Multiscale method with patches for the propagation of localized uncertainties in (semi-)linear elliptic and parabolic equations

Florent Pled

To cite this version:
Florent Pled. Multiscale method with patches for the propagation of localized uncertainties in (semi-)linear elliptic and parabolic equations. 2017. hal-01552196v2

HAL Id: hal-01552196
https://hal-upec-upem.archives-ouvertes.fr/hal-01552196v2
Submitted on 3 Jul 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Multiscale method with patches for the propagation of localized uncertainties in (semi-)linear elliptic and parabolic equations

Florent Pled\textsuperscript{1}

joint work with Mathilde Chevreuil\textsuperscript{2} and Anthony Nouy\textsuperscript{2}

\textsuperscript{1}Laboratoire de Modélisation et Simulation Multi-Echelle (MSME) Université Paris-Est, France
\textsuperscript{2}Institut de Recherche en Génie Civil et Mécanique (GeM) Ecole Centrale Nantes, France

LMSSC Seminar, June 30, 2017, Conservatoire National des Arts et Métiers (CNAM), Paris, France
Uncertainty quantification

- Linear transient model problem

\[ c\dot{u} - \nabla \cdot B(u, \nabla u) + C(u, \nabla u) = f \quad \text{on} \quad \Omega \]

with given initial conditions in \( \Omega \) and boundary conditions on \( \partial \Omega \), and where \( \dot{u} = \frac{\partial u}{\partial t} \)

- Input variabilities
  - Source term \( f \)
  - Evolution coefficient \( c \) and operators \( B, C \)
  - Geometrical domain \( \Omega \) and boundary \( \partial \Omega \)

- Parametric modeling of uncertainties
  - Finite set of random variables \( \xi = (\xi_1, \ldots, \xi_m) \)
  - Finite dimensional probability space \( (\Xi, B(\Xi), \mu) \) with \( \Xi \subset \mathbb{R}^m \)

Goal

- Propagate the uncertainties through the transient model

\[ c(\xi)\dot{u}(\xi) - \nabla \cdot B(u(\xi), \nabla u(\xi); \xi) + C(u(\xi), \nabla u(\xi); \xi) = f(\xi) \quad \text{on} \quad \Omega(\xi) \]
Stochastic spectral methods: Functional representation of random variables

- Functional expansion of the solution

\[ u(\xi) \approx \sum_{\alpha \in A} u_{\alpha} \psi_{\alpha}(\xi) \in L_{\mu}^{p}(\Xi; \mathcal{V}) \]

- Stochastic approximation space
  - Lagrange interpolation (or collocation) polynomial basis [Babuška, Nobile, and Tempone 2007; Xiu and Hesthaven 2005]
  - Piecewise polynomial basis (stochastic finite elements [Babuška, Tempone, and Zouraris 2004, 2005; Deb, Babuška, and Oden 2001; Wan and Karniadakis 2005], multi-wavelets [Le Maître et al. 2004a,b])

Advantages

- Explicit and accurate prediction of the solution in terms of the random variables
- Quick post-processing / Fast evaluation of quantities of interest
  - Statistical moments
  - Probability of events
  - Quantiles
  - Confidence regions
  - Sensitivity indices
  - ...
Multiscale problems with localized uncertainties on a subdomain called a patch

- **Uncertain** operators on local subdomains

\[
\begin{align*}
c(\cdot; \xi) &= \begin{cases} 
  c(\cdot) & \text{on } \Omega \setminus \Lambda \\
  c(\cdot; \xi) & \text{on } \Lambda(\xi)
\end{cases} \quad \text{(deterministic)} \\
B(\cdot; \xi) &= \begin{cases} 
  B(\cdot) & \text{on } \Omega \setminus \Lambda \\
  B(\cdot; \xi) & \text{on } \Lambda(\xi)
\end{cases} \\
C(\cdot; \xi) &= \begin{cases} 
  C(\cdot) & \text{on } \Omega \setminus \Lambda \\
  C(\cdot; \xi) & \text{on } \Lambda(\xi)
\end{cases}
\end{align*}
\]

- Dedicated strategy to handle high dimensionality and complexity

**Challenges**

- Develop a **multiscale method** that exploits the **localized** side of uncertainties
  - Simple **global** problem
    - Traditional solvers / Commercial software packages available in the industry
  - More technical **local** problems
    - Specific solvers / In-house research codes
Motivation

Multiscale domain decomposition method with patches


“Numerical zooms”

Main ingredient - Separation of global and local problems

- Possible refined local solutions with better approximation properties
- Simple global problems
  - Linear
  - No geometrical variabilities
  - Deterministic operator
- Appropriate approximation spaces and solvers to solve efficiently both problems
Outline

1. Context and motivation
2. Multiscale domain decomposition method
3. Computational aspects
4. Numerical illustrations
5. Conclusions and outlooks
Outline

1. Context and motivation

2. Multiscale domain decomposition method
   - Global-local formulation with patches
   - Global-local iterative algorithm

3. Computational aspects

4. Numerical illustrations

5. Conclusions and outlooks
Initial formulation

Domain featuring localized uncertainties and possible non-linearities

- Partition of random domain $\Omega(\xi)$ into random patch $\Lambda(\xi)$ and deterministic complementary subdomain $\Omega \setminus \Lambda$

Transient linear advection-diffusion-reaction stochastic problem - Formulation

Model problem with random material parameters and source terms

\[
\begin{cases}
  c\dot{u} - \nabla \cdot (K \nabla u) + \alpha \cdot \nabla u + Ru = f & \text{on } (\Omega \setminus \Lambda) \times I \\
  c(\xi)\dot{u} - \nabla \cdot (K(\xi) \nabla u) + \alpha(\xi) \cdot \nabla u + R(\xi)u = f(\xi) & \text{on } \Lambda(\xi) \times I
\end{cases}
\]

with possibly random boundary and initial conditions

\[
\begin{cases}
  u = 0 & \text{on } \Gamma_D \times I \\
  K \nabla u \cdot n = g & \text{on } (\Gamma_N \setminus \Gamma_N^\Lambda) \times I \\
  K(\xi) \nabla u \cdot n = g(\xi) & \text{on } \Gamma_N^\Lambda(\xi) \times I
\end{cases}
\quad \text{and} \quad
\begin{cases}
  u = u_0 & \text{on } (\Omega \setminus \Lambda) \times \{0\} \\
  u = u_0(\xi) & \text{on } \Lambda(\xi) \times \{0\}
\end{cases}
\]

where $\dot{u} = \frac{\partial u}{\partial t}$. 
Initial formulation

Domain featuring localized uncertainties and possible non-linearities

- Partition of random domain $\Omega(\xi)$ into random patch $\Lambda(\xi)$ and deterministic complementary subdomain $\Omega \setminus \Lambda$

Transient linear advection-diffusion-reaction stochastic problem - Space weak formulation

Find $u \in \mathcal{V}^\Xi$ such that it holds almost surely

$$
m_\Omega(\dot{u}(\xi), \delta u; \xi) + a_\Omega(u(\xi), \delta u; \xi) = \ell_\Omega(\delta u; \xi) \quad \forall \delta u \in \mathcal{V}
$$

$$
u|_{t=0+}(\xi) = u_0(\xi)
$$

with bilinear forms $a_\Omega(u, v; \xi) = \int_\Omega K(\xi) \nabla u \cdot \nabla v + \int_\Omega \alpha(\xi) \cdot \nabla u v + \int_\Omega R(\xi) uv,$

$m_\Omega(u, v; \xi) = \int_\Omega c(\xi) uv$ and linear form $\ell_\Omega(v; \xi) = \int_\Omega f(\xi)v + \int_{\Gamma_N} g(\xi)v.$
Global-local formulation with patches

**Domain decomposition - Introduction of a patch**

- Separation of the solution

\[ u = \begin{cases} 
U & \text{on } \Omega \setminus \Lambda \\
w & \text{on } \Lambda 
\end{cases} \]

with \( U = w \) on \( \Gamma \) (weak sense)
Global-local formulation with patches

Domain decomposition - Introduction of a patch

- **Uncertain** domain $\Omega$
- **Deterministic** subdomain $\Omega \setminus \Lambda$
- **Uncertain** patch $\Lambda$
- **Deterministic** interface $\Gamma$

Reformulation using Lagrange multiplier

Find $(U, w, \lambda) \in U^\Xi \times W^\Xi \times M^\Xi$ such that it holds almost surely

- **Global** equation on subdomain $\Omega \setminus \Lambda$

  \[ m_{\Omega \setminus \Lambda}(\dot{U}(\xi), \delta U) + a_{\Omega \setminus \Lambda}(U(\xi), \delta U) + b_\Gamma(\lambda(\xi), \delta U) = \ell_{\Omega \setminus \Lambda}(\delta U; \xi) \quad \forall \delta U \in U \]

- **Local** equation on patch $\Lambda$

  \[ m_{\Lambda}(\dot{w}(\xi), \delta w; \xi) + a_{\Lambda}(w(\xi), \delta w; \xi) - b_\Gamma(\lambda(\xi), \delta w) = \ell_{\Lambda}(\delta w; \xi) \quad \forall \delta w \in W \]

- **Coupling** equation on interface $\Gamma$

  \[ b_\Gamma(\delta \lambda, U(\xi)) - b_\Gamma(\delta \lambda, w(\xi)) = 0 \quad \forall \delta \lambda \in M \]
Global-local formulation with patches

### Overlapping domain decomposition - Introduction of a fictitious patch

- **Deterministic fictitious** domain $\tilde{\Omega}$
- **Deterministic fictitious** patch $\tilde{\Lambda}$
- **Uncertain** patch $\Lambda$
- **Deterministic** interface $\Gamma$

### Reformulation with overlapping domains

**Find** $(U, w, \lambda) \in \tilde{U}^\Xi \times W^\Xi \times M^\Xi$ **such that** it holds almost surely

- **Reformulated global equation** on fictitious domain $\tilde{\Omega} = (\Omega \setminus \Lambda) \cup \tilde{\Lambda}$

\[
\begin{align*}
  n_{\tilde{\Omega}}(\dot{U}(\xi), \delta U; \xi) - n_{\tilde{\Lambda}}(\dot{U}(\xi), \delta U; \xi) + c_{\tilde{\Omega}}(U(\xi), \delta U; \xi) - c_{\tilde{\Lambda}}(U(\xi), \delta U; \xi) + b_{\Gamma}(\lambda(\xi), \delta U) &= 0
\end{align*}
\]

- **Local equation** on patch $\Lambda$

\[
\begin{align*}
  m_{\Lambda}(\dot{w}(\xi), \delta w; \xi) + a_{\Lambda}(w(\xi), \delta w; \xi) - b_{\Gamma}(\lambda(\xi), \delta w) &= \ell_{\Lambda}(\delta w; \xi) \quad \forall \delta w \in W
\end{align*}
\]

- **Coupling equation** on interface $\Gamma$

\[
\begin{align*}
  b_{\Gamma}(\delta \lambda, U(\xi)) - b_{\Gamma}(\delta \lambda, w(\xi)) &= 0 \quad \forall \delta \lambda \in M
\end{align*}
\]
Global-local formulation with patches

Overlapping domain decomposition - Introduction of a fictitious patch

- Deterministic fictitious domain $\tilde{\Omega}$
- Deterministic fictitious patch $\tilde{\Lambda}$
- Uncertain patch $\Lambda$
- Deterministic interface $\Gamma$

Reformulation with overlapping domains

Find $(U, w, \lambda) \in \tilde{\mathcal{U}}^{\Xi} \times \mathcal{W}^{\Xi} \times \mathcal{M}^{\Xi}$ such that it holds almost surely

- Reformulated global equation on fictitious domain $\tilde{\Omega} = (\Omega \setminus \Lambda) \cup \tilde{\Lambda}$

$$n_{\tilde{\Omega}}(\dot{U}(\xi), \delta U; \xi) - n_{\tilde{\Lambda}}(\dot{U}(\xi), \delta U; \xi) + c_{\tilde{\Omega}}(U(\xi), \delta U; \xi) - c_{\tilde{\Lambda}}(U(\xi), \delta U; \xi) + b_{\Gamma}(\lambda(\xi), \delta U) = 0$$

where bilinear forms $n_{\mathcal{O}}(U, V; \xi) = \int_{\mathcal{O}} \tilde{c}(\xi)UV$ and

$$c_{\mathcal{O}}(U, V; \xi) = \int_{\mathcal{O}} \tilde{K}(\xi) \nabla U \cdot \nabla V + \int_{\mathcal{O}} \tilde{\alpha}(\xi) \cdot \nabla U V + \int_{\mathcal{O}} \tilde{R}(\xi)UV.$$
Global-local formulation with patches

Overlapping domain decomposition - Introduction of a fictitious patch

- Deterministic fictitious domain \( \widetilde{\Omega} \)
- Deterministic fictitious patch \( \widetilde{\Lambda} \)
- Uncertain patch \( \Lambda \)
- Deterministic interface \( \Gamma \)

Reformulation with overlapping domains

Find \((U, w, \lambda) \in \tilde{U}^\Xi \times \mathcal{W}^\Xi \times \mathcal{M}^\Xi\) such that it holds almost surely

- Reformulated global equation on fictitious domain \( \widetilde{\Omega} = (\Omega \setminus \Lambda) \cup \widetilde{\Lambda} \)

\[
n_{\Omega}(\dot{U}(\xi), \delta U; \xi) - n_{\Lambda}(\dot{U}(\xi), \delta U; \xi) + c_{\Omega}(U(\xi), \delta U; \xi) - c_{\Lambda}(U(\xi), \delta U; \xi) + b_{\Gamma}(\lambda(\xi), \delta U) = \ell_{\Omega}(\delta U; \xi)
\]

where bilinear forms \( n_{\mathcal{O}}(U, V; \xi) = \int_{\mathcal{O}} \tilde{c}(\xi) UV \) and

\[
c_{\mathcal{O}}(U, V; \xi) = \int_{\mathcal{O}} \tilde{K}(\xi) \nabla U \cdot \nabla V + \int_{\mathcal{O}} \tilde{a}(\xi) \cdot \nabla V U + \int_{\mathcal{O}} \tilde{R}(\xi) UV.
\]
Global-local iterative algorithm

Iterative coupling procedure

1. Global step: Compute \( \hat{U}^k \in \tilde{\mathcal{U}}^\Xi \) such that it satisfies almost surely

\[
\tilde{n}(\hat{U}^k(\xi), \delta U) + \tilde{c}(\hat{U}^k(\xi), \delta U) = \tilde{n}(\hat{U}^{k-1}(\xi), \delta U) + \tilde{c}(U^{k-1}(\xi), \delta U) - b(\lambda^{k-1}(\xi), \delta U) + \ell_{\tilde{\Omega} \setminus \Lambda}
\]
Global-local iterative algorithm

Iterative coupling procedure

1. **Global step**
   Compute \( \hat{U}^k \in \tilde{U}^\Xi \) such that it satisfies almost surely
   \[ n_{\Omega}(\hat{U}^k(\xi), \delta U) + c_{\Omega}(\hat{U}^k(\xi), \delta U) = n_{\Lambda}(\hat{U}^{k-1}(\xi), \delta U) + c_{\Lambda}(U^{k-1}(\xi), \delta U) - b_\Gamma(\lambda^{k-1}(\xi), \delta U) + \ell_{\Omega \setminus \Lambda} \]

2. **Relaxation step**
   Define \( U^k \in \tilde{U}^\Xi \) by
   \[ U^k(\xi) = \rho_k \hat{U}^k(\xi) + (1 - \rho_k)U^{k-1}(\xi) \text{ with relaxation parameter } \rho_k > 0 \]

\( \rightarrow \) Significant influence on the convergence properties of the algorithm
Global-local iterative algorithm

1. **Global step**: Compute \( \hat{U}^k \in \tilde{\Omega} \) such that it satisfies almost surely

\[
n_{\Omega}^{\sim}(\hat{U}^k(\xi), \delta U) + c_{\Omega}^{\sim}(\hat{U}^k(\xi), \delta U) = n_{\Lambda}^{\sim}(\hat{U}^{k-1}(\xi), \delta U) + c_{\Lambda}^{\sim}(U^{k-1}(\xi), \delta U) - b_{\Gamma}(\lambda^{k-1}(\xi), \delta U) + \ell_{\Omega \setminus \Lambda}(\delta U; \xi) \quad \forall \delta U \in \tilde{\Omega} \setminus \Lambda
\]

2. **Relaxation step**: Define \( U^k \in \tilde{\Omega} \) by

\[
U^k(\xi) = \rho_k \hat{U}^k(\xi) + (1 - \rho_k) U^{k-1}(\xi) \quad \text{with relaxation parameter} \quad \rho_k > 0
\]

3. **Local step**: Compute \( (w^k, \lambda^k) \in \mathcal{W}^{\infty} \times \mathcal{M}^{\infty} \) such that it satisfies almost surely

\[
\begin{align*}
m_{\Lambda}(w^k(\xi), \delta w; \xi) + a_{\Lambda}(w^k(\xi), \delta w; \xi) - b_{\Gamma}(\lambda^k(\xi), \delta w) &= \ell_{\Lambda}(\delta w; \xi) \quad \forall \delta w \in \mathcal{W} \\
b_{\Gamma}(\delta \lambda, w^k(\xi)) &= b_{\Gamma}(\delta \lambda, U^k(\xi)) \quad \forall \delta \lambda \in \mathcal{M}
\end{align*}
\]
**Iterative coupling procedure**

1. **Global step**: Compute \( \hat{U}^k \in \tilde{U}^\Xi \) such that it satisfies almost surely

\[
\begin{align*}
\tilde{n}_\Omega(\hat{U}^k(\xi), \delta U) + \tilde{c}_\Omega(\hat{U}^k(\xi), \delta U) &= \tilde{n}_\Lambda(\hat{U}^{k-1}(\xi), \delta U) + \tilde{c}_\Lambda(U^{k-1}(\xi), \delta U) - b_\Gamma(\lambda^{k-1}(\xi), \delta U) + \ell_{\Omega \setminus \Lambda} \\
\end{align*}
\]

2. **Relaxation step**: Define \( U^k \in \tilde{U}^\Xi \) by

\[
U^k(\xi) = \rho_k \hat{U}^k(\xi) + (1 - \rho_k)U^{k-1}(\xi) \quad \text{with relaxation parameter } \rho_k > 0
\]

3. **Local step**: Compute \((w^k, \lambda^k) \in \mathcal{W}^\Xi \times \mathcal{M}^\Xi\) such that it satisfies almost surely

\[
\begin{align*}
\begin{cases}
m_\Lambda(\hat{w}^k(\xi), \delta w; \xi) + a_\Lambda(w^k(\xi), \delta w; \xi) - b_\Gamma(\lambda^k(\xi), \delta w) &= \ell_\Lambda(\delta w; \xi) \quad \forall \delta w \in \mathcal{W} \\
b_\Gamma(\delta \lambda, w^k(\xi)) &= b_\Gamma(\delta \lambda, U^k(\xi)) \quad \forall \delta \lambda \in \mathcal{M}
\end{cases}
\end{align*}
\]
Consistency

- There exists a unique extension $\tilde{U}(\xi) \in \tilde{U}_*$ of the global solution $U(\xi) \in \tilde{U}$ such that $\tilde{U}(\xi) = U(\xi)$ on complementary subdomain $\Omega \setminus \Lambda$ and the restriction of $\tilde{U}(\xi)$ to fictitious patch $\tilde{\Lambda}$ is uniquely defined by the trace of $U(\xi)$ on interface $\Gamma$.

- The extended global solution $U \in L^p_\mu(\Xi; \tilde{U})$, with exponent $2 \leq p \leq +\infty$.

Convergence

- Assume that the sequence of relaxation parameters $\{\rho_k\}_{k \in \mathbb{N}}$ is such that
  \[ 0 < \rho_{\text{inf}} \leq \rho_k \leq \rho_{\text{sup}} < +\infty, \]
  for some strictly positive constants $\rho_{\text{inf}}$ and $\rho_{\text{sup}}$ independent of $\xi$ and $k$.

- Then, for $\rho_{\text{sup}}$ sufficiently small, the sequence $\{(U^k(\xi), w^k(\xi), \lambda^k(\xi))\}_{k \in \mathbb{N}}$ converges almost surely to the unique solution $(U(\xi), w(\xi), \lambda(\xi))$ of the initial global-local problem in $\tilde{U}_*(\xi) \times \mathcal{W} \times \mathcal{M}$. $\leftrightarrow$ Almost sure convergence

- Also, the sequence $\{(U^k, w^k, \lambda^k)\}_{k \in \mathbb{N}}$ converges to $(U, w, \lambda)$ in $L^p_\mu(\Xi; \tilde{U}) \times L^p_\mu(\Xi; \mathcal{W}) \times L^p_\mu(\Xi; \mathcal{M})$. $\leftrightarrow$ Convergence in $L^p_\mu$
Robustness - Impact of approximations

- Analysis of the sensitivity of the iterative algorithm with respect to approximations introduced at the different steps of the algorithm.

- Control of the approximation error $U^k_\varepsilon - U^k$ between the approximate/perturbed global iterate $U^k_\varepsilon$ and the unperturbed one $U^k$ at iteration $k$ in a $L^p_\mu$-norm.

- Under certain assumptions on the unperturbed and perturbed iteration maps, if the initial global iterate $U^0_\varepsilon$ is in a $\delta$-neighborhood $\mathcal{V}_\delta$ of the exact global solution $U$, then all global iterates $U^k_\varepsilon$ remain in $\mathcal{V}_\delta$, and the approximate sequence $\{U^k_\varepsilon\}_{k \in \mathbb{N}}$ tends to a $\gamma(\varepsilon)$-neighborhood of $U$ in $L^p_\mu(\Xi; \tilde{U})$ with $\gamma(\varepsilon) \to 0$ as $\varepsilon \to 0$.

- Open ball $\mathcal{V}_\delta$ of radius $\delta\|U\|_{L^p_\mu(\Xi; \tilde{U})}$ centered at the exact global solution $U$ and containing all global iterates $U^k_\varepsilon$. 

\[ U^0_\varepsilon \quad \mathcal{V}_\delta \quad U \quad U^k_\varepsilon \]
Outline

1. Context and motivation

2. Multiscale domain decomposition method

3. Computational aspects
   - Adaptive sampling-based approach
   - Relaxation step

4. Numerical illustrations

5. Conclusions and outlooks
Computational aspects

Global problem
- Direct computation of global problem by solving a system of uncoupled linear equations with the same deterministic global finite element matrix
- Use of traditional (parallel) solvers available in commercial deterministic finite element codes

Local problems
- Resolution of stochastic local problems using an adaptive least-squares method
- Sampling-based approaches require evaluations of the solution by standard deterministic codes and are well adapted to massively parallel computation
- Control of the stability of local approximations by increasing the number of samples
- Estimation of the approximation error using cross-validation techniques
- Working sets strategies proposed in [Chkifa, Cohen, and Schwab 2014, 2015; Chkifa et al. 2013] for the construction of an increasing sequence of sparse approximation spaces
- Algorithm with adaptive random sampling and an adaptive selection of approximation spaces for the construction of sparse polynomial approximations of local solutions [Nouy and Pled 2017]
Least-squares method

Least-squares approximation in a given subspace

- Least-squares approximation $v \in \mathbb{R}^n \otimes S_A$ of a random vector $u \in \mathbb{R}^n \otimes L^2_\mu(\Xi)$:
  
  $$v(\xi) = \sum_{\alpha \in A} v_\alpha \psi_\alpha(\xi)$$

  where $S_A = \text{span}\{\psi_\alpha\}_{\alpha \in A}$ is a polynomial space for a given subset $A$ of multi-indices $\alpha$.

- Set of coefficients $V = (v_\alpha)_{\alpha \in A} \in \mathbb{R}^{n \times \#A}$ solution of an unconstrained optimization problem:
  
  $$\min_{(v_\alpha)_{\alpha \in A}} \sum_{l=1}^N \|u(\xi^l) - \sum_{\alpha \in A} v_\alpha \psi_\alpha(\xi^l)\|^2_2 \iff \min_V \|Y - \Psi V^T\|^2_F$$

  $$\implies V^T = (\Psi^T \Psi)^{-1} \Psi^T Y$$

  where $\Psi = (\psi_\alpha(\xi^l))_{1 \leq l \leq N, \alpha \in A} \in \mathbb{R}^{N \times \#A}$ and $Y = (u(\xi^l))_{1 \leq l \leq N} \in \mathbb{R}^{N \times n}$.

- In practice, improvement of the stability of the least-squares approximation $v$ by increasing the number of samples $N$. 


Working set strategy for adaptive approximation

Structure of monotone sets

- A finite subset $\mathcal{A}$ of multi-indices $\alpha$ is called monotone (or lower or downward closed) if it is such that $(\beta \in \mathcal{A} \text{ and } \alpha \leq \beta) \implies \alpha \in \mathcal{A}$

- Margin $\mathcal{M}(\mathcal{A})$ and reduced margin $\mathcal{M}_{\text{red}}(\mathcal{A})$ of a monotone set $\mathcal{A}$

  \[
  \mathcal{M}(\mathcal{A}) = \{\alpha \notin \mathcal{A} ; \exists i \in \{1, \ldots, m\} \text{ such that } \alpha_i \neq 0 \implies \alpha - e_i \in \mathcal{A}\}
  \]

  \[
  \mathcal{M}_{\text{red}}(\mathcal{A}) = \{\alpha \notin \mathcal{A} ; \forall i \in \{1, \ldots, m\} \text{ such that } \alpha_i \neq 0 \implies \alpha - e_i \in \mathcal{A}\}
  \]

**Figure:** Montone multi-index set $\mathcal{A}$, its maximal elements $\max(\mathcal{A})$, its margin $\mathcal{M}(\mathcal{A})$ and reduced margin $\mathcal{M}_{\text{red}}(\mathcal{A})$
Working set strategy for adaptive approximation

Working set strategy - construction of an increasing sequence of monotone sets

- Polynomial function \((d = 3)\) [Sudret 2008]

\[
u(\xi) = \frac{1}{2^d} \prod_{j=1}^{d} (3\xi_j^2 + 1)
\]

where \(\xi_j \sim \mathcal{U}(0, 1)\) and \(\xi = (\xi_j)_{j=1}^{d} \in \Xi = (0, 1)^d\)
Working set strategy for adaptive approximation

Working set strategy - construction of an increasing sequence of monotone sets

- **Ishigami function** [Blatman and Sudret 2010; Blatman and Sudret 2011; Saltelli Andrea and Scott 2000; Sudret 2008]

\[ u(\xi) = \sin(\xi_1) + a\sin(\xi_2)^2 + b\xi_3^4 \sin(\xi_1) \quad \text{with } a = 7, \ b = 0.1 \]

where \( \xi_j \sim \mathcal{U}(-\pi, \pi) \) and \( \xi = (\xi_j)_{j=1}^3 \in \Xi = (-\pi, \pi)^3 \)
Working set strategy for adaptive approximation

Working set strategy - construction of an increasing sequence of monotone sets

- Anisotropic function [Chkifa, Cohen, and Schwab 2014]

\[ u(\xi) = \xi_3 \sin(\xi_4) + \xi_{16} \]

where \( \xi_j \sim \mathcal{U}(0, 1) \) and \( \xi = (\xi_j)_{j=1}^{16} \in \Xi = (0, 1)^{16} \)
Working set strategy for adaptive approximation

Working set strategy - construction of an increasing sequence of monotone sets

- **Anisotropic function** [Chkifa, Cohen, and Schwab 2014]

\[
u(\xi) = \xi_3 \sin(\xi_4) + \xi_{16}
\]

where \(\xi_j \sim \mathcal{U}(0,1)\) and \(\xi = (\xi_j)_{j=1}^{16} \in \Xi = (0,1)^{16}\)
Cross-validation technique

*Leave-one-out cross-validation technique: statistical method based on random partitioning of the available sample set [Cawley and Talbot 2004]*

**Input:** Coefficients $V = (v_\alpha)_{\alpha \in A}$ of the approximation $v(\xi) = \sum_{\alpha \in A} v_\alpha \psi_\alpha(\xi)$ of $u(\xi)$, and matrices $\Psi = (\psi_\alpha(\xi^l))_{1 \leq l \leq N, \alpha \in A}$ and $Y^T = (u(\xi^l))_{1 \leq l \leq N}$

**Output:** Vector $\varepsilon = (\varepsilon_i)_{i \in I}$, where $\varepsilon_i$ is an estimation of the error $\mathbb{E}((u_i(\xi) - v_i(\xi))^2)$

1: for $i = 1, \ldots, n$ do
2: Compute the set of predicted residuals $\{\Delta_i^l\}_{l=1}^N$ associated with sample set $\{\xi^l\}_{l=1}^N$ using Sherman-Morrison-Woodbury formula:

   $\Delta_i^l = \frac{r_{li}}{1 - h_l}$ with
   \[
   \begin{cases}
   r_{li} & \text{the } (l, i)\text{-th entry of matrix } R = \Psi V^T - Y \\
   h_l & \text{the } l\text{-th diagonal term in matrix } H = \Psi (\Psi^T \Psi)^{-1} \Psi^T
   \end{cases}
   \]

3: Compute the leave-one-out error $e_i = \frac{E_i}{\hat{m}_2(Y_i)}$, where $E_i = \frac{1}{N} \sum_{l=1}^N (\Delta_i^l)^2$ and $\hat{m}_2(Y_i)$ is the empirical second moment of the $i$-th column $Y_i = (u_i(\xi^l))_{l=1}^N$ of matrix $Y$

4: Compute the corrected leave-one-out error $\varepsilon_i = e_i \times T(A, N)$, where

   $T(A, N) = \left(1 - \frac{\#A}{N}\right)^{-1} \left(1 + \frac{\text{tr}(\hat{C}^{-1})}{N}\right)$ is a correction factor allowing to reduce the sensitivity to overfitting [Blatman and Sudret 2011; Chapelle, Vapnik, and Bengio 2002] and where $\hat{C} = \frac{1}{N} \Psi^T \Psi$ is the empirical covariance matrix of $(\psi_\alpha(\xi))_{\alpha \in A}$

5: end for
Adaptive sparse algorithm

Adaptive sparse least-squares solver with random sampling and working set strategy
[Nouy and Pled 2017]

**Input:** Initial number of samples $N \geq 1$, sampling factor $p_{\text{add}} > 0$, parameter $\theta \in [0, 1]$

**Output:** Monotone set $A$ and coefficients $V = (v_{\alpha})_{\alpha \in A}$ of the least-squares approximation $v(\xi) = \sum_{\alpha \in A} v_{\alpha} \psi_{\alpha}(\xi)$ of $u(\xi)$

1: Start with null initial set $A = \{0_F\}$
2: Generate the initial sample set $\{\xi^l\}_{l=1}^N$ randomly
3: Compute the matrices $\Psi_A = (\psi_{\alpha}(\xi^l))_{1 \leq l \leq N, \alpha \in A}$ and $Y^T = (u(\xi^l))_{1 \leq l \leq N}$
4: **while** no convergence **do**
5:   // Adaptive random sampling
6:      **while** no convergence and no stagnation **do**
7:         Generate the additional sample set $\{\xi^{N+l}\}_{l=1}^{N_{\text{add}}}$ randomly, with $N_{\text{add}} = \text{ceil}(p_{\text{add}}N)$
8:         Compute the matrices $\Psi_{A,\text{add}}$ and $Y_{\text{add}}^T$
9:         Update the number of samples $N \leftarrow N + N_{\text{add}}$ and the matrices $\Psi_A^T \leftarrow (\Psi_A^T, \Psi_{A,\text{add}}^T)$ and $Y^T \leftarrow (Y^T, Y_{\text{add}}^T)$
10:    Compute the coefficients $V = (v_{\alpha})_{\alpha \in A}$ such that $V^T = (\Psi_A^T \Psi_A)^{-1} \Psi_A^T Y$
11:    **end while**
12: **end while**
Adaptive sparse algorithm

Adaptive sparse least-squares solver with random sampling and working set strategy
[Nouy and Pled 2017]

Input: Initial number of samples $N \geq 1$, sampling factor $p_{\text{add}} > 0$, parameter $\theta \in [0, 1]$
Output: Monotone set $\mathcal{A}$ and coefficients $V = (v_\alpha)_{\alpha \in \mathcal{A}}$ of the least-squares approximation

$v(\xi) = \sum_{\alpha \in \mathcal{A}} v_\alpha \psi_\alpha(\xi)$ of $u(\xi)$

1: Start with null initial set $\mathcal{A} = \{0\}$
2: Generate the initial sample set $\{\xi_l\}_{l=1}^N$ randomly
3: Compute the matrices $\Psi_\mathcal{A} = (\psi_\alpha(\xi_l))_{1 \leq l \leq N, \alpha \in \mathcal{A}}$ and $Y^T = (u(\xi_l))_{1 \leq l \leq N}$
4: while no convergence do
5:     // Adaptive random sampling
6:     while no convergence and no stagnation do
7:         Compute the coefficients $V = (v_\alpha)_{\alpha \in \mathcal{A}}$ such that $V^T = (\Psi_\mathcal{A}^T \Psi_\mathcal{A})^{-1} \Psi_\mathcal{A}^T Y$
        by taking into account additional samples
8:     end while
9: end while
Adaptive sparse algorithm

Adaptive sparse least-squares solver with random sampling and working set strategy

[Nouy and Pled 2017]

**Input:** Initial number of samples $N \geq 1$, sampling factor $p_{\text{add}} > 0$, parameter $\theta \in [0, 1]$

**Output:** Monotone set $\mathcal{A}$ and coefficients $V = (v_\alpha)_{\alpha \in \mathcal{A}}$ of the least-squares approximation

$v(\xi) = \sum_{\alpha \in \mathcal{A}} v_\alpha \psi_\alpha(\xi)$ of $u(\xi)$

1: Start with null initial set $\mathcal{A} = \{0_F\}$
2: Generate the initial sample set $\{\xi^l\}_{l=1}^N$ randomly
3: Compute the matrices $\Psi_\mathcal{A} = (\psi_\alpha(\xi^l))_{1 \leq l \leq N, \alpha \in \mathcal{A}}$ and $Y^T = (u(\xi^l))_{1 \leq l \leq N}$
4: while no convergence do
5:   // Adaptive random sampling
6:     while no convergence and no stagnation do
7:         Compute the coefficients $V = (v_\alpha)_{\alpha \in \mathcal{A}}$ such that $V^T = (\Psi_\mathcal{A}^T \Psi_\mathcal{A})^{-1} \Psi_\mathcal{A}^T Y$
8:         by taking into account additional samples
9:     end while
10:    // Working set strategy
11:    while no convergence and no overfitting do
12:        Compute the reduced margin $\mathcal{M} = M_{\text{red}}(\mathcal{A})$ of $\mathcal{A}$ and the set $\mathcal{T} = \mathcal{A} \cup \mathcal{M}$
13:        Compute the matrices $\Psi_\mathcal{M}$ and $\Psi_\mathcal{T} = (\Psi_\mathcal{A}, \Psi_\mathcal{M})$
14:        Compute the coefficients $V = (v_\alpha)_{\alpha \in \mathcal{T}}$ such that $V^T = (\Psi_\mathcal{T}^T \Psi_\mathcal{T})^{-1} \Psi_\mathcal{T}^T Y$
15:        Define the smallest (monotone) subset $\mathcal{N}$ of $\mathcal{M}$ such that $e(\mathcal{N}) \geq \theta e(\mathcal{M})$, with $e(\mathcal{N}) = \sum_{\alpha \in \mathcal{N}} \|v_\alpha\|^2_2$ and $e(\mathcal{M}) = \sum_{\alpha \in \mathcal{M}} \|v_\alpha\|^2_2$
16:        Update the multi-index set $\mathcal{A} \leftarrow \mathcal{A} \cup \mathcal{N}$ and the matrix $\Psi_\mathcal{A} \leftarrow (\Psi_\mathcal{A}, \Psi_\mathcal{N})$
17:        Compute the coefficients $V = (v_\alpha)_{\alpha \in \mathcal{A}}$ such that $V^T = (\Psi_\mathcal{A}^T \Psi_\mathcal{A})^{-1} \Psi_\mathcal{A}^T Y$
18:    end while
Adaptive sparse algorithm

Adaptive sparse least-squares solver with random sampling and working set strategy
[Nouy and Pled 2017]

**Input:** Initial number of samples $N \geq 1$, sampling factor $p_{\text{add}} > 0$, parameter $\theta \in [0,1]$

**Output:** Monotone set $\mathcal{A}$ and coefficients $V = (v_{\alpha})_{\alpha \in \mathcal{A}}$ of the least-squares approximation $v(\xi) = \sum_{\alpha \in \mathcal{A}} v_{\alpha} \psi_{\alpha}(\xi)$ of $u(\xi)$

1: Start with null initial set $\mathcal{A} = \{0\}$
2: Generate the initial sample set $\{\xi^l\}^N_{l=1}$ randomly
3: Compute the matrices $\Psi_\mathcal{A} = (\psi_{\alpha}(\xi^l))_{1 \leq l \leq N, \alpha \in \mathcal{A}}$ and $Y^T = (u(\xi^l))_{1 \leq l \leq N}$
4: while no convergence do
5:  // Adaptive random sampling
6:  while no convergence and no stagnation do
7:    Compute the coefficients $V = (v_{\alpha})_{\alpha \in \mathcal{A}}$ such that $V^T = (\Psi_\mathcal{A}^T \Psi_\mathcal{A})^{-1} \Psi_\mathcal{A}^T Y$
    by taking into account additional samples
8:  end while
9:  // Working set strategy
10: while no convergence and no overfitting do
11:    Compute the coefficients $V = (v_{\alpha})_{\alpha \in \mathcal{A}}$ such that $V^T = (\Psi_\mathcal{A}^T \Psi_\mathcal{A})^{-1} \Psi_\mathcal{A}^T Y$
    by updating the approximation basis
12: end while
13: end while
Impact of relaxation step on convergence

- The relaxation step may affect the convergence rate of the iterative algorithm.

Simplest technique - Fixed relaxation parameter

- A large relaxation parameter may speed up the convergence but can lead to a divergence of the algorithm, while a small relaxation parameter enables to control the convergence but trigger more iterations in return.
- The computation of an optimal relaxation parameter is not obvious in the non-linear framework.
- Even in the linear case, the optimal fixed value of relaxation parameter $\rho$ is problem-dependent and not known a priori.
- In the linear case and under some conditions on the iteration map, an optimal convergent rate of the global-local iterative algorithm can be achieved by computing the optimal relaxation parameter $\rho_{opt}$ defined by [Xu 1992]

$$\rho_{opt} = \frac{2}{\lambda_{\text{min}} + \lambda_{\text{max}}},$$

where $\lambda_{\text{min}}$ and $\lambda_{\text{max}}$ are the lowest and largest eigenvalues of the iteration map (see [Chevreuil, Nouy, and Safatly 2013]).
Convergence acceleration technique - Aitken’s dynamic relaxation

- **Aitken’s Delta-Squared method** [Irons and Tuck 1969; Macleod 1986]
  - Simple and efficient convergence acceleration technique
  - Improvement of the current solution by using information gained at two previous iterations

- **Update Aitken’s formula for the relaxation parameter** [Duval et al. 2014; Küttler and Wall 2008; Liu, Sun, and Fan 2014]

\[
\begin{cases}
\rho_1 = \rho_2 = 1 \\
\rho_k = -\rho_{k-1} \frac{\langle \delta^k(\xi) - \delta^{k-1}(\xi), \delta^{k-1}(\xi) \rangle_{\tilde{U}}}{\| \delta^k(\xi) - \delta^{k-1}(\xi) \|_{\tilde{U}}^2}, & \text{with} & \delta^k(\xi) = \hat{U}^k(\xi) - U^{k-1}(\xi)
\end{cases}
\]

- **Adapted recursive formula** to ensure convergence of the algorithm [Nouy and Pled 2017]

\[
\rho_k = T_{[\rho_{\inf}, \rho_{\sup}]} \left( -\rho_{k-1} \frac{\langle \delta^k(\xi) - \delta^{k-1}(\xi), \delta^{k-1}(\xi) \rangle_{\tilde{U}}}{\| \delta^k(\xi) - \delta^{k-1}(\xi) \|_{\tilde{U}}^2} \right),
\]

where \( T_{[\rho_{\inf}, \rho_{\sup}]}(\rho) \) is the projection of \( \rho \) on the interval \([\rho_{\inf}, \rho_{\sup}]\)
Outline

1. Context and motivation
2. Multiscale domain decomposition method
3. Computational aspects
4. Numerical illustrations
   - Stationary non-linear diffusion-reaction stochastic problem
   - Transient linear advection-diffusion-reaction deterministic problem
5. Conclusions and outlooks
Stationary non-linear diffusion-reaction stochastic problem

Non-linear diffusion-reaction problem with localized random material heterogeneities
[Nouy and Pled 2017]

\[-\nabla \cdot (K \nabla u) + Ru^3 = 1 \quad \text{on } \Omega = (0, 2) \times (0, 16)\]
\[u = 0 \quad \text{on } \Gamma_D = \partial \Omega\]

- **Random** diffusion and reaction fields

\[K(x, \xi) = \begin{cases} 1 & \text{on } \Omega \setminus \left( \bigcup_{q=1}^8 \Lambda_q \right) \\ 1 + \gamma_q \xi^{2q-1} \chi_q(x) & \text{on } \Lambda_q \end{cases}\]
\[R(x, \xi) = \begin{cases} 0 & \text{on } \Omega \setminus \left( \bigcup_{q=1}^8 \Lambda_q \right) \\ \gamma_q \xi^{2q} \chi_q(x) & \text{on } \Lambda_q \end{cases}\]

- **Uniform** random variables \( \xi = (\xi_i)_{i=1}^{16} \) on \( \Xi_i = (0, 1) \)
- **Uniform** weights \( \gamma_q = 1 \)
Non-linear diffusion-reaction problem with localized random material heterogeneities

[Nouy and Pled 2017]

\[-\nabla \cdot (K \nabla u) + Ru^3 = 1 \quad \text{on } \Omega = (0, 2) \times (0, 16)\]

\[u = 0 \quad \text{on } \Gamma_D = \partial \Omega\]

- **Random** diffusion and reaction fields

\[K(x, \xi) = \begin{cases} 
1 & \text{on } \Omega \setminus \left( \bigcup_{q=1}^{8} \Lambda_q \right) \\
1 + \gamma_q \xi_{2q-1} \chi_q(x) & \text{on } \Lambda_q
\end{cases}\]

\[R(x, \xi) = \begin{cases} 
0 & \text{on } \Omega \setminus \left( \bigcup_{q=1}^{8} \Lambda_q \right) \\
\gamma_q \xi_{2q} \chi_q(x) & \text{on } \Lambda_q
\end{cases}\]

- **Uniform** random variables \(\xi = (\xi_i)_{i=1}^{16}\) on \(\Xi_i = (0, 1)\)

- **Heterogeneous** weights \(\gamma_q = 1 - 0.1(q + 1)\)
Stationary non-linear diffusion-reaction stochastic problem

Non-linear diffusion-reaction problem with localized random material heterogeneities
[Nouy and Pled 2017]

\[-\nabla \cdot (K \nabla u) + Ru^3 = 1 \quad \text{on } \Omega = (0, 2) \times (0, 16)\]
\[u = 0 \quad \text{on } \Gamma_D = \partial \Omega\]

- **Random** diffusion and reaction fields

\[K(x, \xi) = \begin{cases} 
1 & \text{on } \Omega \setminus \left( \bigcup_{q=1}^{8} \Lambda_q \right) \\
1 + \gamma_q \xi_{2q-1} \chi_q(x) & \text{on } \Lambda_q 
\end{cases}\]
\[R(x, \xi) = \begin{cases} 
0 & \text{on } \Omega \setminus \left( \bigcup_{q=1}^{8} \Lambda_q \right) \\
\gamma_q \xi_{2q} \chi_q(x) & \text{on } \Lambda_q 
\end{cases}\]

- **Uniform** random variables \(\xi = (\xi_i)_{i=1}^{16}\) on \(\Xi_i = (0, 1)\)
- **Heterogeneous** weights \(\gamma_q = 1 - 0.1(q + 1)\)

Approximation spaces

- **Spatial level**: nested global and local finite element approximation spaces
  - \(\dim(\tilde{U}_H) = 3381\) on \(\tilde{\Omega}\)
  - \(\dim(\mathcal{W}_{\eta}^q) = 441\) on \(\Lambda_q\)

- **Stochastic level**: adaptive construction of sparse Legendre polynomial approximation spaces using working sets strategies [Chkifa et al. 2013]
Convergence analysis

- **Control of the accuracy** of global approximations $U^k$ with respect to a global reference solution $U^{ref}$ using the relative error indicator

$$\varepsilon_{\Xi,\Omega \setminus \Lambda}(U^k; U^{ref}) = \frac{\|U^k - U^{ref}\|_{L^2_{\mu}(\Xi; L^2(\Omega \setminus \Lambda))}}{\|U^{ref}\|_{L^2_{\mu}(\Xi; L^2(\Omega \setminus \Lambda))}}$$

with $\|U\|_{L^2_{\mu}(\Xi; L^2(\Omega \setminus \Lambda))} = \mathbb{E}(\|U(\xi)\|_{L^2(\Omega \setminus \Lambda)}^2)$

- **Reference solution** $(U^{ref}, w^{ref}_q, \lambda^{ref}_q)$ of the full-scale coupled problem obtained using an adaptive sparse least-squares method
  - **Spatial level**: deterministic non-linear problems solved using a tangent-Newton iterative algorithm with a tolerance $\epsilon = 10^{-12}$
  - **Stochastic level**: control of the number of samples $N^{ref}$ and approximation spaces $S_{A^{ref}}$ using a cross-validation procedure with a tolerance $\varepsilon_{cv}^{ref} = 10^{-6}$
  - **Isotropic case**: $N^{ref} = 795$ samples, $\#A^{ref} = 384$ for $U^{ref}$
  - **Anisotropic case**: $N^{ref} = 491$ samples, $\#A^{ref} = 173$ for $U^{ref}$
Convergence analysis - Influence of relaxation parameter $\rho_k$

- Fixed cross-validation tolerance $\varepsilon_{cv} = 10^{-3}$
- Evolution of relative error indicator $\varepsilon_{\Xi,\Omega\setminus\Lambda}(U^k; U^{ref})$ with respect to iteration $\#k$

![Graph showing convergence](image)

**Figure**: Isotropic case

→ Strong influence of relaxation parameter $\rho_k$ on the convergence properties
Stationary non-linear diffusion-reaction stochastic problem

Convergence analysis - Influence of relaxation parameter $\rho_k$

- Fixed cross-validation tolerance $\varepsilon_{cv} = 10^{-3}$
- Evolution of relative error indicator $\varepsilon_{\Xi,\Omega \setminus \Lambda}(U^k; U^{ref})$ with respect to iteration $\# k$

![Graph showing convergence analysis](image)

- **Figure:** Anisotropic case

$\rightarrow$ **Strong influence of relaxation parameter $\rho_k$ on the convergence properties**
Stationary non-linear diffusion-reaction stochastic problem

Convergence analysis - Influence of cross-validation tolerance $\varepsilon_{cv}$

- Relaxation parameter $\rho_k$ dynamically updated using Aitken's Delta-Squared method
- Evolution of relative error indicator $\varepsilon_{\Xi,\Omega \setminus \Lambda}(U^k; U^{ref})$ with respect to iteration $\#k$

(a) Isotropic case
(b) Anisotropic case

$\varepsilon_{cv} = 10^{-2}$  $\varepsilon_{cv} = 10^{-3}$  $\varepsilon_{cv} = 10^{-4}$  $\varepsilon_{cv} = 10^{-5}$
Computational efficiency - Influence of cross-validation tolerance $\varepsilon_{cv}$

- Relaxation parameter $\rho_k$ dynamically updated using Aitken’s Delta-Squared method
- Evolution of CPU time per iteration with respect to iteration $\#k$

(a) Isotropic case

(b) Anisotropic case

$\varepsilon_{cv} = 10^{-2}$  $\varepsilon_{cv} = 10^{-3}$  $\varepsilon_{cv} = 10^{-4}$  $\varepsilon_{cv} = 10^{-5}$
Stationary non-linear diffusion-reaction stochastic problem

Adaptive sampling

- Initial sample size $N = 1$, Sampling factor $p_{\text{add}} = 0.1$
- Evolution of sample size $N_q$ for local solution $(w^k_q, \lambda^k_q)$ within patch $\Lambda_q$ with respect to iteration $#k$, for patch $#q = 4$

![Graphs showing the evolution of sample size for isotropic and anisotropic cases](image)

(a) Isotropic case

(b) Anisotropic case

- $\varepsilon_{cv} = 10^{-2}$
- $\varepsilon_{cv} = 10^{-3}$
- $\varepsilon_{cv} = 10^{-4}$
- $\varepsilon_{cv} = 10^{-5}$
Adaptive construction of sparse polynomial approximation spaces

- Initial dimension \( \#A_q = 0 \)
- Evolution of the dimension \( \#A_q \) of the stochastic approximation basis for local solution \( w^k_q \) (solid lines) and Lagrange multiplier \( \lambda^k_q \) (dashed lines) with respect to iteration \( \#k \), for patch \( \#q = 4 \)

(a) Isotropic case

(b) Anisotropic case

\[ \varepsilon_{cv} = 10^{-2}, \quad \varepsilon_{cv} = 10^{-3}, \quad \varepsilon_{cv} = 10^{-4}, \quad \varepsilon_{cv} = 10^{-5} \]
Adaptive construction of sparse polynomial approximation spaces

- Partial polynomial degrees $p_i$ (in each random variable $\xi_i$) and dimension $\#A$ of approximation spaces $S_A$ for global solution $U$ and local solutions $(w^q, \lambda^q)$

<table>
<thead>
<tr>
<th></th>
<th>$p_1$</th>
<th>$p_2$</th>
<th>$p_3$</th>
<th>$p_4$</th>
<th>$p_5$</th>
<th>$p_6$</th>
<th>$p_7$</th>
<th>$p_8$</th>
<th>$p_9$</th>
<th>$p_{10}$</th>
<th>$p_{11}$</th>
<th>$p_{12}$</th>
<th>$p_{13}$</th>
<th>$p_{14}$</th>
<th>$p_{15}$</th>
<th>$p_{16}$</th>
<th>$#A$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$U$</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>192</td>
</tr>
<tr>
<td>$w_1$</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>$w_2$</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>$\lambda_2$</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>$w_3$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>$\lambda_3$</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>21</td>
</tr>
<tr>
<td>$w_4$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$\lambda_4$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>$w_5$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$\lambda_5$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>$w_6$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>$\lambda_6$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>28</td>
</tr>
<tr>
<td>$w_7$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>$\lambda_7$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>$w_8$</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>$\lambda_8$</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>26</td>
</tr>
</tbody>
</table>

(a) Isotropic case
Stationary non-linear diffusion-reaction stochastic problem

Adaptive construction of sparse polynomial approximation spaces

- Partial polynomial degrees $p_i$ (in each random variable $\xi_i$) and dimension $\#A$ of approximation spaces $S_A$ for global solution $U$ and local solutions $(w^q, \lambda^q)$

<table>
<thead>
<tr>
<th></th>
<th>$p_1$</th>
<th>$p_2$</th>
<th>$p_3$</th>
<th>$p_4$</th>
<th>$p_5$</th>
<th>$p_6$</th>
<th>$p_7$</th>
<th>$p_8$</th>
<th>$p_9$</th>
<th>$p_{10}$</th>
<th>$p_{11}$</th>
<th>$p_{12}$</th>
<th>$p_{13}$</th>
<th>$p_{14}$</th>
<th>$p_{15}$</th>
<th>$p_{16}$</th>
<th>$#A$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$U$</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>$w_1$</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td>$w_2$</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>$\lambda_2$</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>$w_3$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>$\lambda_3$</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td>$w_4$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>$\lambda_4$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>$w_5$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>$\lambda_5$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>$w_6$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>$\lambda_6$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>$w_7$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>$\lambda_7$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>$w_8$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>$\lambda_8$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>7</td>
</tr>
</tbody>
</table>

(a) Anisotropic case
Aitken’s dynamic relaxation

- Evolution of relaxation parameter $\rho_k$ with respect to iteration $#k$

(a) Isotropic case
(b) Anisotropic case

<table>
<thead>
<tr>
<th>$\epsilon_{cv}$</th>
<th>$\epsilon_{cv} = 10^{-2}$</th>
<th>$\epsilon_{cv} = 10^{-3}$</th>
<th>$\epsilon_{cv} = 10^{-4}$</th>
<th>$\epsilon_{cv} = 10^{-5}$</th>
</tr>
</thead>
</table>
Stationary non-linear diffusion-reaction stochastic problem

Statistical moments of global, local and multiscale solutions at convergence

- Mean $\mathbb{E}$ and variance $\mathbb{V}$ of global solution $U$, local solutions $w_q$ and multiscale solution $u$

![Images showing statistical moments](image)

(a) $\mathbb{E}(U)$ and $\mathbb{E}(w_q)$
(b) $\mathbb{E}(u)$
(c) $\mathbb{V}(U)$ and $\mathbb{V}(w_q)$
(d) $\mathbb{V}(u)$

**Figure:** Isotropic problem

→ The highest spatial contributions to the variance $\mathbb{V}(u)$ are captured by the local solution $w_q$ within every patch $\Lambda_q$
Statistical moments of global, local and multiscale solutions at convergence

- **Mean** $\mathbb{E}$ and **variance** $\mathbb{V}$ of global solution $U$, local solutions $w_q$ and multiscale solution $u$

**Figure:** Anisotropic problem

$\rightarrow$ The highest spatial contributions to the **variance** $\mathbb{V}(u)$ are **localized** in the patches exhibiting the **highest input variabilities**
Stationary non-linear diffusion-reaction stochastic problem

Global sensitivity indices of multiscale solution at convergence

- Sensitivity indices \( \tilde{S}_i(u) = \frac{\nabla(E(u|\xi_i))}{\max_{x \in \Omega} \nabla(u)} \) quantifying the impact of each random variable \( \xi_i \) on the variance \( \nabla(u) \) of multiscale solution \( u \)

\[ \tilde{S}_1(u) \quad \tilde{S}_3(u) \quad \tilde{S}_5(u) \quad \tilde{S}_7(u) \quad \tilde{S}_9(u) \quad \tilde{S}_{11}(u) \quad \tilde{S}_{13}(u) \quad \tilde{S}_{15}(u) \]

\[ \tilde{S}_2(u) \quad \tilde{S}_4(u) \quad \tilde{S}_6(u) \quad \tilde{S}_8(u) \quad \tilde{S}_{10}(u) \quad \tilde{S}_{12}(u) \quad \tilde{S}_{14}(u) \quad \tilde{S}_{16}(u) \]

(a) Isotropic case
Global sensitivity indices of multiscale solution at convergence

- Sensitivity indices \( \tilde{S}_i(u) = \frac{\nabla (\mathbb{E}(u|\xi_i))}{\max_{x \in \Omega} \nabla (u)} \) quantifying the impact of each random variable \( \xi_i \) on the variance \( \nabla (u) \) of multiscale solution \( u \)

![Graphs](a) Anisotropic case
Transient linear advection-diffusion-reaction deterministic problem

\[ c \dot{u} - \nabla \cdot (K \nabla u) + \alpha \cdot \nabla u + Ru = 0 \quad \text{on } \Omega \times [0, T] \]

\[ u = 1 \quad \text{on } \Gamma_D^1, \quad u = 0 \quad \text{on } \Gamma_D^2, \quad u = 0 \quad \text{elsewhere on } \partial \Omega \]

\[ u_0 = 1 \quad \text{on } \Gamma_D^1, \quad u_0 = 0 \quad \text{elsewhere on } \Omega \]

- Space domain \( \Omega = \Omega_1 \cup \Omega_2 \cup \Omega_3 \)
- Time interval \( I = [0, T] \) with \( T = 2 \) s
- Thermal capacity \( c = 1 \)
- Diffusion coefficient \( K = 0.01 \)
- Reaction parameter \( R = \begin{cases} 0.1 & \text{on } \Omega_1 \\ 10 & \text{on } \Omega_2 \cup \Omega_3 \end{cases} \)
- Piecewise linear advection field \( \alpha = \nabla \psi \)
### Transient linear advection-diffusion-reaction deterministic problem

**Linear advection-diffusion-reaction problem** - Transport of pollutant inside an active carbon filter [Nouy 2010; Parés, Díez, and Huerta 2008]

\[
c\dot{u} - \nabla \cdot (K \nabla u) + \alpha \cdot \nabla u + Ru = 0 \quad \text{on } \Omega \times [0, T]
\]

- \( u = 1 \) on \( \Gamma_D^1 \), \( u = 0 \) on \( \Gamma_D^2 \), \( u = 0 \) elsewhere on \( \partial\Omega \)
- \( u_0 = 1 \) on \( \Gamma_D^1 \), \( u_0 = 0 \) elsewhere on \( \Omega \)

- **Space domain** \( \Omega = \Omega_1 \cup \Omega_2 \cup \Omega_3 \)
- **Time interval** \( I = [0, T] \) with \( T = 2 \) s
- **Thermal capacity** \( c = 1 \)
- **Diffusion coefficient** \( K = 0.01 \)
- **Reaction parameter** \( R = \begin{cases} 0.1 & \text{on } \Omega_1 \\ 10 & \text{on } \Omega_2 \cup \Omega_3 \end{cases} \)
- **Piecewise linear advection field** \( \alpha = \nabla \psi \)

**Potential flow** \( \psi \)
Linear advection-diffusion-reaction problem - Transport of pollutant inside an active carbon filter [Nouy 2010; Parés, Díez, and Huerta 2008]

\[
c \dot{u} - \nabla \cdot (K \nabla u) + \alpha \cdot \nabla u + Ru = 0 \quad \text{on } \Omega \times [0, T]
\]

\[
u = 1 \text{ on } \Gamma_D^1, \quad u = 0 \text{ on } \Gamma_D^2, \quad u = 0 \text{ elsewhere on } \partial \Omega
\]

\[
u_0 = 1 \text{ on } \Gamma_D^1, \quad u_0 = 0 \text{ elsewhere on } \Omega
\]

- Space domain \( \Omega = \Omega_1 \cup \Omega_2 \cup \Omega_3 \)
- Time interval \( I = [0, T] \) with \( T = 2 \) s
- Thermal capacity \( c = 1 \)
- Diffusion coefficient \( K = 0.01 \)
- Reaction parameter \( R = \begin{cases} 0.1 & \text{on } \Omega_1 \\ 10 & \text{on } \Omega_2 \cup \Omega_3 \end{cases} \)
- Piecewise linear advection field \( \alpha = \nabla \psi \)
Transient linear advection-diffusion-reaction deterministic problem

Linear advection-diffusion-reaction problem - Transport of pollutant inside an active carbon filter [Nouy 2010; Parés, Díez, and Huerta 2008]

\[ c\dot{u} - \nabla \cdot (K \nabla u) + \alpha \cdot \nabla u + Ru = 0 \quad \text{on } \Omega \times [0, T] \]
\[ u = 1 \text{ on } \Gamma_D^1, \quad u = 0 \text{ on } \Gamma_D^2, \quad u = 0 \text{ elsewhere on } \partial \Omega \]
\[ u_0 = 1 \text{ on } \Gamma_D^1, \quad u_0 = 0 \text{ elsewhere on } \Omega \]

- Space domain \( \Omega = \Omega_1 \cup \Omega_2 \cup \Omega_3 \)
- Time interval \( I = [0, T] \) with \( T = 2 \) s
- Thermal capacity \( c = 1 \)
- Diffusion coefficient \( K = 0.01 \)
- Reaction parameter \( R = \begin{cases} 0.1 & \text{on } \Omega_1 \\ 10 & \text{on } \Omega_2 \cup \Omega_3 \end{cases} \)
- Piecewise linear advection field \( \alpha = \nabla \psi \)

Approximation spaces

- Spatial level: coarse global and fine local finite element approximation spaces
  \[ \dim(\tilde{U}_H) = 3416 \text{ on } \tilde{\Omega} \quad \dim(W^q_h) = 441 \text{ on } \Lambda_q \]
- Time level: implicit Euler time scheme with \( N_t = 100 \) time steps
Convergence analysis - Influence of relaxation parameter $\rho_k$

- Evolution of relative error indicator $\varepsilon_{\Omega \setminus \Lambda, I}(U^k; U_{ref})$ with respect to iteration $\# k$

$\leftarrow$ **Strong influence** of relaxation parameter $\rho_k$ on the convergence properties

$\leftarrow$ **Aitken’s acceleration technique** provides better results as those obtained with an optimal fixed relaxation parameter
Global, local and multiscale solutions at convergence

- Time evolutions of \textit{global} solution $U$, \textit{local} solutions $w_q$ and \textit{multiscale} solution $u$

\textbf{Local solutions} $w_q$

\textbf{Global solution} $U$
Global, local and multiscale solutions at convergence

- Time evolutions of **global** solution $U$, **local** solutions $w_q$ and **multiscale** solution $u$
Global, local and multiscale solutions at convergence

- Time evolutions of **global** solution $\dot{U}$, **local** solutions $\dot{w}_q$ and **multiscale** solution $\dot{u}$

**Local solutions** $\dot{w}_q$

**Global solution** $\dot{U}$
Global, local and multiscale solutions at convergence

- Time evolutions of **global** solution $\dot{U}$, **local** solutions $\dot{w}_q$ and **multiscale** solution $\dot{u}$

**Multiscale solution** $\dot{u}$
Transient linear advection-diffusion-reaction deterministic problem

Quantity of interest: concentration of pollutant captured by the trap domain

- Evolution of the concentration of pollutant $\ell_2(u; t) = \int_{\Omega_2} u(x, y, t) \, d\Omega$ captured by the trap domain $\Omega_2$ with respect to time $t$

(a) Pollutant trap $\Omega_2$

(b) Concentration of pollutant $\ell_2(u; t)$

[Graph showing concentration change over time for multiscale and monoscale methods]
Quantity of interest: concentration of pollutant captured by the trap domain

- **Total concentration of pollutant** captured by the trap domain $\Omega_2$ along the complete time evolution $I = [0, T]$:

$$L_2(u) = \int_0^T \ell_2(u; t) \, dt = \int_0^T \int_{\Omega_2} u(x, y, t) \, d\Omega \, dt = 1.60876 \times 10^{-3}$$

$$L_2(u^{\text{ref}}) = \int_0^T \ell_2(u^{\text{ref}}; t) \, dt = \int_0^T \int_{\Omega_2} u^{\text{ref}}(x, y, t) \, d\Omega \, dt = 1.60876 \times 10^{-3}$$

with a relative error

$$\varepsilon_L = \frac{L_2(u) - L_2(u^{\text{ref}})}{L_2(u^{\text{ref}})} \approx 3 \times 10^{-15} \quad \text{(machine precision)}$$

$\leftarrow$ The canister is considered to work properly if $L_2(u)$ is small enough.
$\leftarrow$ If the outcome of pollutant exceeds a threshold value, the canister breaks and the design is not admissible.
Outline

1 Context and motivation
2 Multiscale domain decomposition method
3 Computational aspects
4 Numerical illustrations
5 Conclusions and outlooks
Conclusions and outlooks

Conclusions

- A multiscale method dedicated to localized uncertainties and possible localized non-linearities
  - Domain decomposition method with patches
  - Global-local iterative algorithm with proven convergence properties
- Analysis of the consistency, convergence and robustness properties of the algorithm under general assumptions
- Application to a broad class of problems
  - linear [Chevreuil, Nouy, and Safatly 2013] and semi-linear [Nouy and Pled 2017] elliptic equations (linear elasticity, stationary (non-)linear diffusion-reaction)
  - linear parabolic equations (transient linear advection-diffusion-reaction)

On-going works

- Extension to linear hyperbolic equations (linear elasticity structural dynamics)
- Validation on industrial applications with 3D complex geometry
- Extension to non-linear dynamic problems by incorporating non-linear effects
  - geometrical or material non-linearities
  - contact with friction (see [Hager et al. 2012] for a 3D dynamic tire problem with non-linear material and frictional contact)
- Improved sampling-based methods for solving time-dependent local problems
Thanks for your attention!


This research was supported by the French National Research Agency (ANR) (ICARE project, grant number ANR-12-MONU-0002) and by the French Association of Mechanics (AFM)


