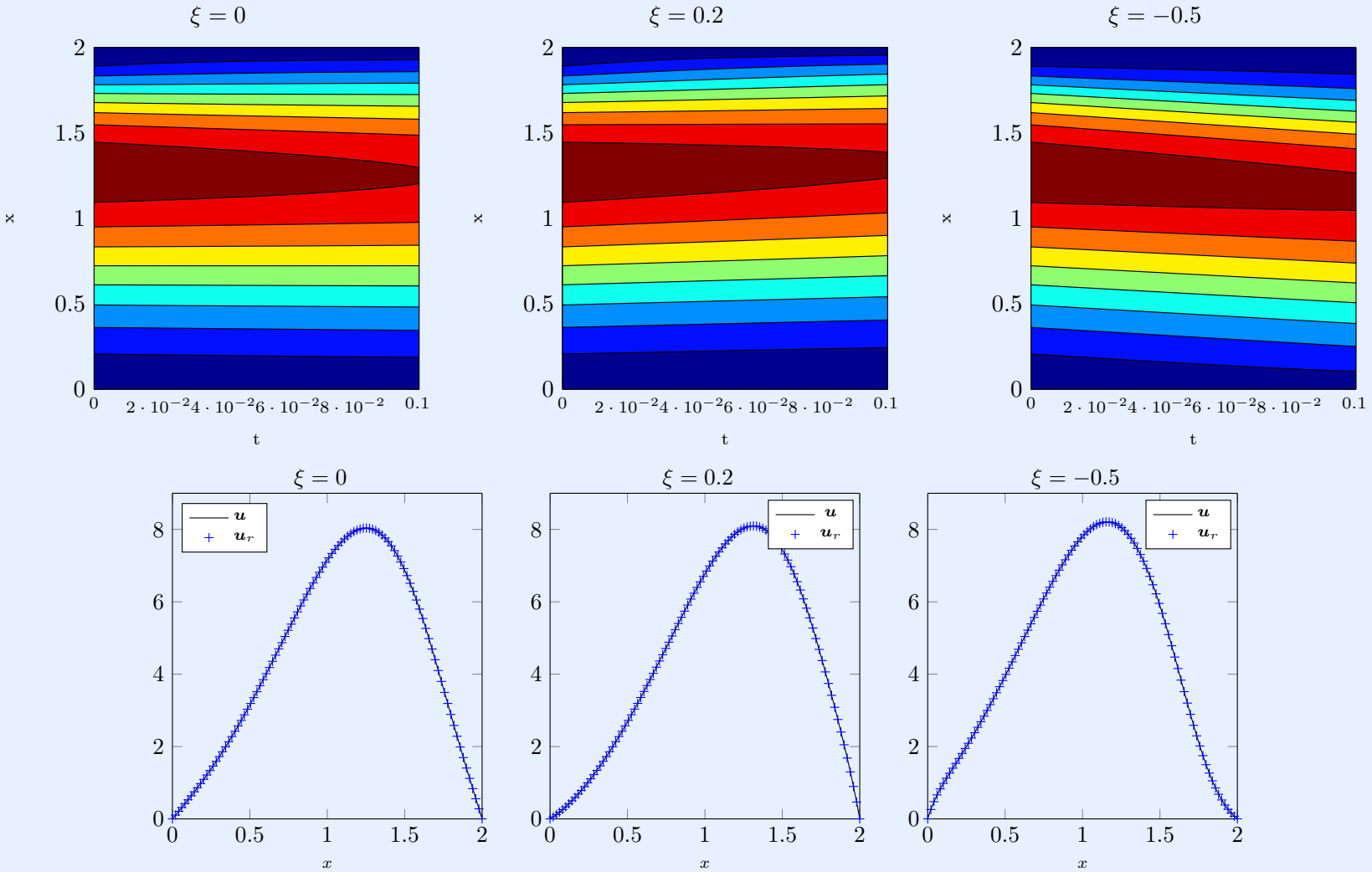


Fig.3 : Surface contours of $\boldsymbol{u}_r$ (top) and comparison to the exact solution at $T = 1$ (bottom)

T-greedy vs. POD-greedy

Greedy algorithm stopped for $r = r_{max}$

Data : Use of Δ_{NL}, with $\#\Xi_{train} = 30$ and $\ell = 1$ for POD-greedy

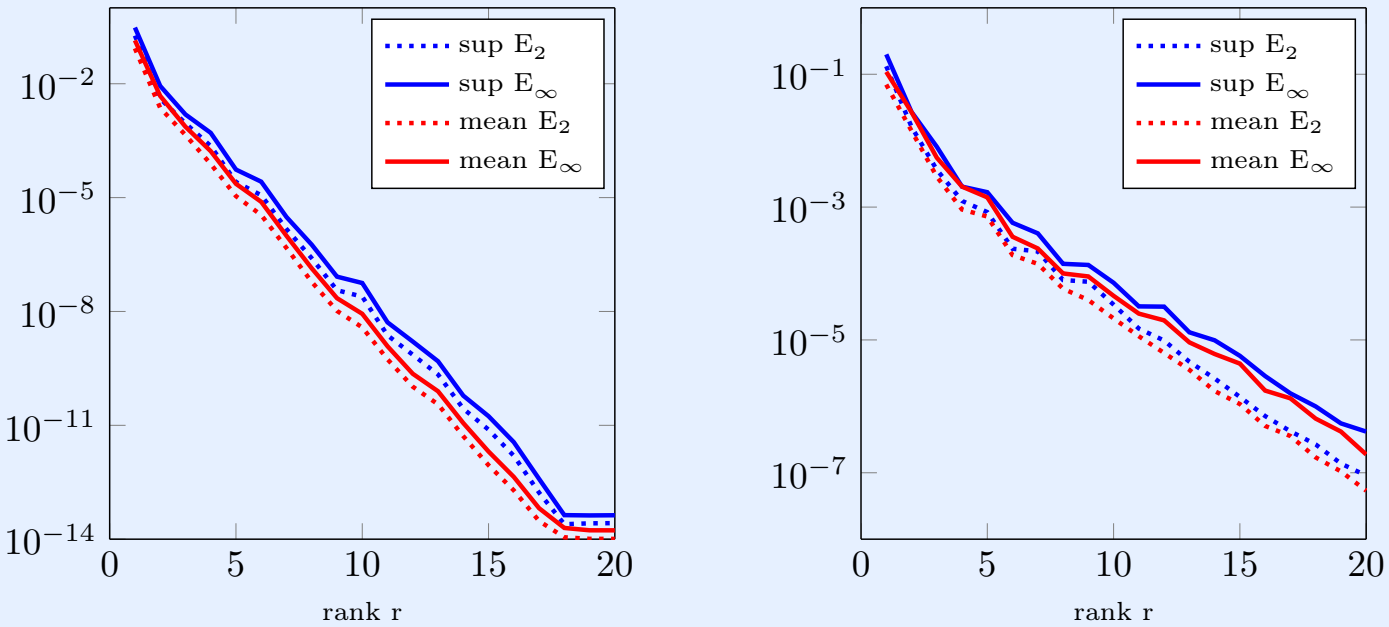


Fig. 4 : Relative error with respect to r : for T-greedy (left) and POD-greedy (right) methods

- Methods provide approximation with good keeping r small.
- POD-greedy converges more slowly than T-greedy.
- Coarser approximation for POD-greedy that requires richer reduced space.

Results for $r = 10$ and $\xi = 0.2$

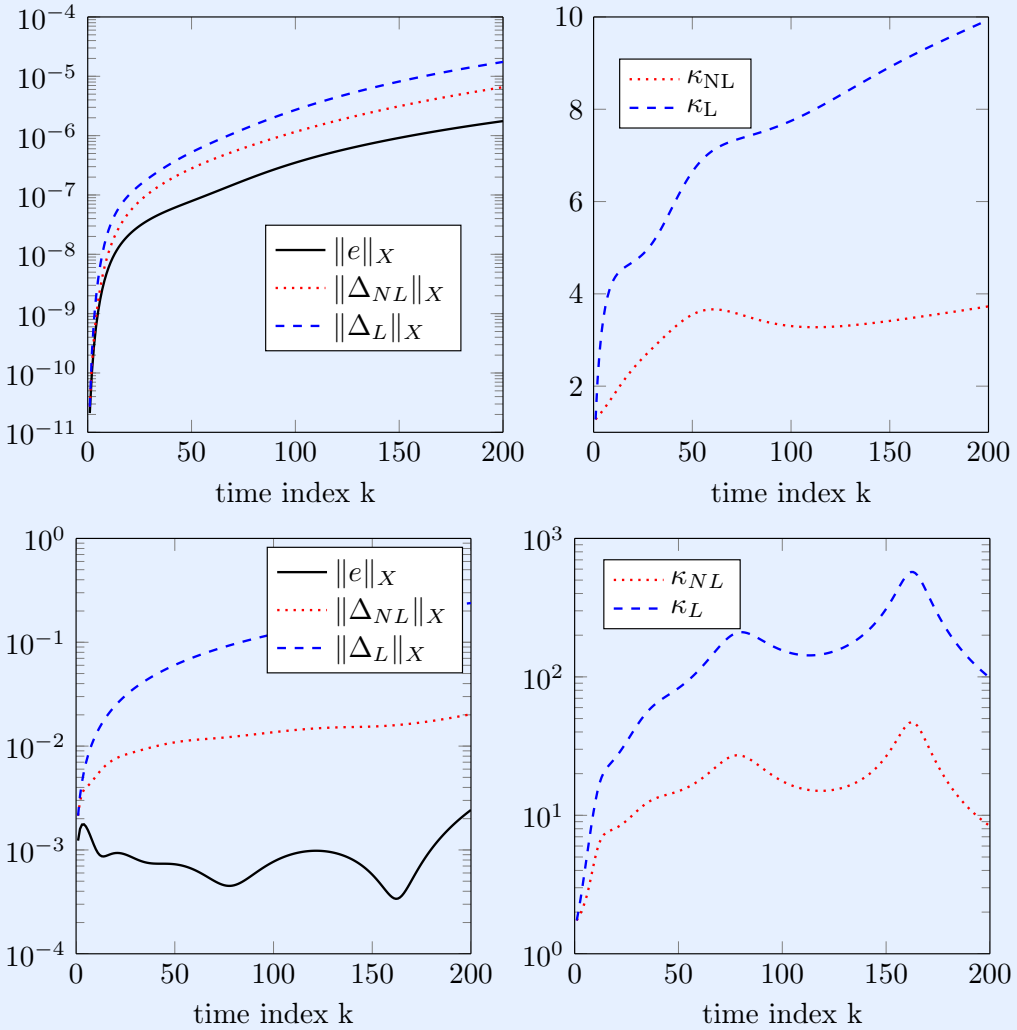


Fig. 5 : T-greedy (top) and POD-greedy (above) algorithm

Denoting the *effectivity index* as

$$\kappa_{L,NL} = \|\Delta_{L,NL}\|_X / \|e\|_X$$

① **For T-greedy algorithm :** efficient error estimates for with Δ_{NL} the sharpest.

$\max_t \|\kappa_L(k)\|_X$	$\max_k \|\kappa_{NL}(k)\|_X$
9.9393	3.72878

② **For the POD-greedy algorithm :** coarse error estimates with Δ_{NL} the sharpest.

$\max_k \|\kappa_L(k)\|_X$	$\max_k \|\kappa_{NL}(k)\|_X$
572.101	47.0514

T-greedy vs. POD-greedy

Greedy alg. with stopping criterion in precision ε

Data : Use of Δ_{NL}, with $\#\Xi_{train} = 30$ and $\ell = 1$ for POD-greedy

Strong greedy algorithm		
ε	T-G	POD-G
1e-1	5	7
1e-2	6	11
1e-3	8	15
1e-4	9	19
1e-5	11	23
1e-6	12	27

weak-greedy algorithm		
ε	T-G	POD-G
1e-1	6	12
1e-2	7	17
1e-3	8	21
1e-4	11	25
1e-5	12	-
1e-6	14	-

Tab. 3 : Reduced space dimension with respect to ε with **exact error criterion** (left), with **error estimate criterion** (right)

- Small reduced spaces with T-greedy.
- Influence of the error estimator especially for the POD-greedy algorithm

Test case 3

Burgers's equation with homogeneous Dirichlet boundary conditions

$$\frac{\partial}{\partial t}u(x,t,\xi) - \mu(\xi)\frac{\partial^2}{\partial x^2}u(x,t,\xi) + u(t,x,\xi)\frac{\partial}{\partial x}u(x,t,\xi) = \boldsymbol{b}(x)^T\boldsymbol{s}(t,\xi), \quad u(x,0) = 0,$$

- Domain $\Omega = (0,1)$ and time interval $I = (0,1)$
- $\mu(\xi) = \xi$ with $\xi \sim \mathcal{U}(0.01, 0.06)$.
- Source terme $\boldsymbol{s}(t) = (s_1(t), s_2(t))$, $\boldsymbol{b}(x) = (b_1(x), b_2(x))$ with

$$s_1(t) = \sin(4\pi t) \quad s_2(t) = \mathbf{1}_{[0.2,0.4]}(t),$$

$$b_1(x) = 4e^{-(\frac{x-0.2}{0.03})^2}\mathbf{1}_{[0.1,0.3]}(x), \quad b_2(x) = 4 \cdot \mathbf{1}_{[0.6,0.7]}(x)$$

Discretization : Finite difference scheme on a regular mesh with $d = 200$ nodes and time integration with semi-implicit Euler scheme on uniform discretization of I with $K = 300$

$$(\boldsymbol{I}_X - dt\boldsymbol{A}(\xi))\boldsymbol{u}^{k+1}(\xi) = \boldsymbol{u}^k(\xi) + dt[\boldsymbol{f}(\boldsymbol{u}^k) + \boldsymbol{B}\boldsymbol{s}(t^k)]$$

Results for T-greedy algorithm $r = 10$

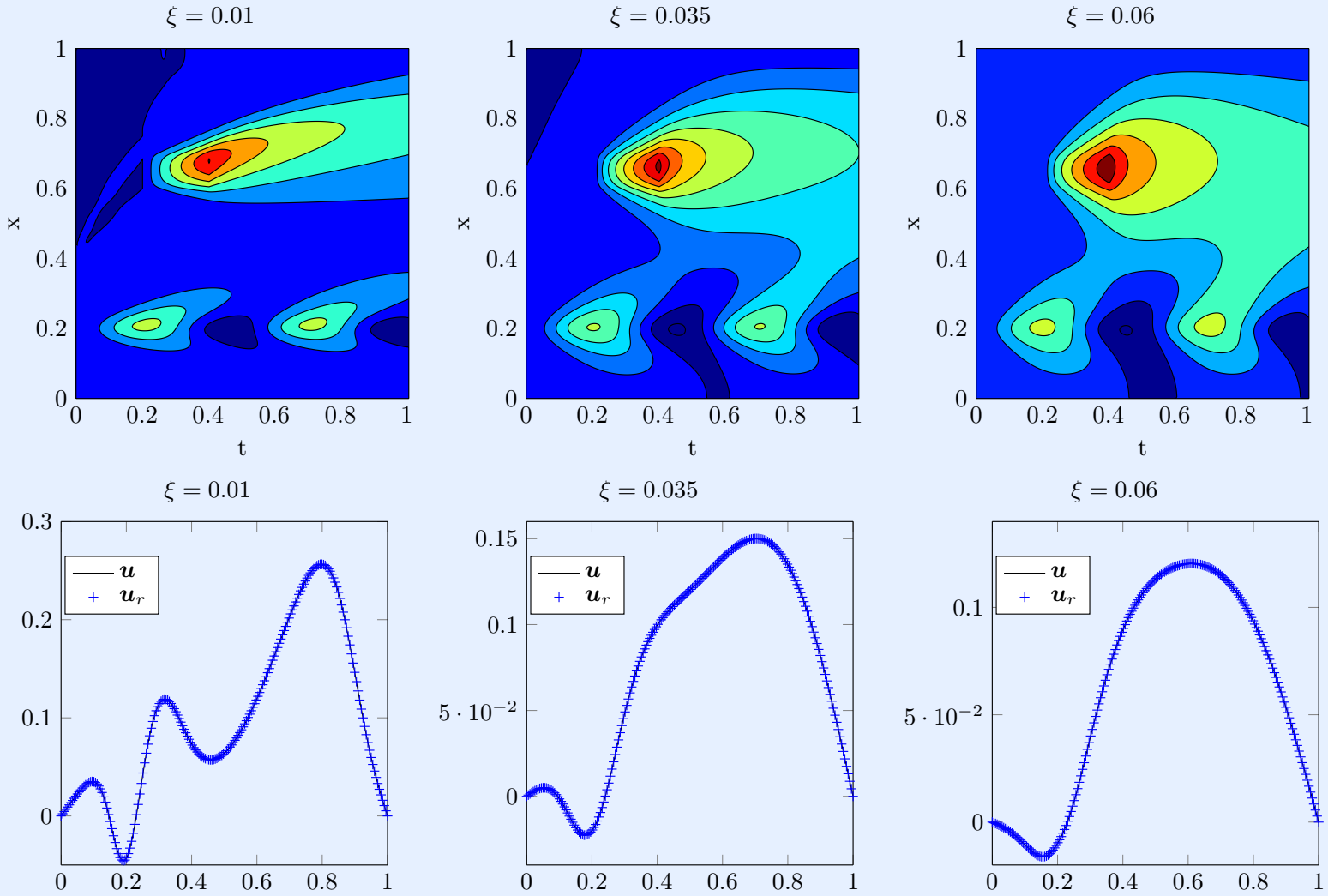


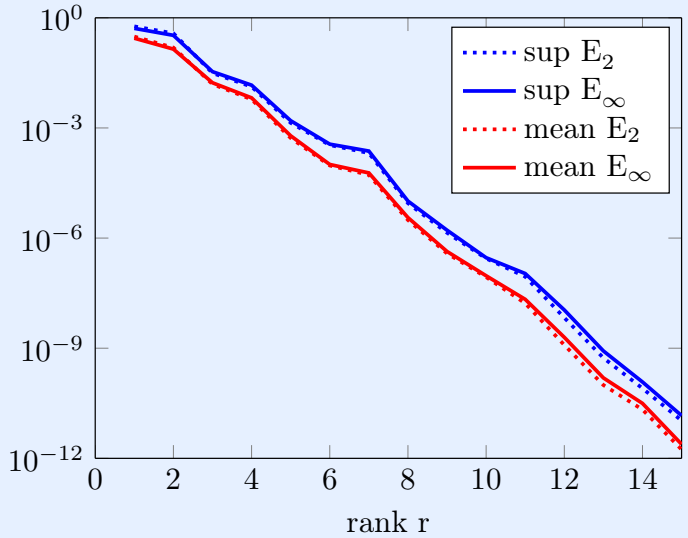
Fig.7 : Surface contours of $\boldsymbol{u}_r$ (top) and comparison to the exact solution at $T = 1$ (bottom)

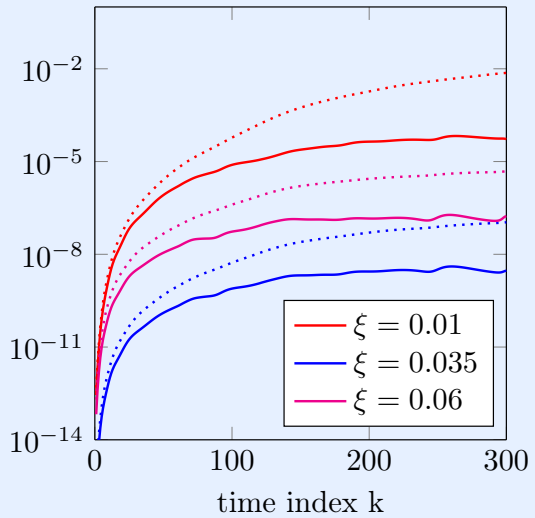
Study of the T-greedy algorithm

Greedy algorithm stopped for $r = r_{max}$

Data : $\#\Xi_{train} = 30$ and use of Δ_{NL}

Results for T-greedy algorithm $r = 10$





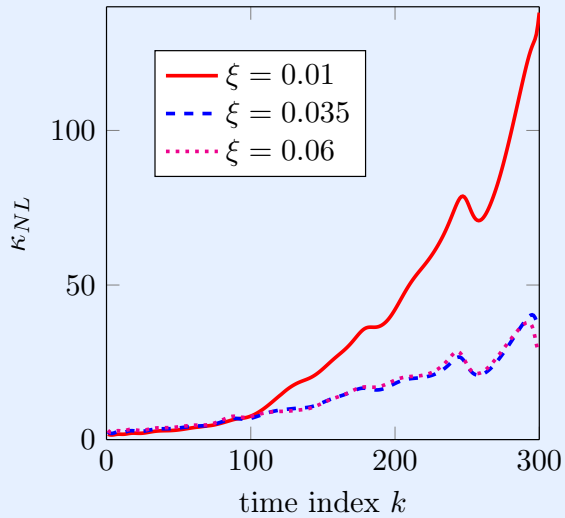


Fig. 8 Relative error with respect to r **Fig.9 :** Error est. (left) and effectivity index (right)

- Good convergence with rank
- Error estimate more pessimistic for this problem
- Influence of ξ on the error estimate

ξ	$\max_t \|\kappa_{NL}(t)\|_X$
0.01	110.064
0.035	26.8578
0.06	25.7706

Study of the T-greedy algorithm

Greedy alg. with stopping criterion in precision ε

	Strong greedy alg.			weak-greedy alg.		
ε	$\xi = 0.01$	$\xi = 0.035$	$\xi = 0.06$	$\xi = 0.01$	$\xi = 0.035$	$\xi = 0.06$
1e-1	4	4	4	7	7	6
1e-2	5	5	5	8	7	6
1e-3	7	8	7	9	9	8
1e-4	8	9	8	11	10	9
1e-5	10	11	9	12	12	10
1e-6	10	12	10	13	11	11

Tab. 6 : Reduced space dimension with respect to ε

- Richer reduced spaces needed (e.g. for $\xi = 0.01$)

Test case 4

Multi-parameter convection-reaction-diffusion with dirichlet boundary conditions

$$\frac{\partial}{\partial t}u(\boldsymbol{x},t,\boldsymbol{\xi})-\mu(\boldsymbol{\xi})\triangle u(\boldsymbol{x},t,\boldsymbol{\xi})+\boldsymbol{c}(\boldsymbol{x},\boldsymbol{\xi})\cdot\nabla u(\boldsymbol{x},t,\boldsymbol{\xi})+\kappa(\boldsymbol{\xi})u(\boldsymbol{x},t,\boldsymbol{\xi})=b(\boldsymbol{x}),\quad u(\boldsymbol{x},0)=0$$

- Domain $\Omega=(0,1)^2$ and time interval $I=(0,0.03)$
- $\mu(\boldsymbol{\xi})=1+0.2\xi_1$, $\boldsymbol{c}(\boldsymbol{x},\boldsymbol{\xi})=250(1+0.2\xi_2)(x_2-\frac{1}{2},\frac{1}{2}-x_1)^T$, $\kappa(\boldsymbol{\xi})=10(1+0.2\xi_3)$ $\boldsymbol{\xi}=(\xi_1,\ldots,\xi_3)$ where ξ_i are i.i.d. $\xi_i\sim\mathcal{U}(-1,1)$.
- Source term $b(\boldsymbol{x})=\mathbf{1}_{(0.7,0.8)^2}(\boldsymbol{x})$.

Discretization :

- $\mathbb{P}_1$ finite element approximation with $d=2212$ nodes

$$\mathcal{M}\frac{d}{dt}\mathsf{u}(t,\boldsymbol{\xi})=\mathcal{A}(\boldsymbol{\xi})\mathsf{u}(t,\boldsymbol{\xi})+\mathsf{b}\Rightarrow\frac{d}{dt}\boldsymbol{u}(t,\boldsymbol{\xi})=\boldsymbol{A}(\boldsymbol{\xi})\boldsymbol{u}(t,\boldsymbol{\xi})+\boldsymbol{b},$$

  with $\mathcal{M}=\mathcal{L}^T\mathcal{L}$ and $\mathcal{L}\mathsf{u}=\boldsymbol{u}$.
- Semi-implicit Euler scheme with $K=100$

$$(\boldsymbol{I}_d-dt\boldsymbol{A}(\boldsymbol{\xi}))\boldsymbol{u}^{k+1}(\boldsymbol{\xi})=\boldsymbol{u}^k(\boldsymbol{\xi})+dt\ \boldsymbol{b}$$

First results, $r = 10$ and $\boldsymbol{\xi} = (-0.2244, 0.6744, 0.5326)$

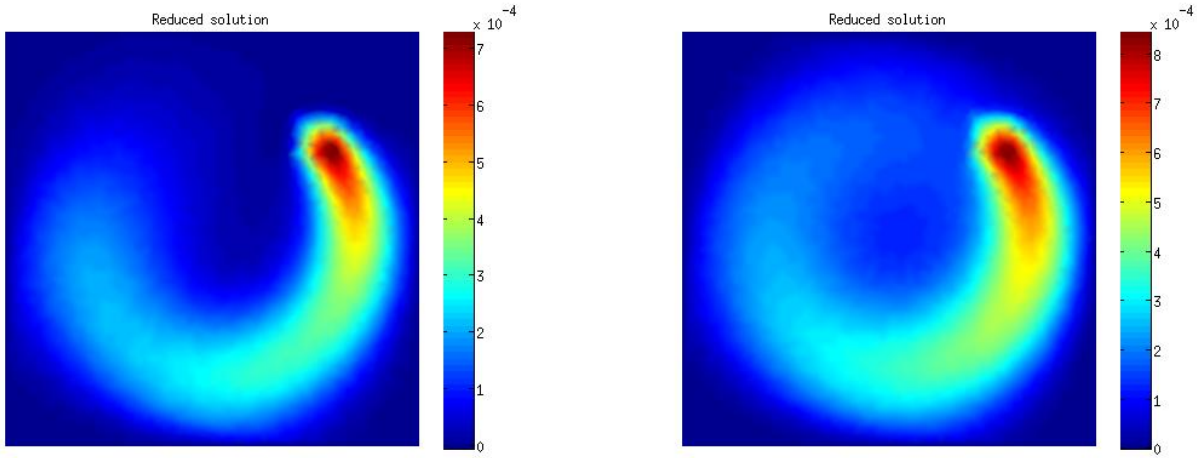


Fig.10 : $\boldsymbol{u}_r$ for $t = \frac{T}{2}$ (left) and $t = T$ (right)

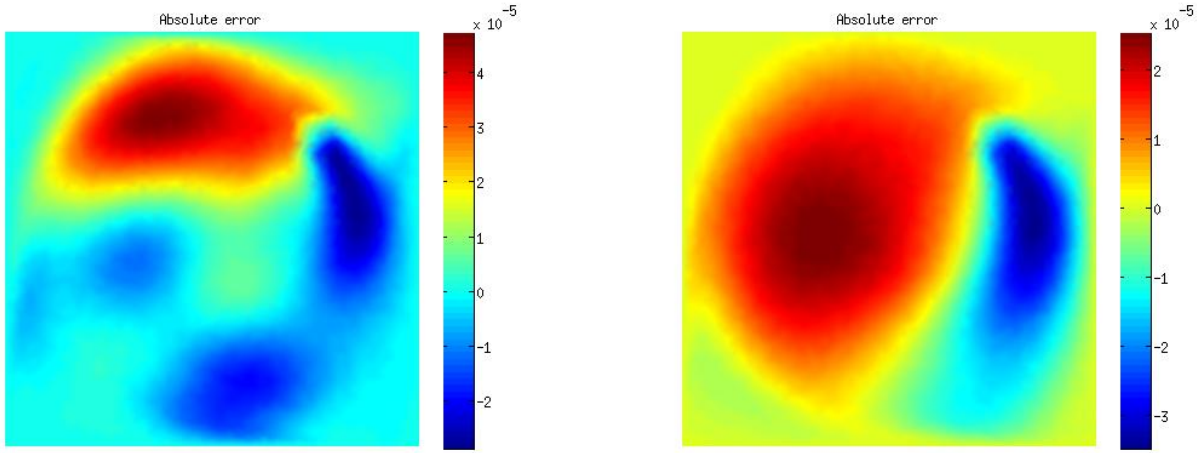


Fig.11 : Absolute error $\boldsymbol{u} - \boldsymbol{u}_r$

Conclusions & perspectives

T-greedy method : A low-rank approximation method, with sub-space point of view, for solving parameter dependent dynamical systems that can be interpreted as a RB method involving time-dependent reduced space.

- ✓ Practical method with efficient error estimates, validated for both linear and non-linear cases
- ✓ An alternative to POD-greedy method

Improvement and further explorations

- Efficient treatment of non-linearity : combining EIM and T-greedy strategies
- Analysis of the T-greedy algorithm

Conclusions & perspectives

T-greedy method : A low-rank approximation method, with sub-space point of view, for solving parameter dependent dynamical systems that can be interpreted as a RB method involving time-dependent reduced space.

- ✓ Practical method with efficient error estimates, validated for both linear and non-linear cases
- ✓ An alternative to POD-greedy method

Improvement and further explorations

- Efficient treatment of non-linearity : combining EIM and T-greedy strategies
- Analysis of the T-greedy algorithm

Thank you for your attention

Any questions ?

Bibliography

[Bin11] **P. Binev, A. Cohen, W. Dahmen, R. De Vore, G. Petrova, P. Wojtaszczyk** Convergence Rates for Greedy Algorithms in Reduced Basis Algorithms. *SIAM Journal on Mathematical Analysis*, 2011.

[Buff12] **A. Buffa, Y. Maday, A. Patera, C. Prud'homme, G. Turicini** A priori convergence of the greedy algorithm for the parametrized reduced basis. *ESAIM-Math. Model. Numer. Anal.*, 2012

[Haas11] **B. Haasdonk, M. Ohlberger, M.** Efficient reduced models and a posteriori error estimation for parametrized dynamical systems by offline/online decomposition. *Mathematical and Computer Modelling of Dynamical Systems*, 2011.

[Haas13] **B. Haasdonk** Convergence Rates of the POD-Greedy method. *M2AN Math. Model. Numer. Anal.*, 2013.

[Hom07] **C. Homescu, L.R. Petzold, R. Serban** Error Estimation for Reduced-Order Models of Dynamical Systems. *j-SIAM-REVIEW* 2007.

[Koch07] **O. Koch, C. Lubich** Dynamical Low-Rank Approximation. *SIAM Journal on Matrix Analysis and Applications*, 2007.

[Sap09] **T.P. Sapsis, P.F.J. Lermusiaux** Dynamically orthogonal field equations for continuous stochastic dynamical systems. *Physica D : Nonlinear Phenomena*, 2009.

[Wir14] **D. Wirtz, D.C. Sorensen, B. Haasdonk** A-posteriori error estimation for DEIM reduced nonlinear dynamical systems. *SIAM J. Sci. Comput.*, 2014.