



HAL
open science

A fast and robust sub-optimal control approach using reduced order model adaptation techniques

Mourad Oulghelou, Cyrille Allery

► **To cite this version:**

Mourad Oulghelou, Cyrille Allery. A fast and robust sub-optimal control approach using reduced order model adaptation techniques. *Applied Mathematics and Computation*, 2018, 333, pp.416-434. 10.1016/j.amc.2018.03.091 . hal-01939171

HAL Id: hal-01939171

<https://hal.science/hal-01939171>

Submitted on 31 Aug 2021

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.

A fast and robust sub-optimal control approach using reduced order model adaptation techniques



M. Oulghelou*, C. Allery

LaSIE, UMR 7356 CNRS, Université de La Rochelle Pôle Science et Technologie Avenue Michel Crépeau, 17042 La Rochelle Cedex 1, France

ARTICLE INFO

Keywords:

Optimal control

Reduced order models (ROMs)

Proper orthogonal decomposition (POD)

Proper generalized decomposition (PGD)

ROMs interpolation

Grassmann manifold

ABSTRACT

Classical adjoint-based optimization approach for the optimal control of partial differential equations is known to require a large amount of CPU time and memory storage. In this article, in order to reduce these requirements, a posteriori and a priori model order reduction techniques such as POD (Proper Orthogonal Decomposition) and PGD (Proper Generalized Decomposition) are used. As a matter of fact, these techniques allows a fast access to the temporal dynamics of a solution approximated in a suitable subspace of low dimension, spanned by a set of basis functions that form a reduced basis. The costly high fidelity model is then projected onto this basis and results in a system of ordinary differential equations which can be solved in quasi-real time. A disadvantage of considering a fixed POD basis in a suboptimal control loop, is basically the dependence of such bases on a posteriori information coming from high fidelity simulations. Therefore, a non robustness of the POD basis can be expected for certain perturbations in the original parameter for which it was built. As a result, update the reduced bases with respect to each variation in the control parameter using the POD method is still costly. To get over this difficulty, we equip the usual reduced optimal control algorithm with an intermediate basis adaptation step. The first proposed approach consists in adapting the reduced basis for a new control parameter by interpolating over a set of POD bases previously computed for a range of control parameters. To achieve that, an interpolation technique based on properties of the tangent subspace of the Grassmann manifold (ITSGM) is considered. The second approach is the PGD method, which by nature, enrich a space time decomposition trying to enhance the approximation by learning from its own errors. Relying on this property, this method is employed in the control loop as a basis corrector, in such a way the given spatial basis is adapted for the new control parameter by performing just few enrichments. These two approaches are applied in the sub-control of the two dimensional non-linear reaction-diffusion equations and Burgers equations.

© 2018 Elsevier Inc. All rights reserved.

1. Introduction

Optimal control of partial differential equations (PDE) using the adjoint-based approach requires to solve many times the state and their associated adjoint equations, which results in very large computational time and important memory storage. Therefore it is not possible to consider doing real-time or quasi-real time control by using conventional resolution tech-

niques based for example on finite element, finite volume or finite difference methods. In this scope, model order reduction techniques are attracting increasing interest due to their ability to represent a solution of any dynamical system in a low dimensional subspace, resulting by consequence in a fast access to the temporal dynamics of the problem. Reducing the order of a PDE requires first the construction of a representative reduced spatial basis. Then, by the Galerkin projection of the PDE onto this basis, a system of ordinary differential equations of low order, called through this paper reduced order model (ROM), is obtained. Finally, the approximation of the PDE solution over the time interval is achieved through the ROM equation which is very fast to solve. Among existing model reduction methods, the POD has received much attention over the past two decades and is still the most widely used. Since the POD method requires a set of snapshots, often obtained from solving the high fidelity model, the POD basis is not robust when the parameter rolls away the original parameter for which it was built. This is a blocking aspect, especially inside an optimal control loop where the control parameter vary.

Initial studies in the context of sub-optimal control using the POD method have been carried out, more particularly for fluid mechanics applications. For instance Bergmann et al. [1] controlled the time angular velocity of a rotating cylinder to optimize the drag of an incompressible viscous flow past a circular cylinder. Ravindran [2] considered the control problem of reducing recirculation behind the step of a flow in a backward facing step channel, by acting on the movement of a wall portion or through blowing on it. In these two studies the POD basis was chosen once and fixed all along the optimization process. Nevertheless, this hypothesis is strong and assumes that this chosen POD basis is able to approximate quite accurately the solutions associated parameters which will be tracked by the optimization algorithm. The fact that a POD basis is optimal only for the parameter for which it was built make this hypothesis suspect in general. An ensemble of snapshots sets obtained from a set of different parameters can be used to generate a basis with a wider region of trust. This approach was used for instance by Tallet et al. [3] in the control of the anisothermal Navier–Stokes equations in a differential heated lid driven cavity, by acting on the velocity and temperature on the walls. In the following this approach is named Multiple Parametrized Snapshots method (MPS).

In this paper, we propose an efficient and fast sub-optimal control algorithm with a basis adaptation step. Two approaches are considered for adaptation. The first approach consists in generating a set of POD bases for different parameters, and using a special method of interpolation based on properties of the tangent subspace of the Grassmann manifold (ITSGM) to obtain the basis associated to a given parameter. This approach was firstly introduced by Amsallem and Farhat [4] in the context of aeroelasticity. It involves notions of a geodesic path between two points, and of the tangent subspace at a point on the Grassmann manifold. The second approach consists in a correction process of the basis at each iteration of the control algorithm by using the so called PGD method. This method originally based on a space-time separation of the solution was proposed by Ladevéze et al [5–7] under the name of radial approximation to solve nonlinear structural mechanics. This space-time representation was also used in [8,9] to solve stochastic problems; in this context, the method is named Generalized Spectral Decomposition. Ammar et al. extended the PGD to multidimensional decomposition to solve models of polymeric systems [10,11]. Since these works, the PGD was applied to efficiently solve many problems of heat transfer [12], fluid mechanics [13–17], quantum chemistry [18], solid mechanics [19]... A review of the method can be found in [20]. In the context of optimal control and optimization, the PGD had already been investigated in previous works [21–24]. In these studies the PGD is used to build a general separated form of the solution for a large amount of parameters. This computational vade-mecum is afterwards used in order to perform real-time optimal control. In the present work, the considered approach is different and does not require any computations beforehand. More specifically, the PGD is used here to enrich time-space separated functions in order to enhance their accuracy. This process is embedded in the optimal control loop to operate whenever is needed like a basis corrector.

The paper is organized as follows : The 2nd section introduces the general optimal control problem and highlights the resolution difficulties faced with the classical adjoint based approach. In the 3rd section, the reduced order model approach using the Galerkin projection is presented. The 4th and 5th sections provide the details of calculations of POD and MPS-POD bases. The bases adaptation approaches ITSGM and PGD are exposed in Sections 6 and 7, respectively. The proposed reduced control approach with the basis adaptation step is afterwards outlined in Section 8. In Sections 9 and 10, the reduced control approach using ITSGM and PGD is numerically investigated in the examples of two dimensional reaction-diffusion and Burgers equations. Finally, some conclusions and future works are presented.

2. General optimal control problem formulation

Let K be a closed convex subset of a Banach space Z and V and W two Hilbert spaces endowed respectively with the inner products $\langle \cdot, \cdot \rangle_V$ and $\langle \cdot, \cdot \rangle_W$. The subject of investigation in the present paper is the non-linear variational problems of the form

$$\min \mathcal{J}(u, \omega) \text{ over } (u, \omega) \in V \times K \quad (1)$$

subject to

$$\mathcal{N}(u, \omega) = 0 \quad (2)$$

where $\mathcal{J} : V \times K \rightarrow \mathbb{R}$ denotes the cost functional, $u \in V$ the state variable and $\omega \in K$ the control parameter. The relationship between the state and control variables is described by the constraints map $\mathcal{N} : V \times Z \rightarrow W$. This relationship can typically be governed by a partial differential equation evolving in a time interval $]0, T]$, $T > 0$, and defined in a domain space $\Omega \subset \mathbb{R}^d$

($d = 1, 2, 3$). In the following paper, the spaces Z , V and W of interest are such as $Z = \mathbb{R}^p$, $V = L^2(0, T; H^1(\Omega))$ and $W = L^2(0, T; L^2(\Omega))$.

We assume that \mathcal{J} and \mathcal{N} are continuously Fréchet differentiable¹. Moreover, for the existence of a (locally) unique solution $u(\omega)$ to the state equation $\mathcal{N}(u, \omega) = 0$ in a neighbourhood of $(\bar{u}, \bar{\omega})$ we assume that $\partial_u \mathcal{N}(\bar{u}, \bar{\omega}) \in \mathcal{L}(V, W)$ is a bijection [25]. Let $\xi \in W$ denotes the adjoint state related to $u(\bar{\omega})$ and solution of the equation

$$\partial_u \mathcal{J}(u(\bar{\omega}), \bar{\omega}) + \partial_u \mathcal{N}(u(\bar{\omega}), \bar{\omega})^* \xi = 0 \quad (3)$$

where $\bar{\omega}$ is a local minimum of (1) and $\partial_u \mathcal{N}(u(\bar{\omega}), \bar{\omega})^*$ denotes the adjoint operator of $\partial_u \mathcal{N}(u(\bar{\omega}), \bar{\omega})$. The constrained optimization problem (1) is then converted to a simple optimization problem by using Lagrange multiplier theory. The Lagrange functional associated to the constraint (2) is given by

$$\begin{aligned} \mathcal{L} : V \times Z \times W &\longrightarrow \mathbb{R} \\ (u, \omega, \xi) &\longrightarrow \mathcal{L}(u, \omega, \xi) = \mathcal{J}(u, \omega) + \langle \xi, \mathcal{N}(u, \omega) \rangle_W \end{aligned}$$

By differentiating $\mathcal{L}(u, \omega, \xi)$ with respect to the state variable u , in the direction $\delta u \in V$, we obtain

$$\begin{aligned} \langle \partial_u \mathcal{L}(u, \omega, \xi), \delta u \rangle_V &= \langle \partial_u \mathcal{J}(u, \omega), \delta u \rangle_V + \langle \xi, \partial_u \mathcal{N}(u, \omega) \delta u \rangle_W \\ &= \langle \partial_u \mathcal{J}(u, \omega), \delta u \rangle_V + \langle \partial_u \mathcal{N}(u, \omega)^* \xi, \delta u \rangle_V \end{aligned}$$

Consequently, the adjoint (3) can be also obtained from

$$\partial_u \mathcal{L}(\bar{u}, \bar{\omega}, \xi) = 0.$$

The optimality system of the Lagrangian for a minimizing pair $(\bar{u}, \bar{\omega})$ is expressed as follows:

$$\mathcal{N}(\bar{u}, \bar{\omega}) = 0 \quad (4)$$

$$\partial_u \mathcal{L}(\bar{u}, \bar{\omega}, \xi) = 0. \quad (5)$$

$$(\omega - \bar{\omega})^T \partial_\omega \mathcal{L}(\bar{u}, \bar{\omega}, \xi) \geq 0 \quad \forall \omega \in K. \quad (6)$$

A possibility to numerically solve the problem (1) for the unknowns (u, ω) is to use the classical first order gradient descent algorithm. It consists in constructing a sequence (u^n, ω^n) that converges towards the solution $(\bar{u}, \bar{\omega})$ in $V \times K$ by moving the optimization parameter at each iteration in the negative direction of the functional gradient. Let P_K denote the orthogonal projection operator over the set K , it follows from (6) that $\bar{\omega}$ is characterized by the following formula:

$$\bar{\omega} = P_K(\bar{\omega} - \sigma \partial_\omega \mathcal{L}(\bar{u}, \bar{\omega}, \xi)) \quad \forall \sigma > 0. \quad (7)$$

In the case K has the form $K = \{\omega \in \mathbb{R}^p; \quad \omega_L \leq \omega \leq \omega_R\}$, the projection operator is given by

$$P_K(\omega) = \max(\omega_L, \min(\omega, \omega_R)) \quad (8)$$

The gradient descent algorithm is then defined by the recurrence relation

$$\omega^{n+1} = \max(\omega_L, \min(\omega^n - \sigma_n \partial_\omega \mathcal{L}(u^n, \omega^n, \xi^n), \omega_R)), \quad \sigma_n > 0 \quad (9)$$

In summary the different steps of the adjoint based constrained optimization are stated in [Algorithm 1](#).

Algorithm 1: Classical adjoint based constrained optimization.

- 1 initialization of $\omega^n = \omega^{init}$
 - 2 solve for u^n the state equation $\mathcal{N}(u^n, \omega^n) = 0$,
 - 3 solve for ξ^n the adjoint state equation $\partial_u \mathcal{N}(u^n, \omega^n)^* \xi^n = -\partial_u \mathcal{J}(u^n, \omega^n)$,
 - 4 evaluate the descent direction $d^n = -\partial_\omega \mathcal{L}(u^n, \omega^n, \xi^n)$,
 - 5 compute the new parameter $\omega^{n+1} = \max(\omega_L, \min(\omega^n + \sigma_n d^n, \omega_R))$, $\sigma_n > 0$,
 - 6 check convergence : if $|\mathcal{J}(u^{n+1}, \omega^{n+1}) - \mathcal{J}(u^n, \omega^n)| > \epsilon |\mathcal{J}(u^{n+1}, \omega^{n+1})|$, with $\epsilon > 0$ is an arbitrary real positive number chosen by the user, go to step 2.
-

The main issue with the classical adjoint based approach, is the huge requirements in CPU and memory storage needed to evaluate descent directions (step 4). In fact, the descent directions are calculated through the resolution of the two costly problems : the state and adjoint state problems (steps 2 and 3). Consequently, this approach can not be able to provide quasi-real time control solutions and alternative approaches must be used. In the next section, in order to overcome this issue, the alternative approach of reduced order modelling will be presented.

¹ In the following, we denote by $\partial_{x_i} f$ the partial derivative of a multivariate function f with respect to x_i .

3. Reduced order modeling

Let $\mathcal{T} = L^2(0, T)$, $\mathcal{V} = H^1(\Omega)$ and $\mathcal{W} = L^2(\Omega)$, such that $V = L^2(0, T; H^1(\Omega))$ and $W = L^2(0, T; L^2(\Omega))$ are respectively identified to the tensor Hilbert spaces $\mathcal{T} \otimes \mathcal{V}$ and $\mathcal{T} \otimes \mathcal{W}$. The aim of this section, is basically to reduce CPU and memory storage requirements of the classical adjoint based control approach. The approach of reduced order modelling will then be used for this purpose. It consists in determining a space time separated representation that lays in the tensor Hilbert product space $\mathcal{V} \otimes \mathcal{T}$ and approximates the solution $u(t, x, \omega)$ as follows:

$$u(x, t, \omega) \approx u_m(x, t, \omega) = \sum_{j=1}^m \alpha^j(t, \omega) \varphi^j(x, \omega) \quad (10)$$

where u_m is the orthogonal projection of u onto the finite dimensional subspace \tilde{V} spanned by the set of spatial basis functions $\Phi = \{\varphi^1, \dots, \varphi^m\} \in \mathcal{V}^m$. The equations describing the evolution of the unknown temporal coefficients $\alpha_m = \{\alpha^1, \dots, \alpha^m\} \in \mathcal{T}^m$ are obtained in two steps : first the state variable u is replaced by u_m in the state (2), then the Galerkin projection is performed resulting in the following system of ordinary differential equations

$$\left\langle \mathcal{N} \left(\sum_{j=1}^m \alpha^j \varphi^j, \omega \right), \varphi^i \right\rangle_{L^2(\Omega)} = 0, \quad \forall i = 1, \dots, m \quad (11)$$

Given that the low dimensional subspace \tilde{V} spanned by Φ is known, the determination of u_m depends only on temporal coefficients α_m . It follows that the relationship between the control parameter ω and the approximated state variable u_m can be described by the set of ordinary differential (11), called reduced order model. By introducing the reduced constraints map $\mathcal{N}_\Phi : \mathcal{T}^m \times Z \rightarrow \mathcal{T}^m$, the reduced constraints problem (11) can be reformulated in the following compact form:

$$\begin{cases} \text{Find } \alpha_m \in \mathcal{T}^m \text{ such that} \\ \mathcal{N}_\Phi(\alpha_m, \omega) = 0 \end{cases} \quad (12)$$

The construction of a low dimensional subspace spanned by the spatial basis Φ can be achieved by different methods. One of the most popular and robust methods used to build this low dimensional subspace is the proper orthogonal decomposition (POD). Once the POD basis is known, the temporal dynamics can be recovered by solving the state reduced problem (12). A brief overview of the POD method is given in the next section.

4. Proper orthogonal decomposition

Given its optimality, the Proper Orthogonal Decomposition (POD) is the most widely used method for the generation of spatial basis functions. Compared to other existing model reduction methods, the POD is optimal and allows to represent the solution u in the least number of modes. These modes, forming a basis by construction, are built from the information contained in a set of realizations called snapshots [26]. Let $\mathcal{X} = \{u^1, \dots, u^{N_s}\}$ be an ensemble of snapshots, solutions at different time instants (t_1, \dots, t_{N_s}) of the considered system of evolution (2). The steps to calculate the POD basis are reported in Algorithm 2.

Algorithm 2: POD algorithm.

1 build the correlation matrix \mathcal{C} whose elements are given as

$$\mathcal{C}_{ij} = \langle u^i, u^j \rangle_{L^2(\Omega)}, \quad 1 \leq i, j \leq N_s$$

2 compute eigenvalues $\gamma_1 > \dots > \gamma_{N_s}$ and eigenvectors y^1, \dots, y^{N_s} of \mathcal{C} ,

3 set $\tilde{\phi}_i = \sum_{j=0}^{N_s} y_j^i u^j$ for $1 \leq i \leq N_s$,

4 normalize $\phi_i = \frac{\tilde{\phi}_i}{\|\tilde{\phi}_i\|_{L^2(\Omega)}}$ for $1 \leq i \leq N_s$.

In order to obtain a low dimensional basis, the modes corresponding to small eigenvalues are neglected. Only the m most energetic first modes are considered ($m \ll N_s$). As already mentioned in the introduction, the major issue of the POD in the sub-optimal control approach lays in the fact that the constructed basis is valid only for the parameter for which it was built. This aspect is particularly an obstruction point within an optimal control loop, where the control parameters change.

5. Multiple parametrized snapshots method

A more convenient choice of a POD basis to be used in the sub-optimal control, can be made by using the Multiple Parametrized Snapshots method (MPS). Let $\omega \in \{\omega_1, \dots, \omega_{N_p}\}$ be a control parameter and $\mathcal{X}(\omega) = \{u^1(\omega), \dots, u^{N_s}(\omega)\}$ a set

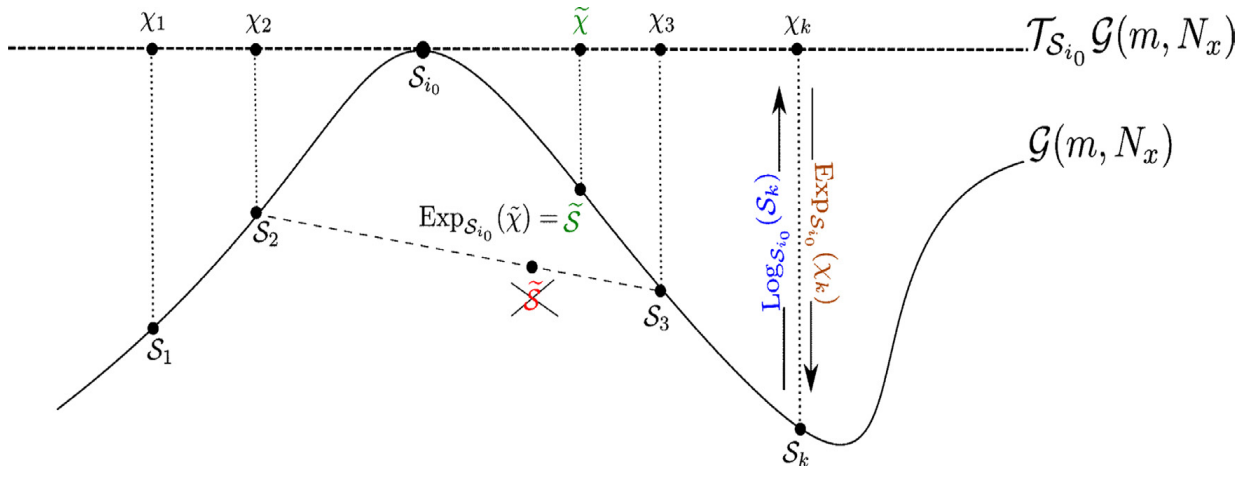


Fig. 1. Visual illustration of the link between the Grassmann manifold $\mathcal{G}(m, N_x)$ and its tangent space $\mathcal{T}_{S_{i_0}} \mathcal{G}(m, N_x)$ at the reference point $S_{i_0} \in \mathcal{G}(m, N_x)$. The subspaces $S_k \in \mathcal{G}(m, N_x)$ are spanned by full rank column matrices and can typically be represented by orthonormal POD bases. The geodesics between the reference point S_{i_0} and the points S_k are characterized by initial velocities $\chi_k \in \mathcal{T}_{S_{i_0}} \mathcal{G}(m, N_x)$ which can be obtained using the logarithmic map. The point $\tilde{\chi}$ represents the interpolation on the tangent space of the set of initial velocities χ_1, χ_2, \dots ; and $\tilde{S} \in \mathcal{G}(m, N_x)$ is its image by the exponential map. It's worth noting that a linear interpolation of the two points S_2 and S_3 may result in $\tilde{S} \notin \mathcal{G}(m, N_x)$.

of parametrized snapshots, solutions at different time instants (t_1, \dots, t_{N_s}) , of the system of evolution (2). The MPS method consists in using Algorithm 2 in order to generate a global POD basis, based on the set of snapshots $\bigcup_{k=1}^{N_p} \mathcal{X}(\omega_k)$. In fact, using such sets of multiple parametrized snapshots in the POD algorithm, enables to produce POD bases which are more rich in terms of information compared to the single parameter POD bases. More specifically, the MPS-POD basis will be optimal for the set of parameters $\omega_1, \dots, \omega_{N_p}$, and can predict quite well the dynamics of other neighbour parameters.

6. POD Bases adaptation using ITSGM

In order to overcome the main disadvantage of the POD approach within control loops, it is possible to construct a bunch of POD bases corresponding to different control parameters and then to interpolate them to obtain a good representing basis for the parameter imposed by the control algorithm. In this context, standard interpolation methods such as Lagrange interpolation or Radial Basis Function (RBF) interpolation of a set of POD bases do not make sense since they do not necessarily produce a basis (see Fig. 1 for illustration).

On the other hand the ITSGM method (Interpolation On a Tangent Subspace of the Grassmann Manifold) based on calculation of geodesics in the Grassmann manifold, is a suitable interpolation method which by nature ensures this property. The aim of the following, is to give a brief description of the mathematical framework from differential geometry that enables to define this interpolation approach on POD bases. For details and proofs we refer the interested reader to [4,27,28].

6.1. Grassmann manifold

Let $\Phi \in \mathbb{R}^{N_x \times m}$ denote the full-rank column matrix, whose columns provide a basis in \mathbb{R}^{N_x} of the subspace S of dimension² m . The set of all these m dimensional subspaces S form what we call the Grassmann manifold $\mathcal{G}(m, N_x)$. The matrices $\Phi \in \mathbb{R}^{N_x \times m}$ spanning these subspaces belong to the so called non-compact Stiefel manifold $ST(m, N_x)$. For the sake of simplicity, we consider the compact Stiefel manifold which is a subset of $ST(m, N_x)$ that contains only orthonormal matrices. To be able to define a suitable interpolation based on notions of differential geometry, we need to define the tangent space at a point of the Grassmann manifold and the concept of geodesic path between two points on a differential manifold.

6.2. Tangent subspace at a point of the Grassmann manifold

An important result about the Grassmann manifold $\mathcal{G}(m, N_x)$ states that at each of its points S there exists a tangent space $\mathcal{T}_S \mathcal{G}(m, N_x)$ of the same dimension with origin the point of tangency. Each point of this tangent space $\mathcal{T}_S \mathcal{G}(m, N_x)$ can be as well represented by a matrix $\Gamma \in \mathbb{R}^{N_x \times m}$.

6.3. Geodesic path

In a differential manifold such as the Grassmann manifold, the shortest path between two points S_{i_0} and S is called a geodesic [28]. This path can uniquely be represented with a twice differentiable function $\mathcal{Y}(t)$ (with $0 \leq t \leq 1$) solution of a

² The columns of these matrices might be the modes of a POD basis, in this case N_x denotes the number of degrees of freedom of the full order model, and m the size of the reduced order basis.

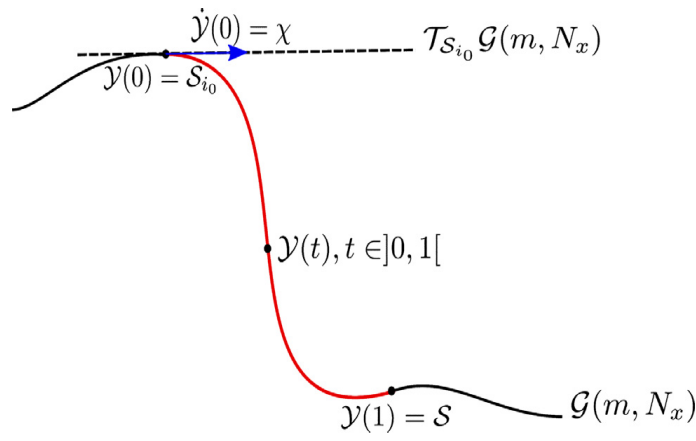


Fig. 2. Visual illustration of the geodesic path $\mathcal{Y}(t)$ between the two points $\mathcal{S}_{i_0}, \mathcal{S} \in \mathcal{G}(m, N_x)$.

second order ordinary differential equation, for which $\mathcal{Y}(0) = \mathcal{S}_{i_0}$ and $\dot{\mathcal{Y}}(0) = \chi$ represent respectively the initial and initial derivative conditions, while the ending point at $t = 1$ of this parametrization is $\mathcal{Y}(1) = \mathcal{S}$ (see Fig. 2). The fact that the initial derivative χ belongs to the tangent space at \mathcal{S}_{i_0} , allows to establish a link between the concept of a geodesic and that of the exponential mapping which maps a tangent space to the manifold itself [27]. The exponential of χ , denoted by $\text{Exp}_{\mathcal{S}_{i_0}}(\chi)$ is given by

$$\text{Exp}_{\mathcal{S}_{i_0}}(\chi) = \mathcal{S} \quad (13)$$

In the case of the Grassmann manifold, an explicit formula that enables a parametrization of the geodesic path is given by the following mathematical result [27]:

Proposition 1. Let $\mathcal{S}_{i_0} \in \mathcal{G}(m, N_x)$ be a point on the Grassmann manifold and $\mathcal{T}_{\mathcal{S}_{i_0}} \mathcal{G}(m, N_x)$ the associated tangent subspace. If Φ_{i_0} is an orthogonal column matrix representing \mathcal{S}_{i_0} and $\Gamma = U \Sigma V^T$ is the thin SVD of the matrix representing $\chi \in \mathcal{T}_{\mathcal{S}_{i_0}} \mathcal{G}(m, N_x)$. Then the parametric representation of the geodesic path on $\mathcal{G}(m, N_x)$ with initial conditions $\mathcal{Y}(0) = \mathcal{S}_{i_0}$ and $\dot{\mathcal{Y}}(0) = \chi$ is given by

$$\mathcal{Y}(t) = \text{span} \left\{ \Phi_{i_0} V \cos(t \Sigma) + U \sin(t \Sigma) \right\} \quad 0 \leq t \leq 1 \quad (14)$$

6.4. Link between the Grassmann manifold and its tangent subspace

The link between the tangent space $\mathcal{T}_{\mathcal{S}_{i_0}} \mathcal{G}(m, N_x)$ at a point \mathcal{S}_{i_0} of the Grassmann manifold and the manifold itself is established by the exponential mapping, which can be explicitly determined using (13) and (14). The inverse mapping between a neighbourhood of \mathcal{S} and $\mathcal{T}_{\mathcal{S}} \mathcal{G}(m, N_x)$ is defined by the logarithmic map. This mapping is denoted by $\text{Log}_{\mathcal{S}}$ and defined in a neighbourhood of $\mathcal{S} \in \mathcal{G}(m, N_x)$ [28]. The following result enables the practical computation of the logarithmic map:

Proposition 2. Let $\mathcal{S}_{i_0} \in \mathcal{G}(m, N_x)$ be a subspace spanned by the full rank column matrix Φ_{i_0} , and $\mathcal{S} \in \mathcal{G}(m, N_x)$ a neighbour subspace spanned by the full rank column matrix Φ . The image of \mathcal{S} by the logarithmic map $\text{Log}_{\mathcal{S}_{i_0}}$, $\chi = \text{Log}_{\mathcal{S}_{i_0}}(\mathcal{S}) \in \mathcal{T}_{\mathcal{S}_{i_0}} \mathcal{G}(m, N_x)$, is represented by the matrix Γ given by

$$\Gamma = U \arctan(\Sigma) V^T$$

where U, V and Σ are the matrices of the thin singular value decomposition

$$(I - \Phi_{i_0} \Phi_{i_0}^T) \Phi (\Phi_{i_0}^T \Phi)^{-1} = U \Sigma V^T$$

6.5. Interpolation of POD Bases by the ITSGM method

Let $\omega_1, \dots, \omega_{N_p}$ be a set of training parameters and $\mathcal{X}_1, \dots, \mathcal{X}_{N_p}$ the corresponding set of snapshots matrices obtained by solving a parametrized partial differential equation (by using for instance a finite element solver). For each parameter ω_i , a POD basis Ψ_i can be calculated as $\Psi_i = \mathcal{X}_i v^i$ such that v^i solves the eigenvalue problem $\mathcal{X}_i^T \mathcal{M} \mathcal{X}_i v^i = v^i \lambda^i$ (\mathcal{M} is the mass matrix). In order to exploit the explicit formulas of the exponential and logarithmic maps on POD bases, we consider the set of modified bases $\Phi_1, \dots, \Phi_{N_p}$ obtained by applying the Gram-Schmidt orthonormalization process to each matrix $\Psi_1, \dots, \Psi_{N_p}$, with respect to the euclidean norm (norm-2). Obviously, Φ_i belongs to the Stiefel manifold $\mathcal{ST}(m, N_x)$ and can be interpreted as a representing basis of the subspace $\mathcal{S}_i \in \mathcal{G}(m, N_x)$. Let i_0 be an arbitrary fixed reference index, and $\mathcal{Y}_i(t)$ the geodesic linking the reference point \mathcal{S}_{i_0} to the point \mathcal{S}_i , $i \in \{1, \dots, N_p\}$. These geodesics are characterized by initial velocities $\chi_1, \dots, \chi_{N_p}$ for which the representative matrices $\Gamma_1, \dots, \Gamma_{N_p}$ can be calculated via the logarithmic map ($\text{Log}_{\mathcal{S}_{i_0}}(\mathcal{S}_{i_0}) = 0$). The aim now is to define a good strategy allowing to approximate the new basis $\tilde{\Phi}$ for a new untrained parameter $\tilde{\omega} \notin \{\omega_1, \dots, \omega_{N_p}\}$. In order to derive the basis interpolation method (ITSGM), the following observation is ex-

exploited : all the geodesics $\mathcal{Y}_i(t)$ sharing the same initial condition $\mathcal{Y}_i(0) = \mathcal{S}_{i_0}$, can be distinguished from each others by only their corresponding initial velocities $\dot{\mathcal{Y}}_i(0) = \chi_i$. Given that these initial velocities belong to the tangent vector subspace $\mathcal{T}_{\mathcal{S}_{i_0}} \mathcal{G}(m, N_x)$ [27], standard interpolations such as RBF, Lagrange, Spline ... etc can be performed as usual. More precisely, the RBF method [29] with the Gaussian radial function is chosen to be used for the numerical applications in this paper. Once the representative matrix $\tilde{\Gamma}$ of the initial velocity $\tilde{\chi}$ is calculated by interpolation for the new untrained parameter $\tilde{\omega}$, the approximated basis $\tilde{\Phi}$ spanning the subspace $\tilde{\mathcal{S}}$ can be calculated via the exponential map $\tilde{\mathcal{S}} = \text{Exp}_{\mathcal{S}_{i_0}}(\tilde{\chi})$ as follows

$$\tilde{\Phi} = \Phi_{i_0} \tilde{V} \cos(\tilde{\Sigma}) + \tilde{U} \sin(\tilde{\Sigma}) \quad \text{where} \quad \tilde{\Gamma} = \tilde{U} \tilde{\Sigma} \tilde{V}^T$$

To summarize, the practical interpolation algorithm for POD bases given in [4], is described in [Algorithm 3](#).

Algorithm 3: ITSGM algorithm.

1 choose a reference point $\mathcal{S}_{i_0} = \text{span}(\Phi_{i_0})$ to be the origin point of interpolation,

2 using the logarithmic map, map each $\mathcal{S}_i = \text{span}(\Phi_i)$ to a matrix Γ_i representing the initial velocity $\chi_i \in \mathcal{T}_{\mathcal{S}_{i_0}} \mathcal{G}(m, N_x)$

$$\Gamma_i = U_i \arctan(\Sigma_i) V_i^T \quad \text{with} \quad (I - \Phi_{i_0} \Phi_{i_0}^T) \Phi_i (\Phi_{i_0}^T \Phi_i)^{-1} = U_i \Sigma_i V_i^T \quad (15)$$

3 interpolate the initial velocities to obtain $\tilde{\Gamma}$ for the new untrained parameter $\tilde{\omega}$,

4 using the exponential map $\text{Exp}_{\mathcal{S}_{i_0}}$, map the matrix $\tilde{\Gamma}$ to the subspace $\tilde{\mathcal{S}}$, and find the spanning matrix $\tilde{\Phi}$ as follows:

$$\tilde{\Phi} = \Phi_{i_0} \tilde{V} \cos(\tilde{\Sigma}) + \tilde{U} \sin(\tilde{\Sigma}) \quad \text{where} \quad \tilde{\Gamma} = \tilde{U} \tilde{\Sigma} \tilde{V}^T \quad (16)$$

7. Proper generalized decomposition

Now, we will present the second method of basis adaptation called the PGD. Suppose we have a time-space decomposition u_m such that

$$u(t, x) \approx u_m(t, x) = \sum_{k=1}^m \alpha^k(t) \varphi^k(x) \quad (17)$$

The PGD method aims to define $u_{m+1} = u_m + \tilde{\alpha} \tilde{\varphi}$ such that the approximation (17) is enhanced. The definition of a new optimal couple $(\tilde{\alpha}, \tilde{\varphi}) \in \mathcal{T} \times \mathcal{V}$ lays in satisfying the following double Galerkin orthogonality criterion:

$$\langle \mathcal{N}(u_m + \tilde{\alpha} \tilde{\varphi}), \tilde{\alpha} \varphi^* + \alpha^* \tilde{\varphi} \rangle_W = 0, \quad \forall (\alpha^*, \varphi^*) \in \mathcal{T} \times \mathcal{V} \quad (18)$$

Let suppose now that the temporal function $\tilde{\alpha} \in \mathcal{T}$ is known, and define the map $f_x^{(m)} : \mathcal{T} \rightarrow \mathcal{V}$ which to $\tilde{\alpha}$ associates a spatial function $\tilde{\varphi}$ as follows:

$$\langle \mathcal{N}(u_m + \tilde{\alpha} \tilde{\varphi}), \tilde{\alpha} \varphi^* \rangle_W = 0, \quad \forall \varphi^* \in \mathcal{V} \quad (19)$$

Similarly, for a known spatial function $\tilde{\varphi} \in \mathcal{V}$, we define the map $f_t^{(m)} : \mathcal{V} \rightarrow \mathcal{T}$ which to $\tilde{\varphi}$ associates a temporal function $\tilde{\alpha}$ as follows:

$$\langle \mathcal{N}(u_m + \tilde{\alpha} \tilde{\varphi}), \alpha^* \tilde{\varphi} \rangle_W = 0, \quad \forall \alpha^* \in \mathcal{T} \quad (20)$$

A natural definition of the new couple $(\tilde{\alpha}, \tilde{\varphi}) \in \mathcal{T} \times \mathcal{V}$ can be formulated as the solution of the following fixed point problem:

$$f_x^{(m)} \circ f_t^{(m)}(\tilde{\alpha}) \stackrel{\text{def}}{=} g_x^{(m)}(\tilde{\alpha}) = \tilde{\alpha} \quad (21)$$

The one step PGD algorithm is given in [Algorithm 4](#).

Algorithm 4: PGD enrichment step.

1 $k = 0$ and $\tilde{\alpha}^k$ given arbitrary;

2 **while** $|\sigma_m^k - \sigma_m^{k-1}| > \epsilon_u |\sigma_m^k|$ **or** $k \leq k_{max}$ **do**

3 solve $\tilde{\varphi}^{k+1} = f_x^{(m)}(\tilde{\alpha}^k)$;

4 solve $\tilde{\alpha}^{k+1} = f_t^{(m)}(\tilde{\varphi}^{k+1})$;

5 set $k \leftarrow k + 1$ and $\sigma_m^k = \|\tilde{\alpha}^k\|$;

6 **end**

At the end of the algorithm, a correlation of the new temporal function $\tilde{\alpha}$ with the temporal functions $\{\alpha^j\}_{j=0,\dots,m}$ can be achieved by solving the reduced order model (12). After several iterations of the Algorithm 4, the size of the separated form (17) increases and may contain correlated basis functions. If necessary, in order to reduce the size of the separated form, one can perform a POD on the approximated solution in order to keep only q modes associated to the q first higher eigenvalues satisfying $\lambda_q > \eta\lambda_{\max}$, where $\eta > 0$ is set by the user.

8. Reduced optimal control problem

We recall that $\mathcal{T} = L^2(0, T)$ and $K \subset Z$ is a closed convex subset of $Z = \mathbb{R}^p$. The aim of this section is to take advantage of the earlier discussed ROM adaptation methods, in order to formulate a robust suboptimal control methodology. Let us assume that the state variable $u = u(\omega)$ is approximated in the following space time separated form:

$$u(t, x, \omega) \approx u_m(t, x, \omega) = \sum_{k=1}^m \alpha^k(t, \omega) \varphi^k(x, \omega) \quad (22)$$

where the set of temporal coefficients $\alpha_m = \{\alpha^1, \dots, \alpha^m\}$ can be determined from solving the reduced order model (12) obtained through the Galerkin projection of the original equation onto the spatial basis $\Phi = \{\varphi^1, \dots, \varphi^m\}$. For $\omega \in K$ fixed, we define the map

$$\begin{aligned} \mathcal{F}_{\Phi_\omega} : K &\longrightarrow \mathcal{T}^m \\ \tilde{\omega} &\longrightarrow \tilde{\alpha}_m \end{aligned}$$

such that $\tilde{\alpha}_m$ is the solution to $\mathcal{N}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}) = 0$. The special case $\tilde{\omega} = \omega$ coincides with the reduced order model defined by (12). In other words, for any neighbour parameter $\tilde{\omega}$ in the neighbourhood of $\omega \in K$, the map $\mathcal{F}_{\Phi_\omega}$, defines the solution of the ROM obtained from the Galerkin projection of the original equations onto Φ_ω and solved for the parameter $\tilde{\omega}$. Let now Φ_ω be a reduced basis associated to a parameter $\omega \in K$ to be fixed. By substitution of the expression (22) in (1) and (2), we derive the following reduced constrained optimization problem:

$$\min \mathcal{J}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}) \quad \text{over} \quad (\tilde{\alpha}_m, \tilde{\omega}) \in \mathcal{T}^m \times K \quad (23)$$

such that

$$\tilde{\alpha}_m = \mathcal{F}_{\Phi_\omega}(\tilde{\omega}) \quad \text{i.e.} \quad \mathcal{N}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}) = 0 \quad (24)$$

We note that the constraint (24) is a low order system of ordinary differential equations which is very fast to solve compared to the full constraints (2). In the same fashion as in Section 2, we define the reduced Lagrange functional associated to the reduced constraint (24) by

$$\begin{aligned} \mathcal{L}_{\Phi_\omega} : \mathcal{T}^m \times Z \times \mathcal{T}^m &\longrightarrow \mathbb{R} \\ (\tilde{\alpha}_m, \tilde{\omega}, \xi_m) &\longrightarrow \mathcal{L}(\tilde{\alpha}_m, \tilde{\omega}, \xi_m) = \mathcal{J}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}) + \langle \xi_m, \mathcal{N}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}) \rangle_{\mathcal{T}^m} \end{aligned}$$

For a minimizing pair $(\tilde{\alpha}_m, \tilde{\omega})$, by vanishing the differential of $\mathcal{L}_{\Phi_\omega}$ with respect to the reduced state variable $\tilde{\alpha}_m$

$$\partial_{\tilde{\alpha}_m} \mathcal{L}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}, \xi_m) = 0$$

we can find the expression of the reduced adjoint equation

$$\partial_{\tilde{\alpha}_m} \mathcal{N}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega})^* \xi_m + \partial_{\tilde{\alpha}_m} \mathcal{J}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}) = 0 \quad (25)$$

Moreover, by differentiating $\mathcal{L}_{\Phi_\omega}$ with respect to $\tilde{\omega}$ we have

$$\partial_{\tilde{\omega}} \mathcal{L}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}, \xi_m) = \partial_{\tilde{\omega}} \mathcal{J}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}) + \partial_{\tilde{\omega}} \mathcal{N}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega})^* \xi_m \quad (26)$$

and the following optimality condition is deduced

$$(\varrho - \tilde{\omega})^T \left(\partial_{\tilde{\omega}} \mathcal{J}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega}) + \partial_{\tilde{\omega}} \mathcal{N}_{\Phi_\omega}(\tilde{\alpha}_m, \tilde{\omega})^* \xi_m \right) \geq 0 \quad \forall \varrho \in K.$$

The usual suboptimal control methodology assumes the reduced basis to be fixed all along the optimization process. This assumption may have some drawbacks namely when a bad choice of the reduced basis is made. The proposed technique on the other hand can handle that choice, by defining the reduced control as a sequence of optimization sub-problems intermediated with a basis adaptation step. That is, for a candidate control parameter $\omega^{(n)}$, the reduced basis $\Phi_{\omega^{(n)}}$ is adapted using for instance ITSGM or PGD, and the reduced optimization problem (23) (24) is formulated and solved. This process is iterated until a convergence criteria is achieved. The suboptimal control optimization is resumed in Algorithm 5.

When the PGD method is used in the sub-optimal control algorithm, it is worth noting that one PGD enrichment at each time of the adaptation step is sufficient to allow good predictions of the new descent directions. When the stagnation of control parameters is achieved, the sub optimal control algorithm (Algorithm 4), will act exactly like the classical PGD algorithm [17].

Algorithm 5: Sub-optimal control algorithm.

```
1 set  $k = 0$ , and chose  $\omega^{(k)}$  arbitrary;
2 while  $|\mathcal{J}_{\Phi_{\omega^{(k)}}}(\alpha_m^{(k)}, \omega^{(k)}) - \mathcal{J}_{\Phi_{\omega^{(k)}}}(\alpha_m^{(k-1)}, \omega^{(k-1)})| \geq \epsilon |\mathcal{J}_{\Phi_{\omega^{(k)}}}(\alpha_m^{(k)}, \omega^{(k)})|$  do
3   set  $l = 0$ ,  $\tilde{\omega}^{(l)} = \omega^{(k)}$ ;
4   while  $|\mathcal{J}_{\Phi_{\omega^{(k)}}}(\tilde{\alpha}_m^{(l)}, \tilde{\omega}^{(l)}) - \mathcal{J}_{\Phi_{\omega^{(k)}}}(\tilde{\alpha}_m^{(l-1)}, \tilde{\omega}^{(l-1)})| \geq \epsilon_r |\mathcal{J}_{\Phi_{\omega^{(k)}}}(\tilde{\alpha}_m^{(l)}, \tilde{\omega}^{(l)})|$  do
5      $l \leftarrow l + 1$ ;
6     For  $\tilde{\omega} = \tilde{\omega}^{(l-1)}$ , solve the state ROM (26)  $\rightarrow \tilde{\alpha}_m^{(l)}$ ;
7     For  $\tilde{\alpha}_m = \tilde{\alpha}_m^{(l)}$  and  $\tilde{\omega} = \tilde{\omega}^{(l-1)}$ , solve the adjoint ROM (28)  $\rightarrow \tilde{\xi}_m^{(l)}$ ;
8     Compute the descent direction  $d^{(l)} = -\partial_{\tilde{\omega}} \mathcal{L}_{\Phi_{\omega}}(\tilde{\alpha}_m^{(l)}, \tilde{\omega}^{(l)}, \tilde{\xi}_m^{(l)})$  (29);
9     Compute the new control  $\tilde{\omega}^{(l)} = P_K(\tilde{\omega}^{(l-1)} + \sigma_l d^{(l)})$  such that  $\sigma_l > 0$  can be determined using the Armijo rule;
10  end
11   $k \leftarrow k + 1$ ;
12  set  $\omega^{(k)} \leftarrow \tilde{\omega}^{(l)}$  and  $\alpha_m^{(k)} \leftarrow \tilde{\alpha}_m^{(l)}$ ;
13  adapt  $\Phi^{(k)}$  to the new parameter  $\omega^{(k)}$  using ITSGM or PGD method;
14 end
```

Remark 1. Contrary to the a posteriori methods ITSGM and MPS, the PGD is an a priori method which can operate on a given time-space separated approximation u_m , by enriching it with a new couple $\{\alpha^{m+1}, \varphi^{m+1}\}$ such that $u_{m+1} = u_m + \alpha^{m+1}\varphi^{m+1}$ is a better approximation. This process can be repeated many times, until the maximal accuracy of the method is attained. Basically, the accuracy of solutions obtained using ITSGM or MPS is determined by the initial choice of the training parameters distribution, where the sets of snapshots used to build POD bases are calculated. In the sub-optimal control loop, the MPS and ITSGM methods will achieve there maximal accuracy after a certain number of iterations. Beyond these iterations, the cost functional as well as the control parameters will stagnate, meaning that a suboptimal control is achieved. A possible way to enhance the achieved sub-optimal control results can be by keeping up the optimization process using the PGD method. In the following, these combinations of methods are referred to as ITSGM/PGD and MPS/PGD.

9. Application to the control of non-linear reaction-diffusion equation

In this section, the control of 2D non-linear reaction-diffusion equation is numerically investigated in order to test the pertinence of the ROMs adaptation methods ITSGM and PGD in the proposed sub-optimal control strategy. Let $\Omega \subset \mathbb{R}^2$ a bounded domain of boundaries $\Gamma = \Gamma_D \cup \Gamma_N$ and $T > 0$. The non-stationary reaction-diffusion equation with initial condition $u_0 \in L^2(\Omega)$ submitted to a body force $f_\omega \in L^2([0, T], L^2(\Omega))$ is defined by

$$\begin{cases} \frac{\partial u}{\partial t} - \nu \Delta u + \frac{1}{2}u^2 = f(\omega) & \text{in } \Omega \times]0, T] \\ u = g & \text{on } \Gamma_D \times]0, T] \\ \frac{\partial u}{\partial n} = 0 & \text{on } \Gamma_N \times]0, T] \\ u(0) = u_0 & \text{in } \Omega. \end{cases} \quad (27)$$

The body force is defined by the following elliptical two-dimensional Gaussian function centred at (ω_x, ω_y) :

$$f(t, x, y) = \lambda \cos\left(\frac{2\pi}{T_0 t}\right) \exp\left(-\frac{(x - \omega_x)^2}{2\sigma_x^2} - \frac{(y - \omega_y)^2}{2\sigma_y^2}\right) \quad (28)$$

and the boundary condition g is given by

$$g(t, x) = \eta(t) \cos(x^2 + y^2) \quad (29)$$

The objective functional \mathcal{J} is chosen to be the following tracking functional:

$$\mathcal{J}(u, \omega) = \frac{1}{2} \int_0^T \int_{\Omega_c} |u - \hat{u}|^2 dx dt + \frac{1}{2} \int_{\Omega_c} |u - \hat{u}|^2 dx + \frac{\kappa}{2} |\omega|^2 \quad (30)$$

where $\kappa > 0$ is a regularization parameter, \hat{u} is a given solution which corresponds to the target solution, $\Omega_c \subseteq \Omega$ and ω is the set of control parameters. The expression of the associated reduced functional is given in [Appendix A](#).

For this numerical case, as shown in [Fig. 3](#) the control region is defined by the union of a set of sub-domains $\bigcup_{i=1}^{16} \{\Omega_i\}$ uniformly distributed in $\Omega = \{x \in \mathbb{R}^2 / 0 < x, y < 1\}$. $\omega_x, \omega_y, \sigma_x, \sigma_y$ and ν are the control parameters. λ, T_0 , and η are arbitrary chosen. The expression of the adjoint equations, the reduced state equations, and the reduced adjoint equations

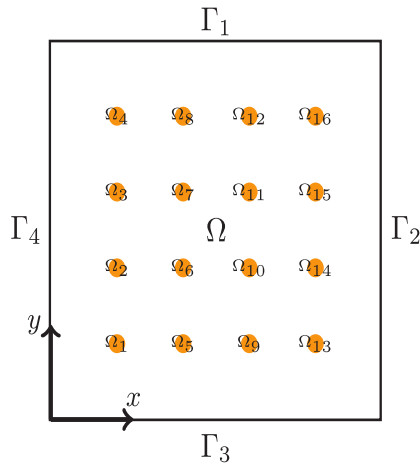


Fig. 3. Domain Ω and control sub-domains for the reaction diffusion problem.

Table 1

Percentage of error between solutions obtained by the full model and solutions obtained with the interpolated ROMs for $(\hat{\omega}_x, \hat{\omega}_y) = (0.55, 0.53)$.

| Method | POD/ROM | Lagrange | RBF | ITSGM |
|---------|---------|----------|--------|-------|
| % error | 0.006% | 37.71% | 40.85% | 3.39% |

are given in [Appendix B](#). The high fidelity model is solved by the first order finite element method on an unstructured mesh with 5511 DOFs.

9.1. Test 1 : control of (ω_x, ω_y)

In a first experiment, the control parameters are the coordinates (ω_x, ω_y) laying in the space of admissible solutions $K \subset \Omega$ and representing the point where the Gaussian force f is applied. Let $\Gamma_D = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 \cup \Gamma_4$, $\Gamma_N = \emptyset$, $\eta(t) = \frac{1}{3}$, $\lambda = \sqrt{2}$, $T_0 = 2.5$, $\sigma_x = 0.15$, $\sigma_y = 0.15$, $T = 10$, $\delta t = 0.005$, and $\nu = 0.01$.

Before starting tests using the proposed sub-optimal control algorithm, the lack of robustness of a ROM obtained by usual interpolation methods (RBF and Lagrange) is highlighted. For this purpose, four precomputed POD bases of dimension 10 at four operating points $(\omega_x, \omega_y) = (0.4, 0.4)$, $(\omega_x, \omega_y) = (0.6, 0.6)$, $(\omega_x, \omega_y) = (0.6, 0.4)$ and $(\omega_x, \omega_y) = (0.4, 0.6)$ are considered. These sampling bases will be used to approximate a representative basis for the new training point $(\hat{\omega}_x, \hat{\omega}_y) = (0.55, 0.53)$. Naturally, the ITSGM interpolation is expected to be more accurate and robust compared to RBF or Lagrange interpolations. This is assessed by the percentage of error listed in [Table 1](#) defined by

$$\%error = 100 \times \frac{\|\hat{u} - u\|_{L^2([0,T],L^2(\Omega))}}{\|\hat{u}\|_{L^2([0,T],L^2(\Omega))}} \quad (31)$$

where \hat{u} is the full solution associated to couple $(\hat{\omega}_x, \hat{\omega}_y)$ and u the solution obtained by the ROM for the same couple of parameters. The results are not particularly surprising given the fact that the ITSGM guarantees by nature that the result is a basis of the same dimension, a property that is not necessarily ensured by usual interpolation methods.

Let now the target \hat{u} be the solution for $(\hat{\omega}_x, \hat{\omega}_y) = (0.35, 0.65)$ of the problem (27), such that \hat{u} is represented in a POD basis of dimension 10. Starting from an initial couple of parameters $(\omega_x^{init}, \omega_y^{init}) = (0.7, 0.3)$ we seek to recover the target \hat{u} by the sub-optimal control methodology ([Algorithm 5](#)). The initial and target solutions are drawn in [Fig. 4](#) for instants $t = T/3$, $t = 2T/3$ and $t = T$. One can see that for a given $(\omega_x, \omega_y) \in \Omega$, the influence of the force applied by the Gaussian function at this point can be clearly noticed on the solution. The POD sampling bases are constructed with respect a distribution of 16 training parameters in the form of a grid obtained by considering the values 0.2, 0.4, 0.6, 0.8 for ω_x and the same for ω_y . To build each POD basis, 400 snapshots equally distributed in the time interval $[0, T]$ are considered. The union of all these sets which forms a set of 6400 snapshots, will be afterwards used to generate the MPS-POD basis.

It is worth reminding that ITSGM is based on local notions, thus the set of POD bases used for the interpolation must be carefully chosen in a sufficiently small neighbourhood. Here only four POD bases, associated to the four nearest parameters to the parameter where the interpolation is done, are considered. The evolution of control parameters (ω_x, ω_y) and the cost functional for the various considered methods is reported in [Figs. 5](#) and [6](#).

The PGD reduced cost functional has the best convergence behaviour. At iteration 9, the PGD-sub-optimal control gives already a sufficient approximation of the desired parameters $(\hat{\omega}_x, \hat{\omega}_y)$. In the next iterations, the control parameters stagnate and the PGD keeps minimizing the error of the achieved solution. From iteration 18, the functional start to stabilize, indicating that the PGD method has attained its maximal accuracy. For ITSGM and MPS methods, the maximal accuracy is achieved at iteration 8 and 10, respectively. This can be seen from the stagnation of the functional along with control parameters. In order to improve the accuracy of the solution produced by MPS or ITSGM methods, a coupling with the PGD can be

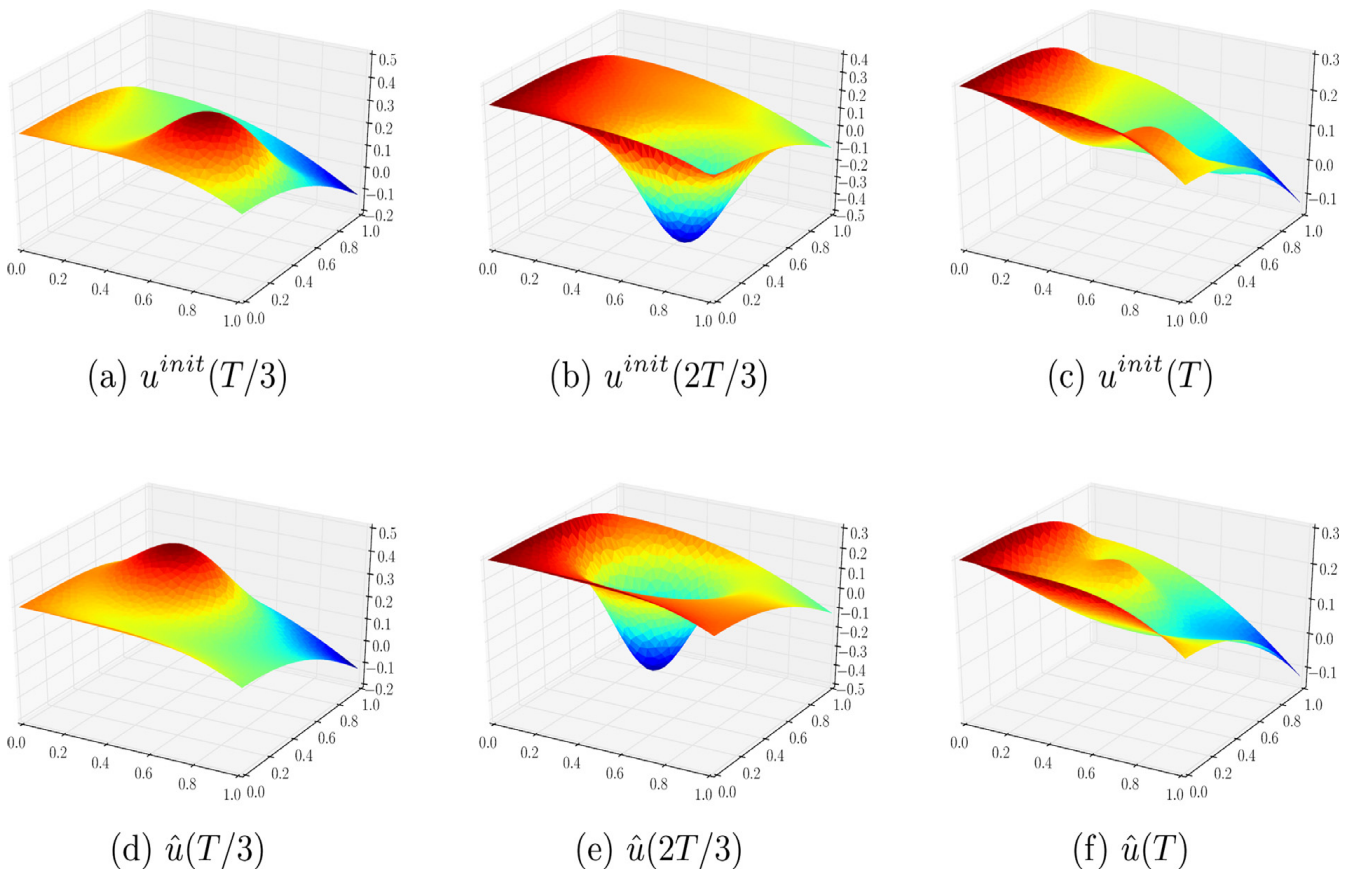


Fig. 4. Representation of the target solution \hat{u} associated to $(\hat{\omega}_x, \hat{\omega}_y) = (0.35, 0.65)$, and of the control initial guess solution u^{init} associated to $(\omega_x^{init}, \omega_y^{init}) = (0.7, 0.3)$.

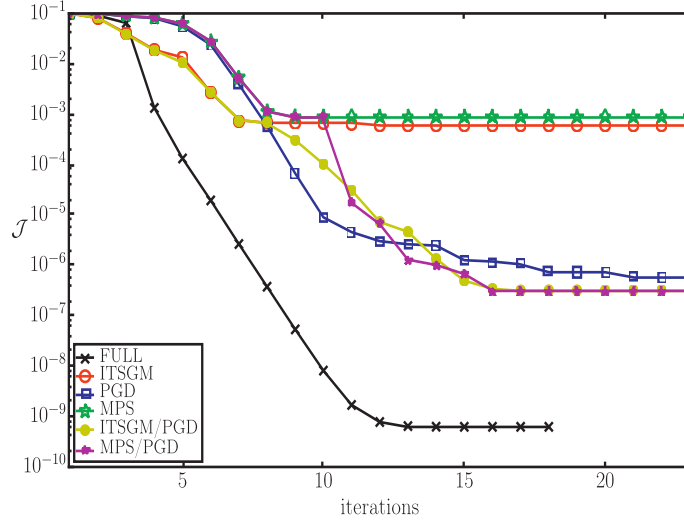


Fig. 5. Evolution of the cost functional.

performed. It is worth noting that at the convergence of the reduced optimal control algorithms using PGD, ITSGM/PGD and MPS/PGD, the differences between the final values of the functional are only due to the initial error of the ROM provided to start the PGD method. Finally, it can be reported that usual interpolation methods RBF and Lagrange behaved as was expected and inevitably result in a non convergence of the sub-optimal control algorithm.

In order to compare simulation time for each method, we have considered the ratio between the CPU time \mathcal{T}^{full} for the full adjoint based optimal control and the CPU time³ $\mathcal{T}_{online}^{method}$ needed for the sub-optimal control algorithm. These ratios are reported in Table 2. All the reduced order approaches allow an important CPU time gain compared to the full approach. At iteration 12, the simulation time is respectively divided by 94 ,98 and 134 for PGD , ITSGM and MPS. At this iteration the percentage of error between the target solution and the converged solutions obtained by PGD, ITSGM and MPS sub-optimal control methods is respectively⁴ of order 0.39%, 5.47% and 6.61%. By coupling the MPS and ITSGM methods with the PGD

³ It corresponds to the on-line time.

⁴ The percentage error is defined by (31).

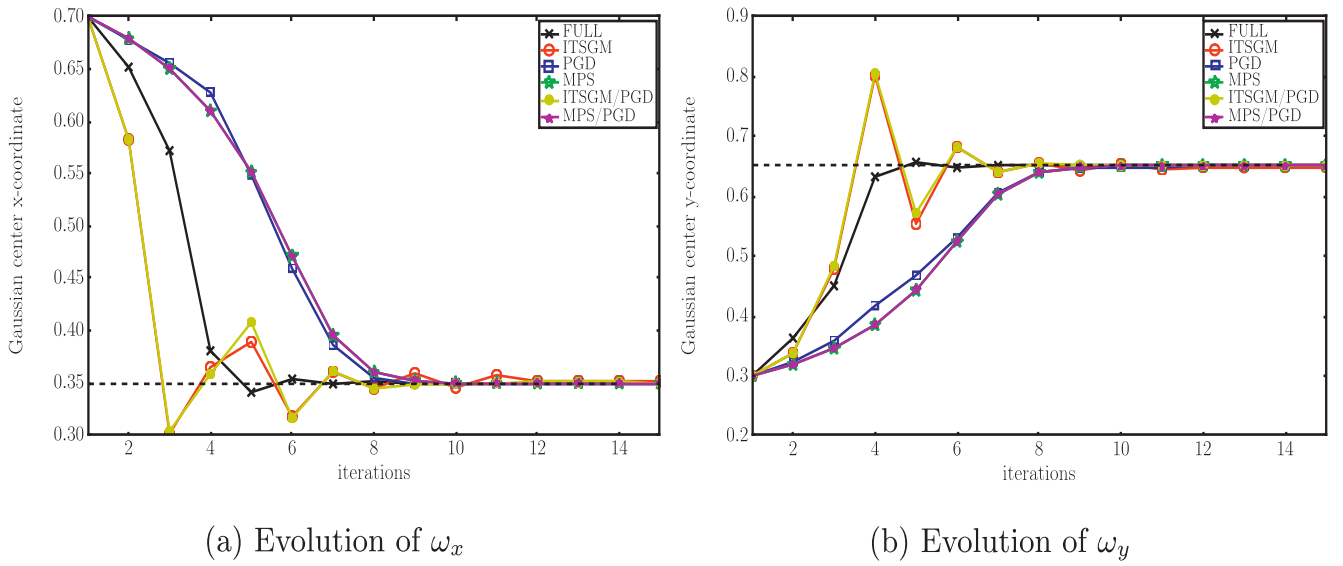


Fig. 6. Control parameters evolution with respect to control iterations.

Table 2

Time ratio between the time of full control and the time of reduced control, control parameters values (ω_x , ω_y), functional values and percentage errors between the obtained solution and the target solution.

| Method/iteration | $\mathcal{T}^{\text{full}} / \mathcal{T}_{\text{online}}^{\text{method}}$ | (ω_x, ω_y) | Functional | % Error |
|---------------------|---------------------------------------------------------------------------|------------------------|-----------------------|---------|
| full (iter.12) | 1 | (0.350,0.650) | 7.6×10^{-10} | 0.003% |
| ITSGM (iter.12) | 98 | (0.352,0.649) | 5.9×10^{-4} | 5.47% |
| MPS (iter.12) | 134 | (0.350,0.651) | 8.9×10^{-4} | 6.61% |
| PGD (iter.12) | 94 | (0.350,0.649) | 2.9×10^{-6} | 0.39% |
| ITSGM/PGD (iter.16) | 72 | (0.350,0.650) | 3.3×10^{-7} | 0.13% |
| MPS/PGD (iter.16) | 87 | (0.350,0.650) | 2.9×10^{-7} | 0.12% |

method, it can be noticed that the accuracy of the achieved solution is considerably improved and attain 0.12% and 0.13%, respectively (see Table 2).

9.2. Test 2 : control of $(\nu, \sigma, \omega_x, \omega_y)$

The application of the sub-optimal control algorithm using the interpolation method ITSGM or the MPS method depends on precomputed data. When the control is performed on more than two parameters this off-line phase can be very expansive and these approaches can not be used. The goal of this paragraph is to assert applicability and power of the PGD-sub-optimal control method to deal with this situation. In the following test case, four parameters ν , ω_x and ω_y and $\sigma = \sigma_x = \sigma_y$ are considered to control the non-linear reaction-diffusion problem (27). We set $\eta(t) = \cos(t)$, $\lambda = 2\sqrt{2}$, $T_0 = 2$, $T = 5$, $\Gamma_D = \Gamma_1 \cup \Gamma_3$ and $\Gamma_N = \Gamma_2 \cup \Gamma_4$. Now, the target solution \hat{u} corresponds the solution of (27) for $(\hat{\nu}, \hat{\sigma}, \hat{\omega}_x, \hat{\omega}_y) = (0.001, 0.35, 0.3, 0.7)$. Starting from an arbitrary first guess $(\nu^{\text{init}}, \sigma^{\text{init}}, \omega_x^{\text{init}}, \omega_y^{\text{init}}) = (0.005, 0.1, 0.6, 0.3)$ we aim to test the ability of the PGD-sub-optimal control method to recover the target solution \hat{u} with significant gains in CPU-time. The difference between the dynamics of the initial guess and target states is depicted in Fig. 7. The reader can obviously see that a variation in the four directions of the control parameter, leads to a remarkable variation in the rich dynamics of the solution. The high fidelity model considered on an unstructured triangular mesh of 33025 DOFs (P_1 finite element) is solved using the implicit Euler scheme such that $\delta t = 10^{-3}$. 400 snapshots are afterwards considered in order to construct a POD basis of size 10 for the initial guess and target solutions. The space of admissible solutions is given by

$$K = \{(\nu, \sigma, \omega_x, \omega_y) \in \mathbb{R}_+^2 \times \mathbb{R}^2 \mid 10^{-4} < \nu < 1; \quad 0 < \sigma, \omega_x, \omega_y < 1\} \quad (32)$$

Fig. 8 reports the evolution of the functional according to control iterations. A good convergence behaviour of the PGD-sub-optimal control approach was observed. This result is approved by Fig. 9 which shows that the looked after parameters were perfectly recovered after 20 iterations. At this stage, the percentage of relative error of the achieved solution is 2.8%. In the next iterations, the parameters are stagnated and only the accuracy of the solution is improved (the percentage of error attains 1.1% after 25 iterations). The full control approach on the other hand attains convergence at iteration 25. As shown in Table 3, important gain in CPU-time using the PGD approach is achieved. More precisely, about 98% of CPU-time needed for the full approach was saved.

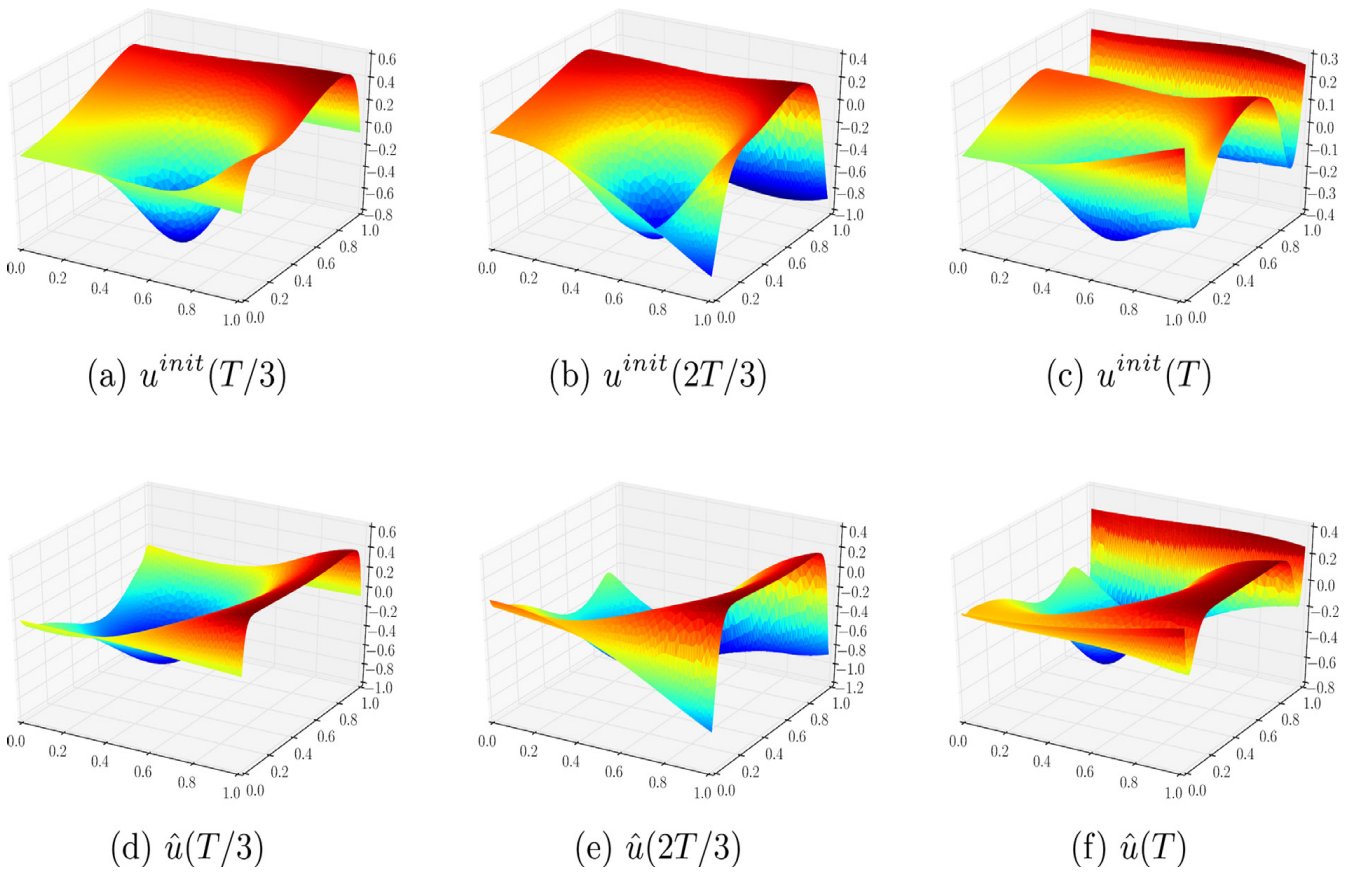


Fig. 7. Representation of the target solution associated to $(\hat{\omega}_x, \hat{\omega}_y)$, and of the control initial guess solution associated to $(\omega_x^{init}, \omega_y^{init})$.

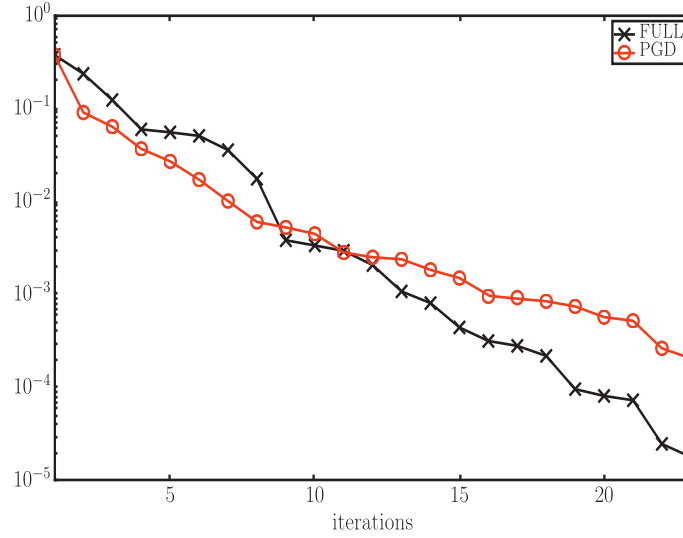


Fig. 8. Evolution of the functional with respect to control algorithm iterations.

10. Application to the control of Burgers equations

In this section, the problem of control of the 2D Burgers equations is numerically investigated. Let $T > 0$ and $\Omega \subset \mathbb{R}^2$ a square domain of side d and boundaries Γ . The two dimensional Burgers equations with initial condition $\mathbf{u}_0 \in [L^2(\Omega)]^2$ submitted to a body force $\mathbf{f}(\omega) \in L^2(]0, T], [L^2(\Omega)]^2)$ are

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} = \mathbf{f}(\omega) & \text{in } \Omega \times]0, T] \\ \mathbf{u} = \mathbf{0} & \text{on } \Gamma \times]0, T] \\ \mathbf{u}(0) = \mathbf{u}_0 & \text{in } \Omega. \end{cases} \quad (33)$$

Initial condition and body forces are respectively given as follows:

$$\mathbf{u}_0(x, y) = \begin{pmatrix} y - 0.5 \\ 0.5 - x \end{pmatrix} \quad (34)$$

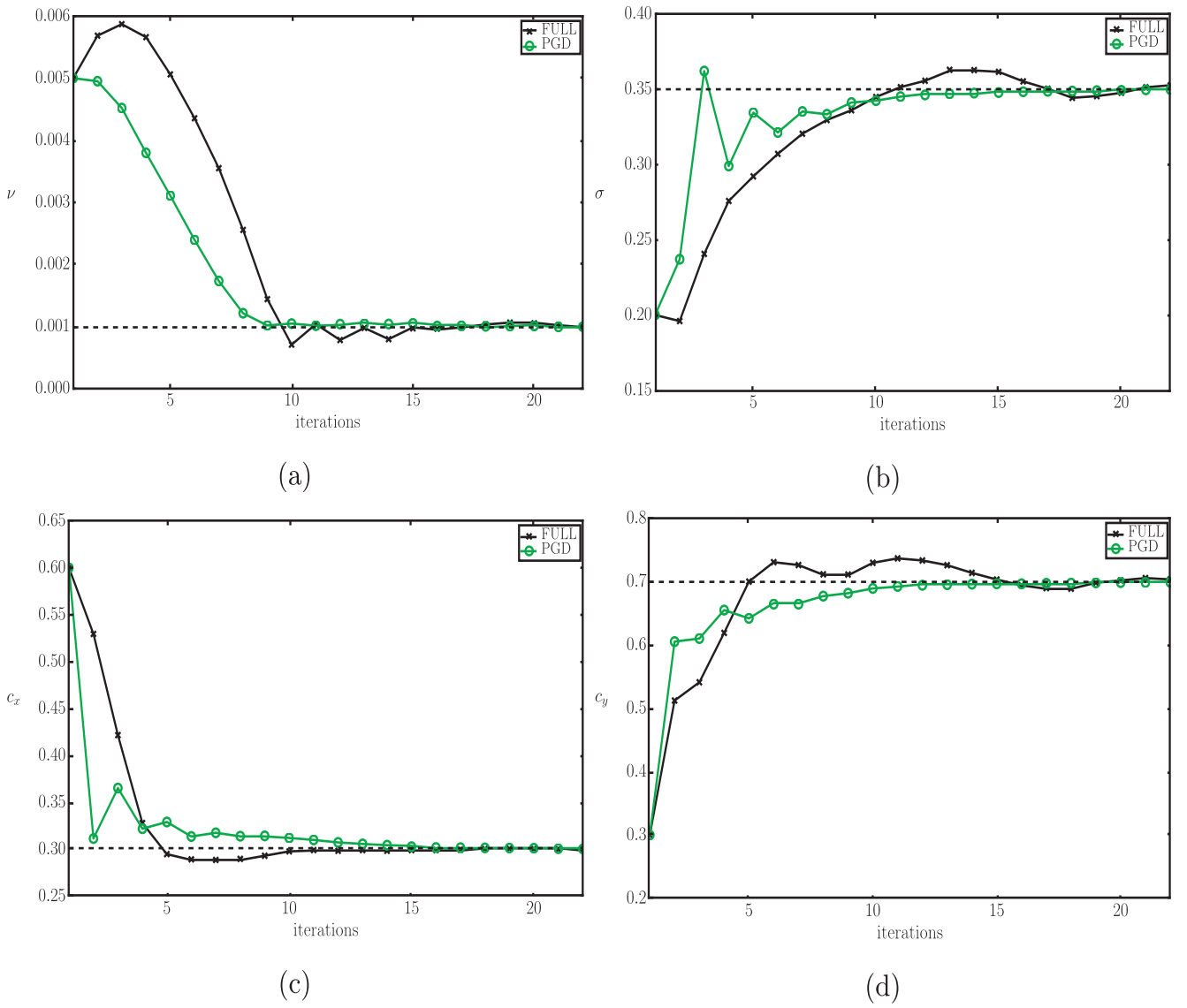


Fig. 9. Evolution of control parameters with respect to control algorithm iterations.

Table 3

Time ratio between the time of full control and the time of reduced control, functional values and percentage errors between the obtained solution and the target solution.

| Method/iteration | $\mathcal{T}^{full} / \mathcal{T}^{method}_{online}$ | Functional | % Error |
|------------------|------------------------------------------------------|-----------------------|---------|
| full (iter. 25) | 1 | 1.05×10^{-5} | 0.34% |
| PGD (iter. 20) | 94 | 5.69×10^{-4} | 2.8% |
| PGD (iter. 25) | 73 | 9.53×10^{-5} | 1.1% |

and

$$\mathbf{f}(t, x, y, \omega) = \lambda \begin{pmatrix} \omega_y - y \\ x - \omega_x \end{pmatrix} e^{-\frac{(x - \omega_x)^2 + (y - \omega_y)^2}{2\sigma^2}} \quad (35)$$

The expression of the objective functional \mathcal{J} is the same as in the previous section and is defined by (30). Here we assume $\Omega_c = \Omega$, $d = 1$, $\lambda = \frac{\sqrt{2}}{2}$, $T = 6$ and $\nu = 10^{-3}$. The control is performed on the couple of parameters (ω_x, ω_y) varying in the space of admissible solutions $K \subset \Omega$ defined as

$$K = \{(\omega_x, \omega_y) \in \mathbb{R}^2 / 0.25 < \omega_x < 0.75; 0.25 < \omega_y < 0.85\} \quad (36)$$

Starting from $(\omega_x^{init}, \omega_y^{init}) = (0.45, 0.75)$ the goal is to recover the target parameters $(\hat{\omega}_x, \hat{\omega}_y) = (0.63, 0.52)$ associated to the solution \hat{u} of (33). Streamlines associated to the initial and target solutions are drawn in Fig. 10 for $t = T/3$, $t = 2T/3$ and T . This figure describes the rich dynamics between the initial and target state solutions. The initial condition \mathbf{u}_0 is a divergence free field, which rotate in reverse with respect to the applied body force localized by the Gausssian function. It can be observed that the recirculation zone initially centered at $(\omega_x, \omega_y) = (0.5, 0.5)$ is reversed and transported over time to the point where the body force is applied. The expression of the adjoint equations, the reduced state equations, and the

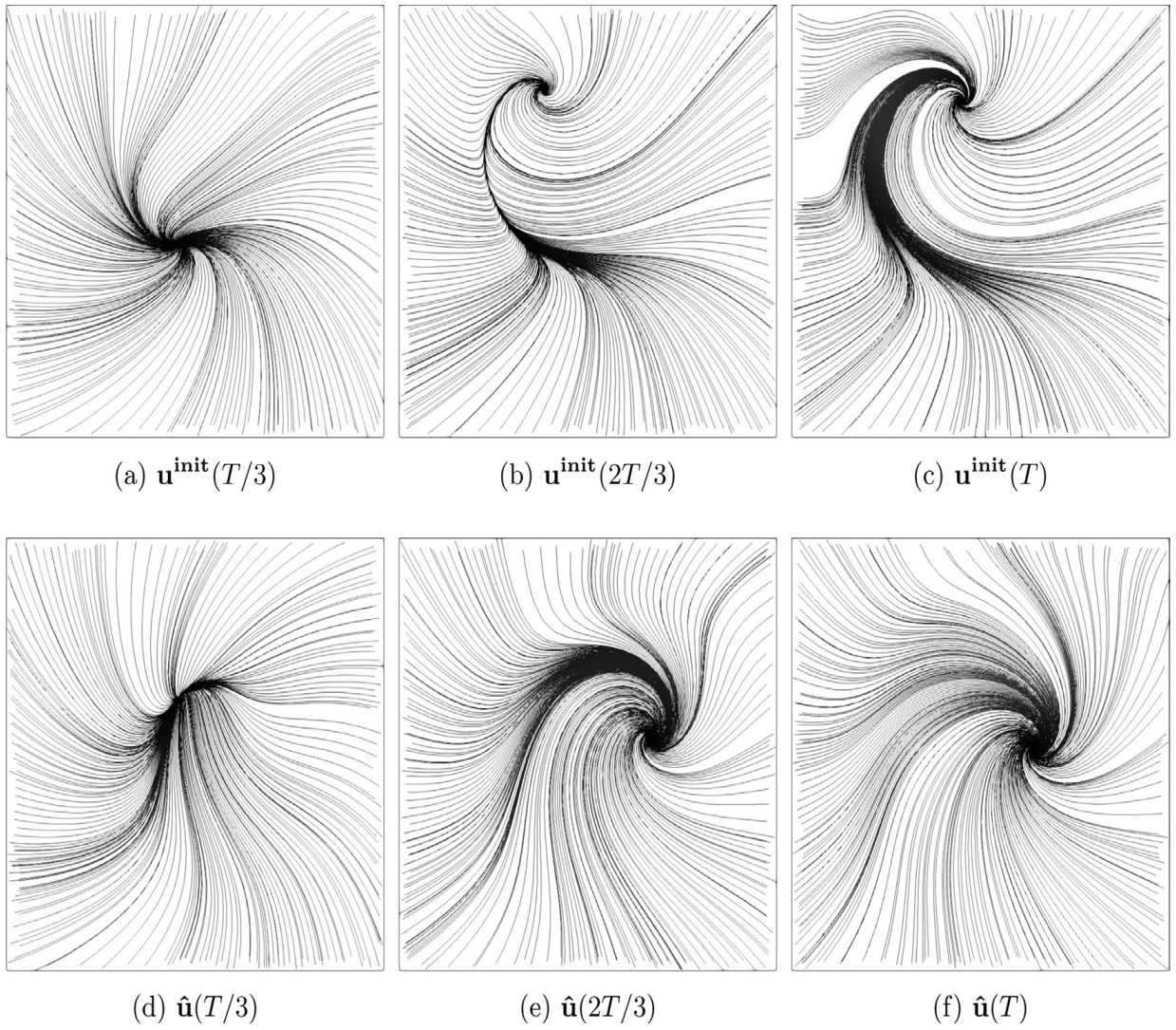


Fig. 10. Streamlines at $T/3$, $2T/3$ and T of the target solution associated to $(\hat{\omega}_x, \hat{\omega}_y) = (0.63, 0.52)$ (bottom), and of the control initial guess solution associated to $(\omega_x^{init}, \omega_y^{init}) = (0.45, 0.75)$ (top).

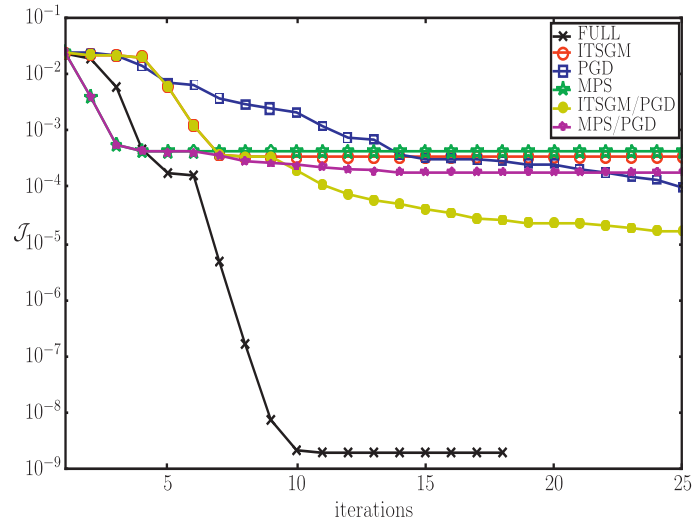


Fig. 11. Evolution of the cost functional.

reduced adjoint state equations, are given in [Appendix C](#). The high fidelity model is discretized in time using an implicit Euler scheme (with the time step $\delta t = 0.002$) and in space using a second order finite element method on an unstructured mesh of 4705 DOFs. 500 snapshots for each parameter in the grid constructed by considering the values 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 for ω_x and the same for ω_y . This distribution of training parameters will be used to construct the POD bases sampling.

For the MPS method the union of these snapshot sets is considered to build the associated POD basis. During the control algorithm, for every suggested parameter (ω_x, ω_y) , only the four nearest POD bases are considered to perform ITSGM. [Fig. 11](#) and [Fig. 12](#) show for each method, the evolution of the functional and control parameters with respect to control

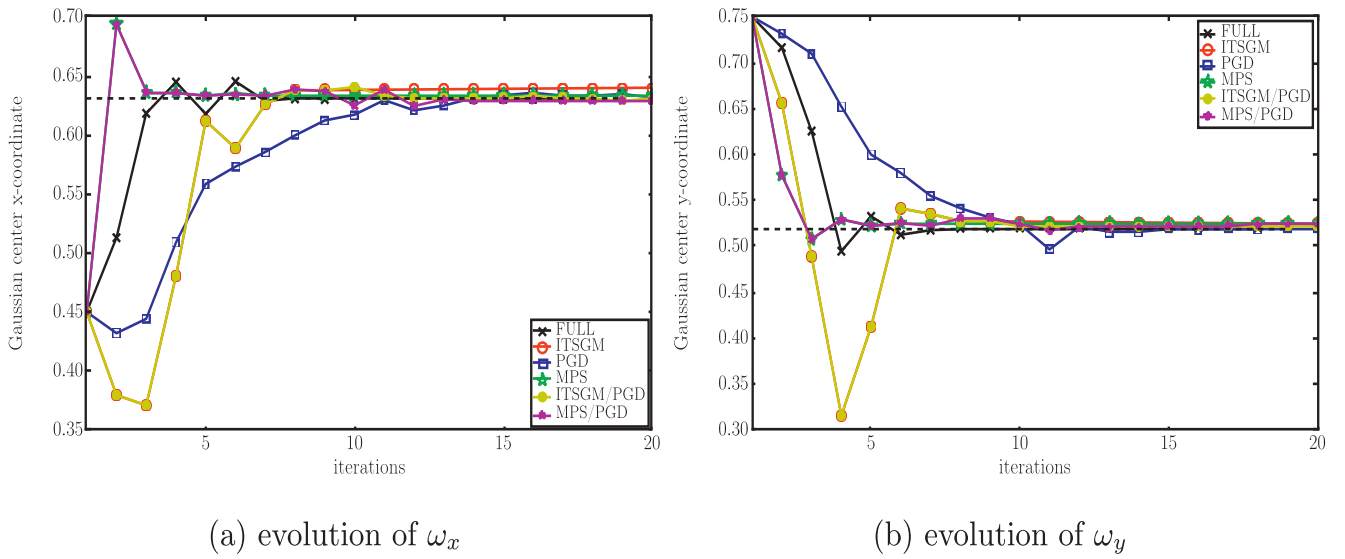


Fig. 12. Control parameters evolution with respect to control iterations.

Table 4

Time ratio between the time of full control and the time of reduced control, control parameters values (ω_x , ω_y), functional values and percentage errors between the obtained solution and the target solution

| Method/iteration | $\mathcal{T}^{full}/\mathcal{T}^{method}_{online}$ | (ω_x, ω_y) | Functional | % Error |
|----------------------|----------------------------------------------------|------------------------|-----------------------|-------------|
| Full (iter. 11) | 1 | (0.630,0.520) | 1.98×10^{-9} | $10^{-4}\%$ |
| ITSGM (iter. 10) | 351 | (0.638,0.526) | 3.3×10^{-4} | 5.17% |
| MPS (iter. 6) | 754 | (0.633,0.524) | 4.1×10^{-4} | 5.45% |
| PGD (iter. 15) | 93 | (0.635,0.517) | 3.2×10^{-4} | 5.22% |
| ITSGM-PGD (iter. 15) | 90 | (0.631,0.521) | 3.3×10^{-5} | 1.53% |
| MPS-PGD(iter. 15) | 89 | (0.629,0.522) | 1.8×10^{-4} | 3.90% |

iterations. Table 4 reports the percentage of relative errors between the target solution and solutions obtained at the end of control algorithm.

These results indicate that all methods behave the same way as the case of non-linear reaction-diffusion equation. Usual methods of interpolation (RBF and Lagrange) are inefficient and fail in sub-optimal control. While MPS, ITSGM and PGD sub-optimal control approaches succeed to converge. The maximal accuracy of MPS, ITSGM and PGD sub-optimal control approach is attained respectively at iteration 6, 10 and 15. MPS method is the fastest, followed by ITSGM and PGD methods. For MPS (resp. ITSGM and PGD) the sub-optimal control CPU time represents 0.13% (resp. 0.28% and 1%) of the CPU time required to solve the full control problem with a relative error of 5.45% (resp. 5.17% and 5.22%). In order to improve the accuracy of MPS and ITSGM methods it is possible to couple these approaches with PGD. The PGD is used when the functional stagnates (from iteration 6 for MPS and 10 for ITSGM). A better accuracy for MPS and ITSGM methods is then achieved when a coupling with the PGD method is performed. The improvement of the solution obtained either by MPS or ITSGM after switching to PGD can be observed from results of Fig. 11. A different behaviour of the functional using the PGD, ITSGM/PGD and MPS/PGD can be noticed at the convergence. We believe that this behaviour is due to the initial ROM error inherited over control iterations, which eventually can have a slight effect on the accuracy of the converged solution.

11. Discussion and conclusion

The aim of this paper was to illustrate the reliability and potential gains that can be offered using ITSGM (Interpolation on a Tangent Subspace of the Grassmann Manifold), MPS (Multiple Parametrized Snapshots method), RBF interpolation (Radial Basis Functions), Lagrange interpolation and PGD (Proper Generalized Decomposition) methods within the sub-optimal control methodology. Numerical investigations on the non linear reaction-diffusion and Burgers equations, have demonstrated that the lack of robustness of the classical interpolation techniques (RBF and Lagrange methods) to adapt a ROM for a new configuration makes them not applicable in the proposed sub-optimal control methodology. On the other hand, the good results obtained using MPS and the proposed approaches (PGD and ITSGM) in ROMs adaptation qualified them to be applicable efficiently in sub-optimal control. The convergence behaviour (percentage of error of the obtained solutions between 0.2% and 6%) along with the important gains in CPU-time (time divided by 50 to 750 compared with the full control approach) shows that the proposed reduced control strategy represents a valuable alternative to the classical adjoint based control approach. The MPS method appears to be the fastest and the PGD the most accurate. ITSGM on the other hand gives a good compromise between CPU-time and accuracy. For MPS and ITSGM methods, the selection of operating points for which the POD bases must be constructed can be a limitation, more particularly when the dimension of the control parameter exceeds 2. We aware also that the generation of a MPS-POD basis might be very costly due the huge eigenvalue

problem of size $N_s^2 \times N_p^2$ to be constructed and solved (N_s is the number of snapshots and N_p the number of control parameters). Contrary to MPS and ITSGM the PGD method does not require any precomputed data. As a matter of fact, it has the ability to perform as a ROM corrector based on its current errors and independently of any external information. However, this method in sub-optimal control present some challenges to perform faster and in an optimal way. In fact, the spatial problems that arise from the fixed point process of the PGD method are often ill-conditioned. This issue forces us to use costly direct solvers to solve these problems. So a suitable preconditioner has to be developed for this kind of problems in order to efficiently embed iterative solvers and consequently increase the speed of the PGD. Finally, since the results obtained with the proposed methods ITSGM and PGD in the sub-optimal control approach are promising, the next step of our work will be to extend this study for mixed problems such as Stokes and Navier-Stokes equations.

Acknowledgment

This material is based upon work financially supported by the Nouvelle-Aquitaine region and CPER/FEDER bâtiment durable (P-2017-BAFE-102).

Appendix A. Expression of the reduced functional

Let \hat{u} be given in the following separated form:

$$\hat{u}(x, t) \approx \sum_{j=1}^{\hat{m}} \hat{\alpha}^j(t) \hat{\varphi}^j(x) \quad (\text{A.1})$$

For a given basis $\hat{\Phi} = \{\hat{\varphi}^i\}_{i=1, \dots, \hat{m}}$, associated to a parameter $\hat{\omega}$, the reduced cost functional, obtained by embedding (10) and (A.1) in the functional expression (30), writes

$$\mathcal{J}_{\Phi}(\alpha_m, \omega) = \frac{1}{2} \sum_{k=1}^m \sum_{j=1}^m A_{k,j} \left(\int_0^T \alpha_k \alpha_j dt + \alpha_k(T) \alpha_j(T) \right) \quad (\text{A.2})$$

$$+ \frac{1}{2} \sum_{l=1}^{\hat{m}} \sum_{q=1}^{\hat{m}} \hat{A}_{l,q} \left(\int_0^T \hat{\alpha}_l \hat{\alpha}_q dt + \hat{\alpha}_l(T) \hat{\alpha}_q(T) \right) \quad (\text{A.3})$$

$$- \sum_{k=1}^m \sum_{l=1}^{\hat{m}} C_{kl} \left(\int_0^T \alpha_k \hat{\alpha}_l dt + \alpha_k(T) \hat{\alpha}_l(T) \right) + \frac{\kappa}{2} |\omega|^2. \quad (\text{A.4})$$

where

$$A_{kj} = \langle \varphi^k, \varphi^j \rangle_{L^2(\Omega_c)} \quad \hat{A}_{lq} = \langle \hat{\varphi}^l, \hat{\varphi}^q \rangle_{L^2(\Omega_c)} \quad C_{kl} = \langle \varphi^k, \hat{\varphi}^l \rangle_{L^2(\Omega_c)}$$

Appendix B. Expression of equations for the control of reaction diffusion problem

(a) Adjoint equations of the full problem

$$\begin{cases} \frac{\partial}{\partial t} \xi + \nu \Delta \xi - u \xi = (u - \hat{u}) \chi_{\Omega_c} & \text{in } \Omega \times]0, T] \\ \xi = 0 & \text{on } \Gamma \times]0, T] \\ \xi(T) = (\hat{u}(T) - u(T)) \chi_{\Omega_c} & \text{in } \Omega. \end{cases}$$

where χ_{Ω_c} stands for the mask function, which is equal to 1 in Ω_c and 0 in $\Omega \setminus \Omega_c$.

(b) State reduced order model

The state equation in a reduced form is given $\forall i \in 1, \dots, m$ by

$$\begin{cases} \sum_{j=1}^m M_{ij} \frac{d}{dt} \alpha^j + \nu \sum_{j=1}^m R_{ij} \alpha^j + \frac{1}{2} \sum_{j,k=1}^m N_{ijk} \alpha^j \alpha_k = F_i \\ \sum_{j=1}^m M_{ij} \alpha^j(0) = \langle u_0, \varphi^i \rangle_{L^2(\Omega)} \end{cases}$$

with

$$\begin{aligned} M_{ij} &= \langle \varphi^j, \varphi^i \rangle_{L^2(\Omega)} & N_{ijk} &= \langle \varphi^j \varphi^k, \varphi^i \rangle_{L^2(\Omega)} \\ R_{ij} &= \langle \nabla \varphi^j, \nabla \varphi^i \rangle_{L^2(\Omega)} + \frac{1}{\nu \epsilon} \langle \varphi^j, \varphi^i \rangle_{L^2(\Gamma_D)} & F_i &= \langle f, \varphi^i \rangle_{L^2(\Omega)} + \frac{1}{\epsilon} \langle \bar{g}, \varphi^i \rangle_{L^2(\Gamma_D)} \end{aligned}$$

The essential boundary condition is enforced in a weak integral form using the penalty method [30]. It consists in writing u in the boundary as

$$u|_{\Gamma_D} = g - \epsilon \nu \frac{\partial u}{\partial n}$$

where ϵ is a small parameter. So that we can replace the essential boundary condition by the alternative natural boundary condition

$$\nu \frac{\partial u}{\partial n} = \frac{u|_{\Gamma_D} - g}{\epsilon}$$

This remark explains the presence of the elements B_{ij} and G_i in the ROM equations. As $\epsilon \rightarrow 0$ the domain boundary will converge to the original boundary condition.

(c) Adjoint reduced order model

The adjoint equation in a reduced form is given $\forall q \in 1, \dots, m$ by

$$\begin{cases} \sum_{i=1}^m M_{iq} \frac{d}{dt} \xi_i - \nu \sum_{i=1}^m R_{iq} \xi_i - \sum_{i,k=1}^m N_{iqk} \alpha_k \xi_i = \sum_{i=1}^m A_{qi} \alpha_i - \sum_{i=1}^{\hat{m}} C_{qi} \hat{\alpha}_i \\ \sum_{i=1}^m M_{iq} \xi_q(T) = \sum_{i=1}^{\hat{m}} C_{qi} \hat{\alpha}_i(T) - \sum_{i=1}^m A_{qi} \alpha_i(T) \end{cases}$$

Appendix C. Expression of equations for the control of Burgers problem

(a) Adjoint equations of the full problem

$$\begin{cases} \frac{\partial}{\partial t} \xi + \nu \Delta \xi + (\nabla \cdot u) \xi - (\nabla u)^T \xi + u \cdot \nabla \xi = \mathbf{u} - \hat{\mathbf{u}} & \text{in } \Omega \times]0, T] \\ \xi = 0 & \text{on } \Gamma \times]0, T] \\ \xi(T) = \hat{\mathbf{u}}(T) - \mathbf{u}(T) & \text{in } \Omega. \end{cases}$$

(b) State reduced order model

The state equation in a reduced form is given $\forall i \in 1, \dots, m$ by

$$\begin{cases} \sum_{j=1}^m M_{ij} \frac{d}{dt} \alpha^j + \nu \sum_{j=1}^m R_{ij} \alpha^j + \sum_{j,k=1}^m \tilde{N}_{ijk} \alpha^j \alpha_k = F_i \\ \sum_{j=1}^m M_{ij} \alpha^j(0) = \langle \mathbf{u}_0, \varphi^i \rangle_{L^2(\Omega)} \end{cases}$$

where the tensor \tilde{N} is defined by $\tilde{N}_{ijk} = \int_{\Omega} \varphi^i \cdot \nabla \varphi^k \varphi^j dx$.

(c) Adjoint reduced order model

The adjoint equation in a reduced form is given $\forall q \in 1, \dots, m$ by

$$\begin{cases} \sum_{i=1}^m M_{iq} \frac{d}{dt} \xi_i - \nu \sum_{i=1}^m R_{iq} \xi_i - \sum_{i,k=1}^m (\tilde{N}_{iqk} + \tilde{N}_{ikq}) \alpha_k \xi_i = \sum_{i=1}^m A_{qi} \alpha_i - \sum_{i=1}^{\hat{m}} C_{qi} \hat{\alpha}_i \\ \sum_{i=1}^m M_{iq} \xi_q(T) = \sum_{i=1}^{\hat{m}} C_{qi} \hat{\alpha}_i(T) - \sum_{i=1}^m A_{qi} \alpha_i(T) \end{cases}$$

The expressions of M , C , A and R are the same as in Appendix B.

References

- [1] M. Bergmann, L. Cordier, J.-P. Brancher, Optimal rotary control of the cylinder wake using proper orthogonal decomposition reduced-order model, *AIP Phys. Fluids* 17 (2005) 097101.
- [2] S. Ravindran, A reduced-order approach for optimal control of fluids using proper orthogonal decomposition, *Int. J. Numer. Methods Fluids* 34 (2000) 425–448.
- [3] A. Tallet, C. Allery, C. Leblond, Optimal flow control using a pod based reduced-order model, *Numer. Heat Transf. Part B* 170 (2016) 1–24.
- [4] D. Amsallem, C. Farhat, An interpolation method for adapting reduced-order models and application to aeroelasticity, *AIAA J.* 46 (2008) 1803–1813.
- [5] P. Ladev ze, A. Nouy, On a multiscale computational strategy with time and space homogenization for structural mechanics, *Comput. Methods Appl. Mech. Eng.* 192 (28) (2003) 3061–3087. *Multiscale Computational Mechanics for Materials and Structures*.
- [6] P. Ladeveze, *Nonlinear Computational Structural Mechanics: New Approaches and Non-Incremental Methods of Calculation*, Springer, New York, 1999, pp. 111–133.
- [7] P. Ladev ze, J.-C. Passieux, D. N ron, The latin multiscale computational method and the proper generalized decomposition, *Comput. Methods Appl. Mech. Eng.* 199 (21) (2010) 1287–1296. *Multiscale Models and Mathematical Aspects in Solid and Fluid Mechanics*.
- [8] A. Nouy, generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations., *Comput. Methods Appl. Mech. Eng.*, Elsevier 196 (45–48) (2007) 521–4537.
- [9] A. Nouy, O.P.L. Maitre, Generalized spectral decomposition for stochastic nonlinear problems, *J. Comput. Phys.* 228 (2009) 202–235.

- [10] A. Ammar, B. Mokdad, F. Chinesta, R. Keunings, A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids, *J. Non-Newton. Fluid Mech.* 139 (3) (2006) 153–176.
- [11] B. Mokdad, E. Pruliere, A. Ammar, F. Chinesta, On the simulation of kinetic theory models of complex fluids using the Fokker–Planck approach, *Appl. Rheol.* 17 (2) (2007) 1–14.
- [12] E. Pruliere, F. Chinesta, A. Ammar, On the deterministic solution of multidimensional parametric models using the proper generalized decomposition, *Math. Comput. Simul.* 81 (4) (2010) 791–810.
- [13] A. Dumon, C. Allery, A. Ammar, Proper general decomposition (PGD) for the resolution of Navier–Stokes equations, *J. Comput. Phys.* 230 (4) (2011) 1387–1407.
- [14] A. Dumon, C. Allery, A. Ammar, Proper generalized decomposition method for incompressible flows in stream-vorticity formulation, *Eur. J. Comput. Mech.* 19 (5-7) (2011) 591–617.
- [15] A. Dumon, C. Allery, A. Ammar, Proper generalized decomposition method for incompressible Navier–Stokes equations with a spectral discretization, *Appl. Math. Comput.* 219 (15) (2013) 8145–8162.
- [16] A. Dumon, C. Allery, A. Ammar, Simulation of heat and mass transport in a square lid-driven cavity with proper generalized decomposition (PGD), *Numer. Heat Transf. Part B: Fundam.* 63 (1) (2013) 18–43.
- [17] C. Leblond, C. Allery, A priori space-time separated representation for the reduced order modeling of low Reynolds number flows, *Comput. Methods Appl. Mech. Eng.* 274 (2014) 264–288.
- [18] F. Chinesta, A. Ammar, P. Joyot, The nanometric and micrometric scales of the structure and mechanics of materials revisited: an introduction to the challenges of fully deterministic numerical descriptions, *Int. J. Multiscale Comput. Eng.* 6/3 (2008) 191–213.
- [19] B. Bognet, A. Leygue, F. Chinesta, Separated representations of 3D elastic solutions in shell geometries, *Adv. Model. Simul. Eng. Sci.* 1 (1) (2014) 4.
- [20] F. Chinesta, P. Ladeveze, E. Cueto, A short review on model order reduction based on proper generalized decomposition, *Arch. Comput. Methods Eng.* 18 (4) (2011) 395–404.
- [21] C. Ghnatios, F. Masson, A. Huerta, A. Leygue, C. E., F. Chinesta, Proper generalized decomposition based dynamic data-driven control of thermal processes, *Comput. Methods Appl. Mech. Eng.* 213–216 (2) (2012) 29–41.
- [22] F. Chinesta, A. Leygue, F. Bordeu, J.V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, A. Huerta, PGD-based computational vademecum for efficient design, optimization and control, *Arch. Comput. Methods Eng.* 20 (1) (2013) 31–59.
- [23] E. Nadal, F. Chinesta, P. Díez, F. Fuenmayor, F. Denia, Real time parameter identification and solution reconstruction from experimental data using the proper generalized decomposition, *Comput. Methods Appl. Mech. Eng.* 296 (2015) 113–128.
- [24] R. García-Blanco, D. Borzacchiello, P. Díez, F. Chinesta, P. Díez, Monitoring a PGD solver for parametric power flow problems with goal-oriented error assessment, *Int. J. Numer. Methods Eng.* (2017) 529–552.
- [25] P.G. Ciarlet, *Linear and Nonlinear Functional Analysis with Applications*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2013.
- [26] L. Sirovich, Turbulence and the dynamics of coherent structures: I, II and III, *Q. Appl. Math.* 45 (1987) 461–590.
- [27] A. Absil, R. Mahony, R. Sepulchre, Riemann geometry of Grassmann manifolds with a view on algorithmic computation, *Acta Appl. Math.* 80 (2) (2004) 199–220.
- [28] R. Wald, *General relativity*, The University of Chicago Press, 1984.
- [29] D.S. Broomhead, D. Lowe, Multivariable functional interpolation and adaptive networks, *Complex Syst.* 2 (1988) 321–355.
- [30] E. Liberge, A. Hamdouni, Reduced order modelling method via proper orthogonal decomposition (pod) for flow around an oscillating cylinder, *J. Fluids Struct.* 26 (2) (2010) 292–311.

