Nonincremental proper generalized decomposition solution of parametric uncoupled models defined in evolving domains

Amine Ammar[1], Elías Cueto[2,*,†] and Francisco Chinesta[3]

[1]*Arts et Métiers ParisTech, 2 Boulevard du Ronceray, BP 93525, F-49035 Angers cedex 01, France*
[2]*Aragón Institute of Engineering Research (I3A), Universidad de Zaragoza, Edificio Betancourt, María de Luna, s.n. 50018 Zaragoza, Spain*
[3]*EADS Corporate Foundation International Chair, École Centrale de Nantes, 1, Rue de la Noë, 44300 Nantes, France*

SUMMARY

This work addresses the recurrent issue related to the existence of reduced bases related to the solution of parametric models defined in evolving domains. In this first part of the work, we address the case of decoupled kinematics, that is, models whose solution does not affect the domain in which they are defined. The chosen framework considers an updated Lagrangian description of the kinematics, solved by using natural neighbor Galerkin methods within a nonincremental space–time framework that can be generalized for addressing parametric models. Examples showing the performance and potentialities of the proposed methodology are included. Copyright

KEY WORDS: model reduction; proper generalized decomposition; large geometrical transformations; meshless methods; natural element method; parametric models

1. INTRODUCTION

1.1. On the difficulty of simulating evolving domains

Evolving domains introduce many numerical difficulties. Firstly, when a fixed mesh is used, the domain evolution must be captured by using an appropriate technique (e.g., volume of fluid and level sets). The resulting advection terms must be efficiently stabilized by using, in turn, an adequate technique (SUPG, DG, etc.). Numerous works have addressed such questions during the last decades (see, for instance, [1] and references therein).

Another possibility consists in tracking the domain whose geometry evolves with the material velocity, in a (updated) Lagrangian approach. This approach simplifies the treatment of advection terms that now result in a simple material derivative. The main drawback is, however, that meshes become rapidly too distorted, implying the need for frequent remeshing and the associated field projection between old and new meshes. A particularly elegant analysis of the difficulties associated with this approach to the problem can be found in [2]. Intermediate procedures have been proposed in the framework of ALE methods, alleviating partially the issues of fixed and moving meshes [1]. However, the determination of the optimal velocity of the mesh is a tricky problem.

Some years ago, new discretization techniques, whose accuracy proved to be independent of the nodal distribution used to approximate the different fields involved in the models, were proposed. These techniques were called meshless or meshfree methods, even if some of them employ a background mesh to construct the functional approximation or even to perform numerical integration.

*Correspondence to: Elías Cueto, Aragon Institute of Engineering Research, Betancourt Building, María de Luna, s.n. 50018 Zaragoza, Spain.

†E-mail: ecueto@unizar.es

This designation is justified by the fact that the approximation accuracy does not depend on the relative position of the nodes. As a result, remeshing can be avoided even in the case of large distortions of the background mesh.

Despite the chosen framework for the description of the kinematics, it is well known in the reduced-order modeling community that the determination of an efficient set of reduced basis for problems defined in an evolving domain is a difficult task. This is caused, no doubt, because the deformation of the domain very much complicates the concept of *snapshot*, which is crucial in understanding the overall behavior of the system, and in determining the set of reduced basis itself [3].

In this work, among many different possibilities, we have chosen the natural element method (NEM), widely described in [4] and references therein, to approximate the kinematics in an updated Lagrangian framework. NEM overcomes FEM remeshing needs by employing natural neighbor approximation instead of piecewise polynomials to construct shape functions in a Galerkin setting. Thus, within the NEM framework, one can proceed with the original cloud of nodes moving according to material velocity during the whole simulation, even in the case of very large geometrical transformations. This is not a crucial choice in the development that follows (many other meshless methods that allow for an updated Lagrangian description of kinematics exist), albeit NEM presents some very interesting characteristics that will be analyzed in the succeeding discussions [5].

In what follows, we consider a model defined in a domain that at time $t = 0$ occupies the region $\Omega^0 \subset \mathbb{R}^3$. The different fields in this domain are approximated from a cloud of N_n nodes located at positions $\tilde{\mathbf{x}}_i^0$, $i = 1, \ldots, N_n$. The material domain evolves in time, $\Omega(t)$ representing its configuration at time t. We assume that this evolution is defined by a given, decoupled velocity field $\mathbf{v}(\mathbf{x} \in \Omega(t), t \in \mathcal{I} \subset \mathbb{R}_+)$. Nodes move with the material velocity, and because of the meshless behavior of the NEM approximation, all the fields are approximated in the updated domain $\Omega(t)$ by using the original cloud of nodes. No addition or deletion of nodes is considered, even if it is perfectly possible in an NEM framework. At time t, nodal positions will be noted by $\tilde{\mathbf{x}}_i^t$, $i = 1, \ldots, N_n$.

Hereafter, we assume, without loss of generality, that the model, defined in the evolving domain $\Omega(t)$, involves the unknown field $u(\mathbf{x} \in \Omega(t), t \in \mathcal{I})$. We focus on the possibility of determining a reduced basis approximation for such field in the context of the proper generalized decomposition (PGD) framework [6–12]. In this work, a strategy able to compute transient solutions in evolving domains is proposed. This strategy falls within a nonincremental framework originally proposed in a different context by Ladeveze [13]. Moreover, it will be shown how efficiently parametric models defined in evolving domains can be solved. Here, the model parameter, say, the thermal conductivity k of the thermal model here addressed, could be introduced as an extra-coordinate in the model, and then a multidimensional representation of the unknown field $u(\mathbf{x} \in \Omega(t), t \in \mathcal{I}, k \in \Im)$ will be found.

1.2. Reduced-order modeling of parametric models: the case of fixed domains

In this section, as an introduction, we summarize the PGD-based model reduction strategy for fixed domains. Let us consider the following parametric heat transfer equation:

$$\frac{\partial u}{\partial t} - k\Delta u - s = 0, \tag{1}$$

with homogeneous initial and boundary conditions. Enforcement of nonhomogeneous initial and boundary conditions was deeply analyzed in [14].

Here, $u = u(\mathbf{x}, t, k) \in \Omega \times I \times \Im$, and the source term s is assumed constant for simplicity. In the PGD framework, the conductivity k is viewed as a new coordinate defined in the interval $\Im$, rather than as a parameter. Thus, instead of solving the thermal model for different, discrete values of the conductivity parameter, the strategy developed in [15] and also in [16] aims at solving at once a more general, multidimensional problem. The price to pay is precisely an increase of the problem dimensionality. However, because the complexity of the PGD technique scales only linearly (and not exponentially) with the space dimension, considering the conductivity as a new coordinate still

allows to efficiently obtain an accurate solution. We review here, precisely, the PGD approach to standard parametric problems.

The weak form related to Equation (1) reads as follows: *Find $u(\mathbf{x}, t, k)$ such that*

$$\int_{\Omega \times I \times \Im} u^* \cdot \left(\frac{\partial u}{\partial t} - k \cdot \Delta u - s \right) \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t \cdot \mathrm{d}k \; = \; 0, \tag{2}$$

for all test functions u^* selected in an appropriate functional space.

The PGD solution is sought iteratively in the following form [6]:

$$u(\mathbf{x}, t, k) \approx \sum_{i=1}^{N} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k). \tag{3}$$

Let us assume that the nth term of the PGD approximation is already known:

$$u^n(\mathbf{x}, t, k) = \sum_{i=1}^{n} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k). \tag{4}$$

Computation of the $(n+1)$th term $X_{n+1}(\mathbf{x}) \cdot T_{n+1}(t) \cdot K_{n+1}(k)$, which we write as $R(\mathbf{x}) \cdot S(t) \cdot W(k)$ for simplicity,

$$u^{n+1} = u^n + R(\mathbf{x}) \cdot S(t) \cdot W(k), \tag{5}$$

begins by assuming the simplest choice for the test functions u^* used in Equation (2):

$$u^* = R^*(\mathbf{x}) \cdot S(t) \cdot W(k) + R(\mathbf{x}) \cdot S^*(t) \cdot W(k) + R(\mathbf{x}) \cdot S(t) \cdot W^*(k). \tag{6}$$

With the trial and test functions given by Equations (5) and (6), respectively, Equation (2) is a nonlinear problem that must be solved by means of a suitable iterative scheme. In our earlier papers [6] and [17], we used Newton's method. Simpler linearization strategies can also be applied, however. The simplest one is an alternating direction, fixed-point algorithm, which was found remarkably robust in the present context. Each iteration consists of three steps that are repeated until convergence, that is, until reaching the fixed point. The first step assumes $S(t)$ and $W(k)$ known from the previous iteration and computes an update for $R(\mathbf{x})$ (in this case, the test function reduces to $R^*(\mathbf{x}) \cdot S(t) \cdot W(k)$). From the just updated $R(\mathbf{x})$ and the previously used $W(k)$, we can update $S(t)$ (with $u^* = R(\mathbf{x}) \cdot S^*(t) \cdot W(k)$). Finally, from the just computed $R(\mathbf{x})$ and $S(t)$, we update $W(k)$ (with $u^* = R(\mathbf{x}) \cdot S(t) \cdot W^*(k)$). This iterative procedure continues until reaching convergence. The converged functions $R(\mathbf{x})$, $S(t)$, and $W(k)$ yield the new functional product of the current enrichment step: $X_{n+1}(\mathbf{x}) = R(\mathbf{x})$, $T_{n+1}(t) = S(t)$, and $K_{n+1}(k) = W(k)$. The explicit form of these operations is described as follows.

1.2.1. Computing $R(\mathbf{x})$ from $S(t)$ and $W(k)$.

We consider the weak form of Equation (1):

$$\int_{\Omega \times I \times \Im} u^* \cdot \left(\frac{\partial u}{\partial t} - k \cdot \Delta u - s \right) \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t \cdot \mathrm{d}k = 0. \tag{7}$$

Here, the trial function is given by

$$u(\mathbf{x}, t, k) = \sum_{i=1}^{n} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k) + R(\mathbf{x}) \cdot S(t) \cdot W(k). \tag{8}$$

Because S and W are known from the previous iteration, the test function reads

$$u^*(\mathbf{x}, t, k) = R^*(\mathbf{x}) \cdot S(t) \cdot W(k). \tag{9}$$

Introducing Equations (8) and (9) into Equation (7) yields

$$\int_{\Omega \times I \times \Im} R^* \cdot S \cdot W \cdot \left(R \cdot \frac{\partial S}{\partial t} \cdot W - k \cdot \Delta R \cdot S \cdot W \right) dx\, dt\, dk = - \int_{\Omega \times I \times \Im} R^* \cdot S \cdot W \cdot \mathcal{R}^n\, dx\, dt\, dk,$$

(10)

where $\mathcal{R}^n$ stands for the residual at enrichment step n:

$$\mathcal{R}^n = \sum_{i=1}^{n} X_i \cdot \frac{\partial T_i}{\partial t} \cdot K_i - \sum_{i=1}^{n} k \cdot \Delta X_i \cdot T_i \cdot K_i - s.$$

(11)

Because all functions depending time and conductivity have been already determined, we can integrate Equation (10) over $I \times \Im$. With the following notations,

$$\begin{bmatrix} w_1 = \int_{\Im} W^2\, dk & s_1 = \int_I S^2\, dt & r_1 = \int_{\Omega} R^2\, dx \\ w_2 = \int_{\Im} k W^2\, dk & s_2 = \int_I S \cdot \frac{dS}{dt}\, dt & r_2 = \int_{\Omega} R \cdot \Delta R\, dx \\ w_3 = \int_{\Im} W\, dk & s_3 = \int_I S\, dt & r_3 = \int_{\Omega} R\, dx \\ w_4^i = \int_{\Im} W \cdot K_i\, dk & s_4^i = \int_I S \cdot \frac{dT_i}{dt}\, dt & r_4^i = \int_{\Omega} R \cdot \Delta X_i\, dx \\ w_5^i = \int_{\Im} k W \cdot K_i\, dk & s_5^i = \int_I S \cdot T_i\, dt & r_5^i = \int_{\Omega} R \cdot X_i\, dx \end{bmatrix},$$

(12)

Equation (10) reduces to

$$\int_{\Omega} R^* \cdot (w_1 \cdot s_2 \cdot R - w_2 \cdot s_1 \cdot \Delta R)\, dx = - \int_{\Omega} R^* \cdot \left(\sum_{i=1}^{n} w_4^i \cdot s_4^i \cdot X_i - \sum_{i=1}^{n} w_5^i \cdot s_5^i \cdot \Delta X_i - w_3 \cdot s_3 \cdot s \right) dx.$$

(13)

Equation (13) defines in weak form an elliptic steady-state boundary value problem for the unknown function R that can be solved by using any suitable discretization technique (finite elements, finite volumes, etc.). Another possibility consists in coming back to the strong form of Equation (13),

$$w_1 \cdot s_2 \cdot R - w_2 \cdot s_1 \cdot \Delta R = - \left(\sum_{i=1}^{n} w_4^i \cdot s_4^i \cdot X_i - \sum_{i=1}^{n} w_5^i \cdot s_5^i \cdot \Delta X_i - w_3 \cdot s_3 \cdot s \right),$$

(14)

that can be solved by using any classical collocation technique (finite differences, SPH, etc.).

1.2.2. Computing $S(t)$ from $R(x)$ and $W(k)$. In the present case, the test function is written as

$$u^* (x, t, k) = S^*(t) \cdot R(x) \cdot W(k),$$

(15)

and the weak form becomes

$$\int_{\Omega \times I \times \Im} S^* \cdot R \cdot W \cdot \left(R \cdot \frac{\partial S}{\partial t} \cdot W - k \cdot \Delta R \cdot S \cdot W \right) dx\, dt\, dk = - \int_{\Omega \times I \times \Im} S^* \cdot R \cdot W \cdot \mathcal{R}^n\, dx\, dt\, dk.$$

(16)

Integrating over $\Omega \times \Im$ gives

$$\int_I S^* \cdot \left(w_1 \cdot r_1 \cdot \frac{dS}{dt} - w_2 \cdot r_2 \cdot S \right) dt = - \int_I S^* \cdot \left(\sum_{i=1}^{n} w_4^i \cdot r_5^i \cdot \frac{dT_i}{dt} - \sum_{i=1}^{n} w_5^i \cdot r_4^i \cdot T_i - w_3 \cdot r_3 \cdot s \right) dt.$$

(17)

Equation (17) represents the weak form of the ODE defining the time evolution of the field S that can be solved by using any stabilized discretization technique (Streamline upwind, DG, etc.). The strong form of Equation (17) reads

$$w_1 \cdot r_1 \cdot \frac{\mathrm{d}S}{\mathrm{d}t} - w_2 \cdot r_2 \cdot S = -\left(\sum_{i=1}^{n} w_4^i \cdot r_5^i \cdot \frac{\mathrm{d}T_i}{\mathrm{d}t} - \sum_{i=1}^{n} w_5^i \cdot r_4^i \cdot T_i - w_3 \cdot r_3 \cdot s \right). \qquad (18)$$

Equation (18) can be solved by using backward finite differences, or higher-order Runge–Kutta schemes, among many other possibilities.

1.2.3. Computing $W(k)$ from $R(x)$ and $S(t)$. The test function is now given by

$$u^* (\mathbf{x}, t, k) = W^*(k) \cdot R(\mathbf{x}) \cdot S(t), \qquad (19)$$

and the weak form becomes

$$\int_{\Omega \times I \times \Im} W^* \cdot R \cdot S \cdot \left(R \cdot \frac{\partial S}{\partial t} \cdot W - k \cdot \Delta R \cdot S \cdot W \right) \mathrm{d}\mathbf{x} \, \mathrm{d}t \, \mathrm{d}k = - \int_{\Omega \times I \times \Im} W^* \cdot R \cdot S \cdot \mathcal{R}^n \, \mathrm{d}\mathbf{x} \, \mathrm{d}t \, \mathrm{d}k. \qquad (20)$$

Integration over $\Omega \times I$ yields

$$\int_{\Im} W^* \cdot (r_1 \cdot s_2 \cdot W - r_2 \cdot s_1 \cdot k \cdot W) \, \mathrm{d}k$$

$$= - \int_{\Im} W^* \cdot \left(\sum_{i=1}^{n} r_5^i \cdot s_4^i \cdot K_i - \sum_{i=1}^{n} r_4^i \cdot s_5^i \cdot k \cdot K_i - r_3 \cdot s_3 \cdot s \right) \mathrm{d}k. \qquad (21)$$

Equation (21) does not involve any differential operator. The corresponding strong form reads

$$(r_1 \cdot s_2 - r_2 \cdot s_1 \cdot k) \cdot W = - \left(\sum_{i=1}^{n} (r_5^i \cdot s_4^i - r_4^i \cdot s_5^i \cdot k) \cdot K_i - r_3 \cdot s_3 \cdot s \right). \qquad (22)$$

This is actually an algebraic problem, which is hardly a surprise because the original Equation (1) does not contain derivatives with respect to the parameter k. The introduction of parameter k as an additional model coordinate does not increase the cost of a particular enrichment step. It does, however, necessitate more enrichment steps, that is, more terms (higher N) in the decomposition (3).

Remark 1
The just described procedure assumes that Ω does not depend on time in order to decouple the space and time problems.

Remark 2
We have seen that at each enrichment step, the construction of the new functional product in Equation (3) requires nonlinear iterations. If m_i denotes the number of iterations needed at enrichment step i, the total number of iterations involved in the construction of the PGD approximation is $m = \sum_{i=1}^{i=N} m_i$. In the aforementioned example, the entire procedure thus involves the solution of m three-dimensional problems for the functions $X_i(\mathbf{x})$, m one-dimensional problems for the functions $T_i(t)$, and m algebraic systems for the functions $K_i(k)$. In general, m rarely exceeds 10. The number N of functional products needed to approximate the solution with enough accuracy depends on the solution regularity. All numerical experiments carried to date reveal that N ranges between a few tens and one hundred. Thus, we can conclude that the complexity of the PGD procedure to compute the approximation (3) is of some tens of 3D steady-state problems (the cost related to the 1D and algebraic problems being negligible with respect to the 3D problems). In a classical approach, one must solve for each particular value of parameter k a 3D problem at each time step. In usual applications, this often implies the computation of several millions of 3D solutions. Clearly, the CPU time savings by applying the PGD can be of several orders of magnitude.

2. SEPARATED REPRESENTATION OF MODELS DEFINED IN EVOLVING DOMAINS

In order to show how the just explained strategy can be extended to problems defined in evolving domains, we come back to a nonparametric problem, for the sake of simplicity in the exposition (the procedure ahead can straightforwardly be extended to parametric problems as in the previous section), and consider the advection–diffusion equation defined in a domain $\Omega(t)$ that evolves with a prescribed velocity field $\mathbf{v}(\mathbf{x}, t)$, $\mathbf{x} \in \Omega(t)$, and $t \in \mathcal{I}$. Without loss of generality, we assume homogeneous initial and boundary conditions. The issue related to the enforcement of nonhomogeneous boundary conditions was deeply addressed in [14].

The weak form of the problem, in this case, reads as follows: *Find* $u(\mathbf{x}, t)$ *such that*

$$\int_{\mathcal{I}} \int_{\Omega(t)} u^* \cdot \left(\frac{Du}{Dt} - k \cdot \Delta u - s \right) \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t = 0 \tag{23}$$

holds for every test function u^* *defined in an appropriate Hilbert space.* The source term is considered depending on the space and time coordinates, that is, $s(\mathbf{x}, t)$.

By integrating by parts, we write the weak form as

$$\int_{\mathcal{I}} \int_{\Omega(t)} \left(u^* \cdot \frac{Du}{Dt} + k \cdot \nabla u^* \cdot \nabla u - u^* \cdot s \right) \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t = 0. \tag{24}$$

The approximation of the field $u(\mathbf{x}, t)$ is constructed from nodal values $u_i^t \equiv u(\tilde{\mathbf{x}}_i^t, t)$ by utilizing a natural neighbor interpolation:

$$u(\mathbf{x} \in \Omega(t), t) \approx \sum_{i=1}^{i=N_n} N_i^t(\mathbf{x}) \cdot u_i^t = \mathbf{N}^t \cdot \mathbf{U}^t, \tag{25}$$

where the upper index t associated to the shape functions N_i^t indicates that these shape functions were defined from the nodal positions $\tilde{\mathbf{x}}_i^t$ in $\Omega(t)$.

As mentioned earlier, although it is by no means the only possible choice, natural neighbor interpolation has remarkable properties that make their use in this context very convenient [4]. Undoubtedly, one of them is the Kroenecker delta property that states

$$N_i^t(\tilde{\mathbf{x}}_j^t) = \delta_{ij}, \tag{26}$$

which, together with the exact interpolation on boundaries, makes it possible to easily enforce Dirichlet boundary conditions.

In what follows, we analyze separately the different terms in Equation (24).

2.1. Diffusive term

We consider the diffusive term in Equation (24):

$$\mathcal{D} = \int_{\mathcal{I}} \int_{\Omega(t)} k \cdot \nabla u^* \cdot \nabla u \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t. \tag{27}$$

If we define a matrix $\mathbf{B}^t$ containing the derivatives of the shape functions N_i^t as

$$\mathbf{B}^t = \begin{pmatrix} \frac{\mathrm{d}N_1^t}{\mathrm{d}x} & \frac{\mathrm{d}N_2^t}{\mathrm{d}x} & \cdots & \frac{\mathrm{d}N_n^t}{\mathrm{d}x} \\ \frac{\mathrm{d}N_1^t}{\mathrm{d}y} & \frac{\mathrm{d}N_2^t}{\mathrm{d}y} & \cdots & \frac{\mathrm{d}N_n^t}{\mathrm{d}y} \\ \frac{\mathrm{d}N_1^t}{\mathrm{d}z} & \frac{\mathrm{d}N_2^t}{\mathrm{d}z} & \cdots & \frac{\mathrm{d}N_n^t}{\mathrm{d}z} \end{pmatrix}, \tag{28}$$

the diffusive term can be written as

$$\mathcal{D} = \int_{\mathcal{I}} \int_{\Omega(t)} k \cdot \mathbf{U}^{*^{\mathrm{T}}} \cdot \mathbf{B}^{t^{\mathrm{T}}} \cdot \mathbf{B}^t \cdot \mathbf{U}^t \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t$$

$$= \int_{\mathcal{I}} k \cdot \mathbf{U}^{*^{\mathrm{T}}} \cdot \left(\int_{\Omega(t)} \mathbf{B}^{t^{\mathrm{T}}} \cdot \mathbf{B}^t \, \mathrm{d}\mathbf{x} \right) \cdot \mathbf{U}^t \, \mathrm{d}t = \int_{\mathcal{I}} k \cdot \mathbf{U}^{*^{\mathrm{T}}} \cdot \mathbf{G}(t) \cdot \mathbf{U}^t \, \mathrm{d}t \tag{29}$$

Because $\Omega(t)$ is known $\forall t$, we can evaluate the integral

$$G_k = \int_{\Omega(t_k)} \mathbf{B}^{t_k\,\mathrm{T}} \cdot \mathbf{B}^{t_k} \, \mathrm{d}\mathbf{x} \tag{30}$$

at different times t_k, $k = 1, \ldots, Q$.

Following the spirit of Karhunen–Loeve transform or proper orthogonal decompositions (see [18–20]) from these integrals, we could define a matrix $\mathbf{G}$,

$$\mathbf{G} = (\mathbf{G}_1 \, \mathbf{G}_2 \, \cdots \, \mathbf{G}_Q), \tag{31}$$

that, after applying an SVD, gives

$$\mathbf{G}(t) = \int_{\Omega(t)} \mathbf{B}^{t\,\mathrm{T}} \cdot \mathbf{B}^t \, \mathrm{d}\mathbf{x} \approx \sum_{j=1}^{j=m_1} F_j^d(t) \cdot \mathbf{E}_j^d, \tag{32}$$

with $m_1 < Q$ and $m_1 < N_n$.

Thus, the diffusive term can be advantageously written as

$$\mathcal{D} = \int_{\mathcal{I}} \int_{\Omega(t)} k \cdot \mathbf{U}^{*\,\mathrm{T}} \cdot \mathbf{B}^{t\,\mathrm{T}} \cdot \mathbf{B}^t \cdot \mathbf{U}^t \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t$$

$$= \int_{\mathcal{I}} k \cdot \mathbf{U}^{*\,\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_1} F_j^d(t) \cdot \mathbf{E}_j^d \right) \cdot \mathbf{U}^t \, \mathrm{d}t. \tag{33}$$

2.2. Advective term

We consider now the term involving time derivatives:

$$\mathcal{A} = \int_{\mathcal{I}} \int_{\Omega(t)} u^* \cdot \frac{\mathrm{D}u}{\mathrm{D}t} \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t. \tag{34}$$

The material derivative $\mathrm{D}u/\mathrm{D}t$ writes when using a fixed reference system as follows:

$$\frac{\mathrm{D}u}{\mathrm{D}t} = \frac{\partial u}{\partial t} + \mathbf{v} \cdot \nabla u. \tag{35}$$

However, when the reference system follows matter, the advective term can be discretized along the characteristic lines according to

$$\frac{\mathrm{D}u}{\mathrm{D}t} \approx \frac{u(\mathbf{x}, t) - \hat{u}(\mathbf{x}, t)}{\Delta t}, \tag{36}$$

where

$$\hat{u}(\mathbf{x}, t) = u(\mathbf{x} - \mathbf{v} \cdot \Delta t, t - \Delta t) \tag{37}$$

represents the root of the characteristic line at the previous time step, and hence the advantages of using an updated Lagrangian frame of reference.

Thus, if the time interval $\mathcal{I}$ is decomposed in P time steps of length Δt, that is, $\mathcal{I} = [0, P \cdot \Delta t]$, Equation (34) reduces to

$$\mathcal{A} \approx \sum_{p=1}^{p=P} \int_{\Omega(t_p)} u^* \cdot (u(\mathbf{x}, t_p) - \hat{u}(\mathbf{x}, t_p)) \, \mathrm{d}\mathbf{x}, \tag{38}$$

which is composed of two terms:

$$\mathcal{A}_1 \approx \sum_{p=1}^{p=P} \int_{\Omega(t_p)} u^* \cdot u(\mathbf{x}, t_p) \, \mathrm{d}\mathbf{x} \tag{39}$$

and

$$\mathcal{A}_2 \approx \sum_{p=1}^{p=P} \int_{\Omega(t_p)} u^* \cdot \hat{u}(\mathbf{x}, t_p) \, d\mathbf{x}. \tag{40}$$

Considering the approximation given by Equation (25), it results

$$\mathcal{A}_1 \approx \sum_{p=1}^{p=P} \mathbf{U}^{*\mathrm{T}} \cdot \left(\int_{\Omega(t_p)} \mathbf{N}^{t_p\mathrm{T}} \cdot \mathbf{N}^{t_p} \, d\mathbf{x} \right) \cdot \mathbf{U}^{t_p} = \sum_{p=1}^{p=P} \mathbf{U}^{*\mathrm{T}} \cdot \mathbf{M}(t_p) \cdot \mathbf{U}^{t_p}. \tag{41}$$

Because $\Omega(t)$ is known $\forall t$, we can easily evaluate the integral

$$\mathbf{M}_k = \int_{\Omega(t_k)} \mathbf{N}^{t_k\mathrm{T}} \cdot \mathbf{N}^{t_k} \, d\mathbf{x} \tag{42}$$

at different times t_k, $k = 1, \ldots, Q$.

Again, from these integrals, we can define a matrix $\mathbf{M}$,

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}_1 \ \mathbf{M}_2 \ \cdots \ \mathbf{M}_Q \end{pmatrix}, \tag{43}$$

that, after applying an SVD, allows writing

$$\mathbf{M}(t) = \int_{\Omega(t)} \mathbf{N}^{t\mathrm{T}} \cdot \mathbf{N}^t \, d\mathbf{x} \approx \sum_{j=1}^{j=m_2} F_j^a(t) \cdot \mathbf{E}_j^a, \tag{44}$$

with $m_2 < Q$ and $m_2 < N_n$.

Thus, the term $\mathcal{A}_1$ reads

$$\mathcal{A}_1 \approx \sum_{p=1}^{p=P} \mathbf{U}^{*\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t_p) \cdot \mathbf{E}_j^a \right) \cdot \mathbf{U}^{t_p}. \tag{45}$$

Now, we come back to the second contribution $\mathcal{A}_2$. Firstly, we define the approximation of $\hat{u}(\mathbf{x}, t)$,

$$\hat{u}(\mathbf{x} \in \Omega(t), t) \approx \sum_{i=1}^{i=N_n} \hat{N}_i(\mathbf{x}) \cdot u_i^{t-\Delta t} = \hat{\mathbf{N}} \cdot \mathbf{U}^{t-\Delta t}, \tag{46}$$

from which the integral $\mathcal{A}_2$ now reads

$$\mathcal{A}_2 \approx \sum_{p=1}^{p=P} \mathbf{U}^{*\mathrm{T}} \cdot \left(\int_{\Omega(t_p)} \mathbf{N}^{t_p\mathrm{T}} \cdot \hat{\mathbf{N}} \, d\mathbf{x} \right) \cdot \mathbf{U}^{t_p-\Delta t} = \sum_{p=1}^{p=P} \mathbf{U}^{*\mathrm{T}} \cdot \hat{\mathbf{M}}(t_p) \cdot \mathbf{U}^{t_p-\Delta t}. \tag{47}$$

Because $\Omega(t)$ is known $\forall t$, we can evaluate the integral

$$\hat{\mathbf{M}}_k = \int_{\Omega(t_k)} \mathbf{N}^{t_k\mathrm{T}} \cdot \hat{\mathbf{N}} \, d\mathbf{x} \tag{48}$$

at different times t_k, $k = 1, \ldots, Q$.

From these integrals, we could define a matrix $\hat{\mathbf{M}}$,

$$\hat{\mathbf{M}} = \left(\hat{\mathbf{M}}_1, \hat{\mathbf{M}}_2, \ldots, \hat{\mathbf{M}}_Q \right), \tag{49}$$

that, by applying an SVD, allows writing

$$\hat{\mathbf{M}}(t) = \int_{\Omega(t)} \mathbf{N}^{t\mathrm{T}} \cdot \hat{\mathbf{N}} \, d\mathbf{x} \approx \sum_{j=1}^{j=m_3} F_j^u(t) \cdot \mathbf{E}_j^u, \tag{50}$$

with $m_3 < Q$ and $m_3 < N_n$.

Thus, the term $\mathcal{A}_2$ reads

$$\mathcal{A}_2 \approx \sum_{p=1}^{p=P} \mathbf{U}^{*^\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_3} F_j^u(t_p) \cdot \mathbf{E}_j^u \right) \cdot \mathbf{U}^{t_p - \Delta t}. \tag{51}$$

2.3. Source term

We consider the source term in Equation (24):

$$\mathcal{S} = \int_{\mathcal{I}} \int_{\Omega(t)} u^* \cdot s(\mathbf{x}, t) \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t. \tag{52}$$

By approximating the source term from

$$s(\mathbf{x} \in \Omega(t), t) \approx \sum_{i=1}^{i=N_n} N_i^t(\mathbf{x}) \cdot s_i^t = \mathbf{N}^t \cdot \mathbf{S}^t, \tag{53}$$

we can write the source term in the weak form of the problem as

$$\begin{aligned}
\mathcal{S} &= \int_{\mathcal{I}} \int_{\Omega(t)} \mathbf{U}^{*^\mathrm{T}} \cdot \mathbf{N}^{t^\mathrm{T}} \cdot \mathbf{N}^t \cdot \mathbf{S}^t \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t \\
&= \int_{\mathcal{I}} \mathbf{U}^{*^\mathrm{T}} \cdot \left(\int_{\Omega(t)} \mathbf{N}^{t^\mathrm{T}} \cdot \mathbf{N}^t \, \mathrm{d}\mathbf{x} \right) \cdot \mathbf{S}^t \, \mathrm{d}t = \int_{\mathcal{I}} \mathbf{U}^{*^\mathrm{T}} \cdot \mathbf{M}(t) \cdot \mathbf{S}^t \, \mathrm{d}t,
\end{aligned} \tag{54}$$

which, considering the previous developments, results in

$$\begin{aligned}
\mathcal{S} &= \int_{\mathcal{I}} \int_{\Omega(t)} \mathbf{U}^{*^\mathrm{T}} \cdot \mathbf{N}^{t^\mathrm{T}} \cdot \mathbf{N}^t \cdot \mathbf{S}^t \, \mathrm{d}\mathbf{x} \cdot \mathrm{d}t \\
&= \int_{\mathcal{I}} \mathbf{U}^{*^\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t) \cdot \mathbf{E}_j^a \right) \cdot \mathbf{S}^t \, \mathrm{d}t.
\end{aligned} \tag{55}$$

By applying again an SVD, we can express $\mathbf{S}^t$ in a separated form as

$$\mathbf{S}^t \approx \sum_{l=1}^{l=L} C_l(t) \cdot \mathbf{D}_l, \tag{56}$$

leading to

$$\mathcal{S} = \int_{\mathcal{I}} \mathbf{U}^{*^\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t) \cdot \mathbf{E}_j^a \right) \cdot \left(\sum_{l=1}^{l=L} C_l(t) \cdot \mathbf{D}_l \right) \, \mathrm{d}t. \tag{57}$$

3. BUILDING UP THE SEPARATED REPRESENTATION OF THE MODEL SOLUTION

Once the problem has been stated in a separated form, with the SVD applied to every term in its weak form, the technique here proposed proceeds by constructing a separated, space–time, representation for the solution, $u = u(\mathbf{x}, t)$. In the mentioned separated representation, the model reads

$$\begin{aligned}
&\sum_{p=1}^{k=P} \mathbf{U}^{*^\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t_p) \cdot \mathbf{E}_j^a \right) \cdot \mathbf{U}^{t_p} - \sum_{p=1}^{p=P} \mathbf{U}^{*^\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_3} F_j^u(t_p) \cdot \mathbf{E}_j^u \right) \cdot \mathbf{U}^{t_p - \Delta t} \\
&+ \int_{\mathcal{I}} k \cdot \mathbf{U}^{*^\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_1} F_j^d(t) \cdot \mathbf{E}_j^d \right) \cdot \mathbf{U}^t \, \mathrm{d}t - \int_{\mathcal{I}} \mathbf{U}^{*^\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t) \cdot \mathbf{E}_j^a \right) \cdot \mathbf{S}^t \, \mathrm{d}t = 0. \tag{58}
\end{aligned}$$

Assuming that the model solution accepts a separated space–time representation, one could look for an a priori separated representation of $\mathbf{U}^t$:

$$\mathbf{U}^t \approx \sum_{i=1}^{i=N} T_i(t) \cdot \mathbf{X}_i. \tag{59}$$

For constructing such an approximation, we proceed by computing a term of the finite sum at each iteration. Thus, we assume at iteration n that the n first terms of the sum have been already calculated, from which we can write the nth-order approximation of $\mathbf{U}^t$ as

$$\mathbf{U}^t \approx \sum_{i=1}^{i=n} T_i(t) \cdot \mathbf{X}_i. \tag{60}$$

At the following iteration, $n + 1$, we look for the new functional product $T_{n+1}(t) \cdot \mathbf{X}_{n+1}$. For the sake of simplicity, functions $T_{n+1}(t)$ and $\mathbf{X}_{n+1}$ will be noted by Υ and $\mathbf{R}$, respectively, where the dependence on t of Υ is omitted for the sake of clarity.

Thus, the $(n + 1)$th-order approximation reads

$$\mathbf{U}^t \approx \sum_{i=1}^{i=n} T_i(t) \cdot \mathbf{X}_i + \Upsilon \cdot \mathbf{R}. \tag{61}$$

To compute both functions Υ and $\mathbf{R}$, we consider Equation (58), where the trial function is given by Equation (61) and the test function by

$$\mathbf{U}^* = \Upsilon^* \cdot \mathbf{R} + \Upsilon \cdot \mathbf{R}^*. \tag{62}$$

Because the resulting problem is nonlinear, because of the product of both unknown functions Υ and $\mathbf{R}$, a linearization is therefore compulsory. The simplest one consists, as explained before, of a fixed-point, alternating directions, strategy that computes $\mathbf{R}$ by assuming known Υ, then Υ from the just updated $\mathbf{R}$. Both steps are repeated until reaching the fixed point of both Υ and $\mathbf{R}$.

In what follows, we detail both steps.

3.1. Computing the space function $\mathbf{R}$

When Υ is assumed known, the test function $\mathbf{U}^*$ reduces to $\mathbf{U}^* = \Upsilon \cdot \mathbf{R}^*$. In this case, the integral form writes

$$\sum_{p=1}^{p=P} \mathbf{R}^{*^{\mathrm{T}}} \cdot \Upsilon(t_p) \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t_p) \cdot \mathbf{E}_j^a \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_p) \cdot \mathbf{X}_i + \Upsilon(t_p) \cdot \mathbf{R} \right)$$

$$- \sum_{p=1}^{p=P} \mathbf{R}^{*^{\mathrm{T}}} \cdot \Upsilon(t_p) \cdot \left(\sum_{j=1}^{j=m_3} F_j^u(t_p) \cdot \mathbf{E}_j^u \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_{p-1}) \cdot \mathbf{X}_i + \Upsilon(t_{p-1}) \cdot \mathbf{R} \right)$$

$$+ \int_{\mathcal{I}} \mathbf{R}^{*^{\mathrm{T}}} \cdot k \cdot \Upsilon(t) \cdot \left(\sum_{j=1}^{j=m_1} F_j^d(t) \cdot \mathbf{E}_j^d \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t) \cdot \mathbf{X}_i + \Upsilon(t) \cdot \mathbf{R} \right) \, \mathrm{d}t$$

$$- \int_{\mathcal{I}} \mathbf{R}^{*^{\mathrm{T}}} \cdot \Upsilon(t) \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t) \cdot \mathbf{E}_j^a \right) \cdot \left(\sum_{l=1}^{l=L} C_l(t) \cdot \mathbf{D}_l \right) \, \mathrm{d}t = 0, \tag{63}$$

where $t_{p-1} = t_p - \Delta t$.

With a simple numerical quadrature used, the previous equation becomes

$$
\sum_{p=1}^{p=P} \mathbf{R}^{*\mathrm{T}} \cdot \Upsilon(t_p) \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t_p) \cdot \mathbf{E}_j^a \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_p) \cdot \mathbf{X}_i + \Upsilon(t_p) \cdot \mathbf{R} \right)
$$

$$
- \sum_{p=1}^{p=P} \mathbf{R}^{*\mathrm{T}} \cdot \Upsilon(t_p) \cdot \left(\sum_{j=1}^{j=m_3} F_j^u(t_p) \cdot \mathbf{E}_j^u \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_{p-1}) \cdot \mathbf{X}_i + \Upsilon(t_{p-1}) \cdot \mathbf{R} \right)
$$

$$
+ \sum_{p=1}^{p=P} \mathbf{R}^{*\mathrm{T}} \cdot k \cdot \Upsilon(t_p) \cdot \left(\sum_{j=1}^{j=m_1} F_j^d(t_p) \cdot \mathbf{E}_j^d \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_p) \cdot \mathbf{X}_i + \Upsilon(t_p) \cdot \mathbf{R} \right) \cdot \Delta t
$$

$$
- \sum_{p=1}^{p=P} \mathbf{R}^{*\mathrm{T}} \cdot \Upsilon(t_p) \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t_p) \cdot \mathbf{E}_j^a \right) \cdot \left(\sum_{l=1}^{l=L} C_l(t_p) \cdot \mathbf{D}_l \right) \cdot \Delta t = 0. \tag{64}
$$

Because of the arbitrariness of $\mathbf{R}^*$, after developing all the calculations, Equation (64) results in a linear system:

$$
\mathbf{H} \cdot \mathbf{R} = \mathbf{Z} \tag{65}
$$

from which we can update vector $\mathbf{R}$.

3.2. Computing the time function $\Upsilon(t)$

When $\mathbf{R}$ is assumed known, the test function $\mathbf{U}^*$ reduces to $\mathbf{U}^* = \Upsilon^* \cdot \mathbf{R}$. In this case, it is easy to verify that the discrete form reads

$$
\sum_{p=1}^{p=P} \Upsilon^*(t_p) \cdot \mathbf{R}^{\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t_p) \cdot \mathbf{E}_j^a \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_p) \cdot \mathbf{X}_i + \Upsilon(t_p) \cdot \mathbf{R} \right)
$$

$$
- \sum_{p=1}^{p=P} \Upsilon^*(t_p) \cdot \mathbf{R}^{\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_3} F_j^u(t_p) \cdot \mathbf{E}_j^u \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_{p-1}) \cdot \mathbf{X}_i + \Upsilon(t_{p-1}) \cdot \mathbf{R} \right)
$$

$$
+ \sum_{p=1}^{p=P} \Upsilon^*(t_p) \cdot \mathbf{R}^{\mathrm{T}} \cdot k \cdot \left(\sum_{j=1}^{j=m_1} F_j^d(t_p) \cdot \mathbf{E}_j^d \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_p) \cdot \mathbf{X}_i + \Upsilon(t_p) \cdot \mathbf{R} \right) \cdot \Delta t
$$

$$
- \sum_{p=1}^{p=P} \Upsilon^*(t_p) \cdot \mathbf{R}^{\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t_p) \cdot \mathbf{E}_j^a \right) \cdot \left(\sum_{l=1}^{l=L} C_l(t_p) \cdot \mathbf{D}_l \right) \cdot \Delta t = 0. \tag{66}
$$

Because we are assuming homogeneous initial condition, this results in $\Upsilon(t_0) = 0$.

Then, the arbitrariness of $\Upsilon^*(t_p)$, $\forall p \geqslant 1$, implies

$$
\mathbf{R}^{\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t_p) \cdot \mathbf{E}_j^a \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_p) \cdot \mathbf{X}_i + \Upsilon(t_p) \cdot \mathbf{R} \right)
$$

$$
- \mathbf{R}^{\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_3} F_j^u(t_p) \cdot \mathbf{E}_j^u \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_{p-1}) \cdot \mathbf{X}_i + \Upsilon(t_{p-1}) \cdot \mathbf{R} \right)
$$

$$+ \mathbf{R}^{\mathrm{T}} \cdot k \cdot \left(\sum_{j=1}^{j=m_1} F_j^d(t_p) \cdot \mathbf{E}_j^d \right) \cdot \left(\sum_{i=1}^{i=n} T_i(t_p) \cdot \mathbf{X}_i + \Upsilon(t_p) \cdot \mathbf{R} \right) \cdot \Delta t$$

$$- \mathbf{R}^{\mathrm{T}} \cdot \left(\sum_{j=1}^{j=m_2} F_j^a(t_p) \cdot \mathbf{E}_j^a \right) \cdot \left(\sum_{l=1}^{l=L} C_l(t_p) \cdot \mathbf{D}_l \right) \cdot \Delta t = 0, \tag{67}$$

which, after making the indicated calculation, results in a simple linear equation for each t_p:

$$\Upsilon(t_p) = a_p \cdot \Upsilon(t_{p-1}) + b_p, \quad \forall p \geqslant 1. \tag{68}$$

4. NUMERICAL TEST

In this section, we consider a numerical example consisting of a solid workpiece occupying at $t = 0$ the domain $\Omega^0 = (-0.5, 0.5) \times (0, 5)$. The piece is compressed from its upper face. The domain evolves consequently to take intermediate configurations $\Omega(t)$ until reaching its final geometry at time $t = 1.28$. The tool of unit length $(-0.5, 0.5)$ compresses the workpiece at a constant compression velocity. Assuming known the geometry evolution $\Omega(t)$, we solve the thermal model defined in $\Omega(t)$ in a nonincremental way. We consider the following initial and boundary conditions:

$$\begin{cases} u(\mathbf{x} \in \Omega^0, t = 0) = 1, \\ u(\mathbf{x} \in \Gamma_c(t), t) = 0, \\ \nabla u(\mathbf{x} \in \Gamma_f(t), t) \cdot \mathbf{n} = 0, \end{cases} \tag{69}$$

where $\Gamma_c(t)$ and $\Gamma_f(t)$ represent the parts of the boundary of $\Omega(t)$, $\Gamma(t) \equiv \partial \Omega(t)$, in contact with the tool or the work plane $y = 0$ and the free boundary, respectively.

After the strategy described in the previous section was applied, $N = 15$ modes were found to be enough for representing the whole thermal history $u(\mathbf{x} \in \Omega(t), t)$:

$$\mathbf{U}^t \approx \sum_{i=1}^{i=15} T_i(t) \cdot \mathbf{X}_i. \tag{70}$$

Functions T_i were computed only at times t_p, and $\mathbf{X}_i$ consists of a vector containing the nodal values related to any nodal distribution $\tilde{\mathbf{x}}_j^t$ in $\Omega(t)$.

For the sake of clarity, we defined functions $G_i(t)$ by interpolating $T_i(t_p)$ values and defined functions $F_i(\mathbf{x} \in \Omega^0)$ by interpolating values in $\mathbf{X}_i$ on the initial configuration Ω^0. Figure 1 depicts the five most significant space modes F_i, $i = 1, \ldots, 5$; as well as $G_i(t)$, $i = 1, \ldots, 15$.

Now, we are in the position to assign vectors $\mathbf{X}_i$ to nodes $\tilde{\mathbf{x}}_j^t$ related to the configuration $\Omega(t)$ and then to reconstruct the solution in $\Omega(t)$. Figure 2 depicts the reconstructed temperature field in $\Omega(t)$ for six different time instants.

Figure 3 depicts, in turn, the error of the approximation for different numbers of terms in the approximation (ranging from 10 to 30) versus the incremental, standard, finite element solution of the problem.

5. TOWARDS PARAMETRIC MODELING IN EVOLVING DOMAINS

The extension of the previously introduced technique to the case of parametric models on evolving domains is straightforward. To this end, we should come back to Section 3 and consider the parametric dependency of u on k, looking for the separated representation:

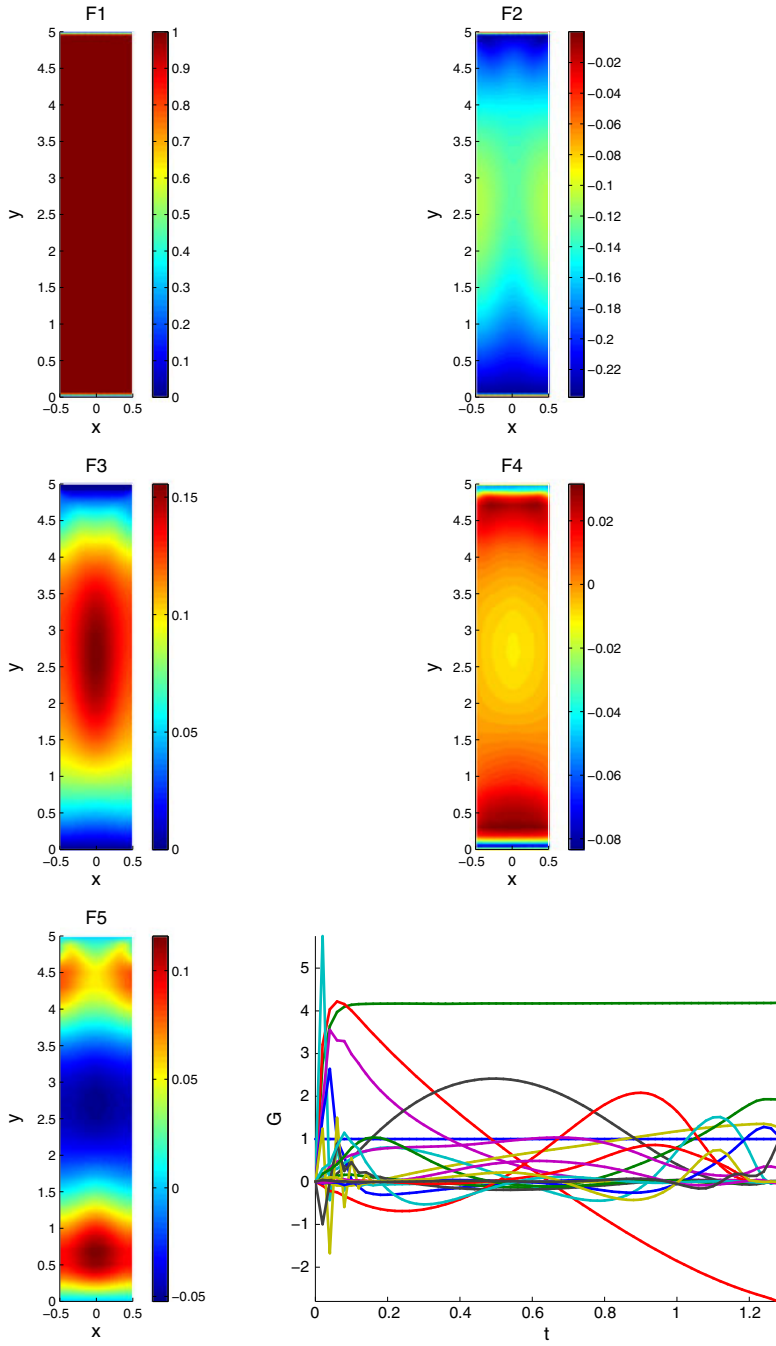$$\mathbf{U}_k^t \approx \sum_{i=1}^{i=N} T_i(t) \cdot K_i(k) \cdot \mathbf{X}_i. \tag{71}$$

Figure 1. Space and time functions involved in the separated representation of $u(\mathbf{x} \in \Omega(t), t)$.

For constructing such an approximation, we proceed by computing a term of the finite sum at each iteration. Thus, we assume at iteration n that the n first terms of the sum have been already calculated, from which we can write the n-order approximation of $\mathbf{U}_k^t$:

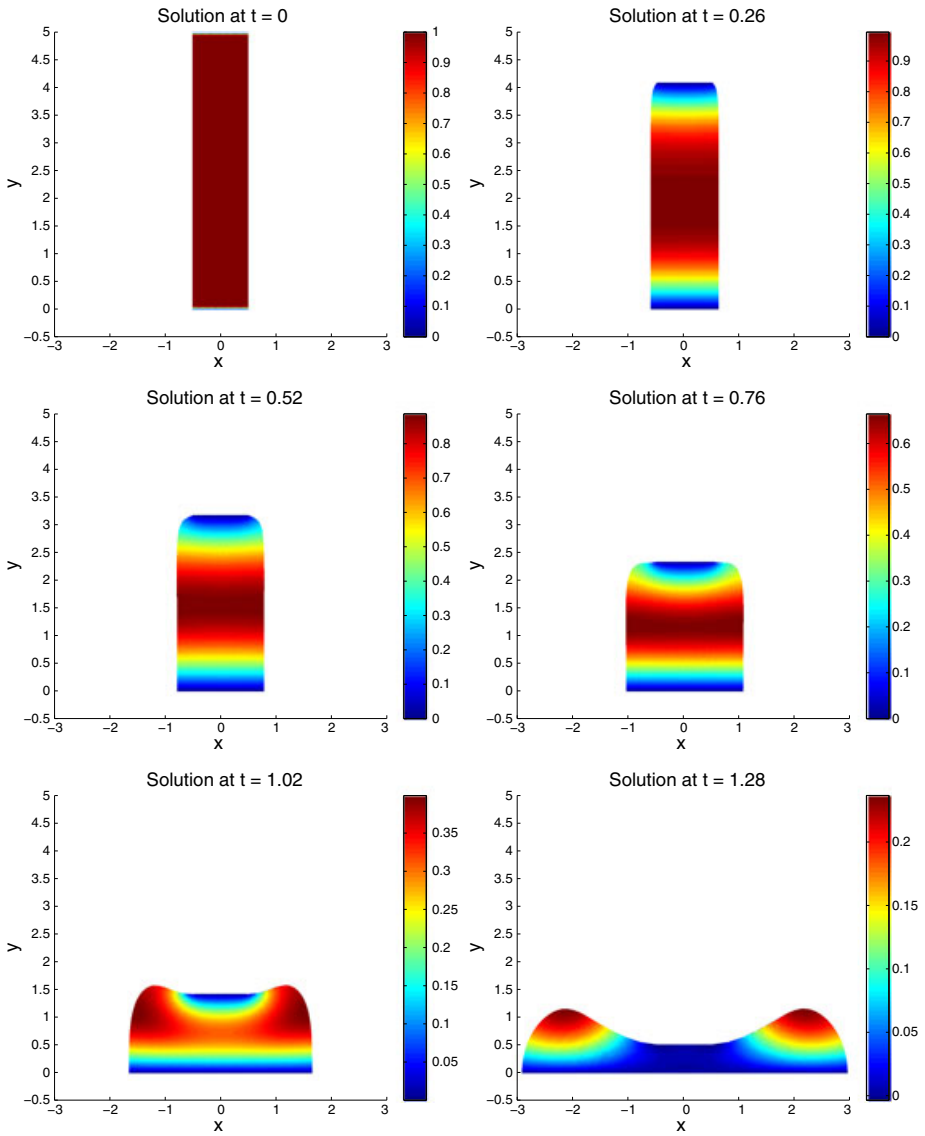$$\mathbf{U}_k^t \approx \sum_{i=1}^{i=n} T_i(t) \cdot K_i(k) \cdot \mathbf{X}_i. \tag{72}$$

Figure 2. Reconstructed temperature field $u(\mathbf{x} \in \Omega(t), t)$ at six different instants.
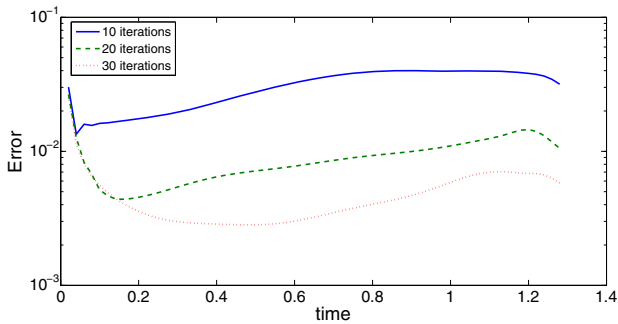

Figure 3. Error in the approximation for different numbers of terms in the sum. A standard, incremental, finite element solution has been taken as reference.

Now, at the next iteration $n + 1$, we look for the new functional product $T_{n+1}(t) \cdot K_{n+1} \cdot \mathbf{X}_{n+1}$. For the sake of simplicity, functions $T_{n+1}(t)$, K_{n+1}, and $\mathbf{X}_{n+1}$ will be noted by Υ, W, and $\mathbf{R}$, respectively, where the dependence on t of Υ, and on k of W is omitted for the sake of clarity.

Thus, the $(n + 1)$th-order approximation reads

$$\mathbf{U}_k^t \approx \sum_{i=1}^{i=n} T_i(t) \cdot K_i(k) \cdot \mathbf{X}_i + \Upsilon \cdot W \cdot \mathbf{R}. \tag{73}$$

For computing functions Υ, W, and $\mathbf{R}$, we consider Equation (58), where the trial function is given by (73) and the test function by

$$\mathbf{U}^* = \Upsilon^* \cdot W \cdot \mathbf{R} + \Upsilon \cdot W^* \cdot \mathbf{R} + \Upsilon \cdot W \cdot \mathbf{R}^*. \tag{74}$$

Because the resulting problem is nonlinear, because of the product of the three unknown functions Υ, W, and $\mathbf{R}$, a linearization is compulsory. The simplest one consists of the fixed-point, alternating directions, strategy presented earlier. By generalizing the procedure widely described in Section 3, we can compute the parametric and nonincremental separated representation. In the next section, we consider the parametric solution of the problem solved in Section 4.

6. NUMERICAL TEST INVOLVING PARAMETRIC MODELING

In this section, we consider the problem analyzed in Section 4 where the material conductivity k is now considered as a model extra-coordinate taking values in the interval $k \in \Im = (0, 1)$. The strategy is now a mere combination of those applied for parametric problems in steady domains and those for standard problems in evolving domains.

Figure 4 depicts the four most significant space modes F_i, $i = 1, \ldots, 4$, where again F_i refers to the interpolation defined from values in $\mathbf{X}_i$ on the initial configuration Ω^0.
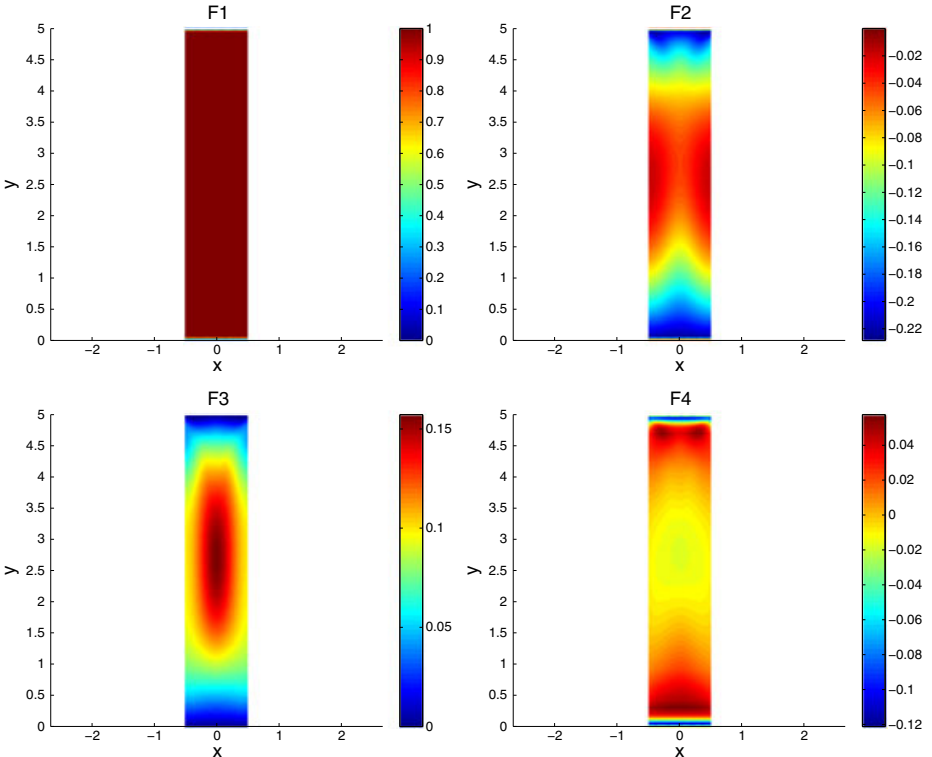

Figure 4. Four most significant space functions involved in the separated representation of $u(\mathbf{x} \in \Omega(t), t, k)$.
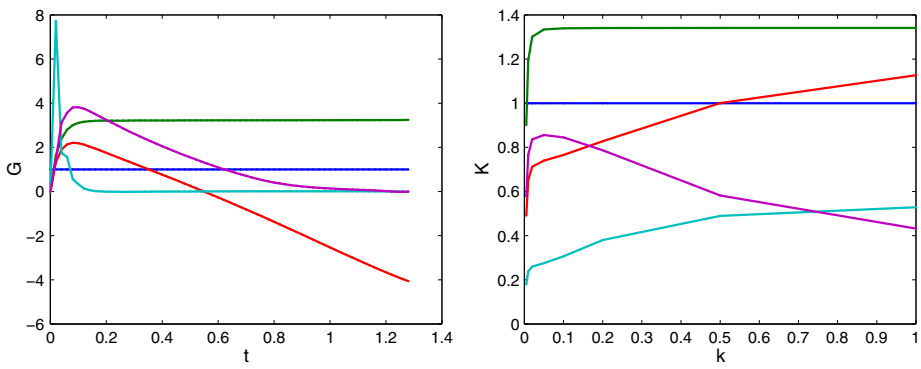
Figure 5. Most significant functions depending on the space and conductivity involved in the separated representation of $u(\mathbf{x} \in \Omega(t), t, k)$.
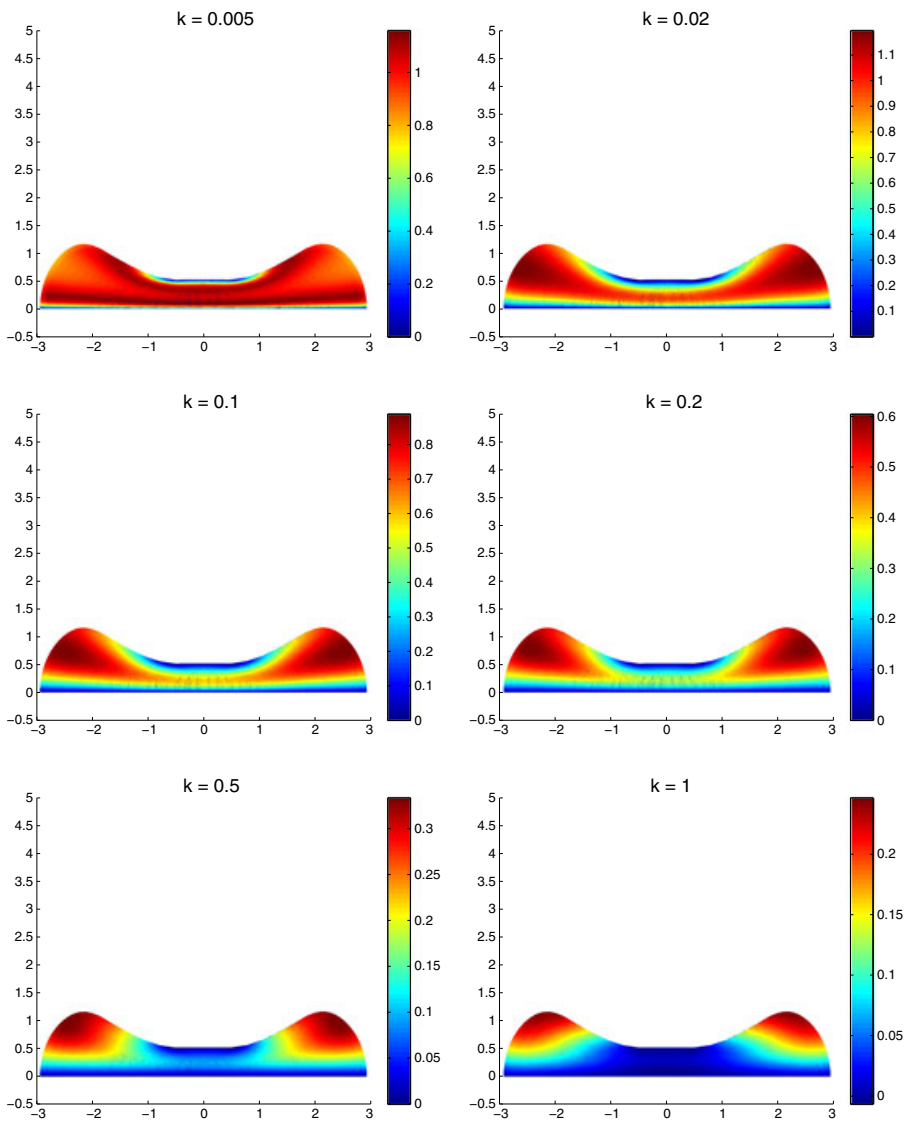

Figure 6. Reconstructed temperature field $u(\mathbf{x} \in \Omega(t = 1.28), t, k)$ for different values of the thermal conductivity k.

Figure 5 depicts the most significant functions depending on time and on conductivity.

Finally, Figure 6 depicts the temperature field reconstructed at the final geometry Ω ($t = 1.28$) for different values of the thermal conductivity. It can be noticed that the higher the conductivity, the faster is the cooling process induced by the lower and constant temperatures enforced on the tool and working plane surfaces in contact with the workpiece.

7. CONCLUSIONS

In this paper a novel strategy for a priori construction of reduced bases for problems defined in evolving domains is presented. The main challenge in this class of problems derives precisely from the deformation of the problem domain, which prevents the direct application of classical, a posteriori techniques such as proper orthogonal decomposition, to obtain appropriate reduced basis. The evolving nature of the domain obscures the concept of snapshot of the system state, requiring specific treatments.

However, it has been demonstrated that a combination of an updated Lagrangian approach for the description of domain's kinematics and a PGD-based obtention of the set of reduced basis in a separated space–time (possibly space–parameters–time) representation gives a very convenient way of constructing reduced basis. This basis can be advantageously employed to simulate complex problems at a very reduced CPU cost, as proven in the vast corps of literature devoted to this end.

REFERENCES

1. Donea J, Huerta A. *Finite Element Methods for Flow Problems*. John Wiley and sons: Chichester, West Sussex, England, 2003.
2. Lewis RW, Navti SE, Taylor C. A mixed Lagrangian–Eulerian approach to modelling fluid flow during mould filling. *International Journal for Numerical Methods in Fluids* 1997; **25**:931–952.
3. Ryckelynck D, Chinesta F, Cueto E, Ammar A. On a priori model reduction: overview and recent developments. *Archives of Computational Methods in Engineering* 2006; **12**(1):91–128.
4. Cueto E, Sukumar N, Calvo B, Martínez MA, Cegonino J, Doblare M. Overview and recent advances in natural neighbour Galerkin methods. *Archives of Computational Methods in Engineering* 2003; **10**(4):307–384.
5. Alfaro I, Yvonnet J, Chinesta F, Cueto E. A study on the performance of natural neighbour-based Galerkin methods. *International Journal for Numerical Methods in Engineering* 2007; **71**:1436–1465.
6. Ammar A, Mokdad B, Chinesta F, Keunings R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. *Journal of Non-Newtonian Fluid Mechanics* 2006; **139**:153–176.
7. Ladeveze P, Passieux JC, Neron D. The LATIN multiscale computational method and the Proper Generalized Decomposition. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(21-22, SI):1287–1296.
8. Chinesta F, Ammar A, Cueto E. Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Archives of Computational Methods in Engineering* 2010; **17**:327–350.
9. Chinesta F, Ladeveze P, Cueto E. A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering* 2011; **18**:395–404.
10. Ammar A, Chinesta F, Diez P, Huerta A. An error estimator for separated representations of highly multidimensional models. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(25-28):1872 –1880.
11. Nouy A. A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(23-24):1603 –1626.
12. Falco A, Nouy A. Proper generalized decomposition for nonlinear convex problems in tensor Banach spaces. *Numerische Mathematik* 2012; **121**(3):503–530. DOI: 10.1007/s00211-011-0437-5.
13. Ladeveze P. *Nonlinear Computational Structural Mechanics*. Springer: N.Y, 1999.
14. Gonzalez D, Ammar A, Chinesta F, Cueto E. Recent advances on the use of separated representations. *International Journal of Numerical Methods in Engineering* 2010; **85**(5):637–659.
15. Pruliere E, Chinesta F, Ammar A. On the deterministic solution of multidimensional parametric models using the Proper Generalized Decomposition. *Mathematics and Computers In Simulation* 2010; **81**(4):791–810.

16. Heyberger Ch, Boucard P-A, Neron D. Multiparametric analysis within the proper generalized decomposition framework. *Computational Mechanics* 2012; **49**:277–289. DOI: 10.1007/s00466-011-0646-x.

17. Ammar A, Mokdad B, Chinesta F, Keunings R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. Part II: transient simulation using space–time separated representations. *Journal of Non-Newtonian Fluid Mechanics* 2007; **144**:98–121.

18. Karhunen K. Uber lineare methoden in der wahrscheinlichkeitsrechnung. *Annales Academiæ Scientiarum Fennicæ Series A. I. Mathematica-Physica* 1947; **37**:1–79.

19. Loève MM. *Probability Theory*, 3rd ed. Van Nostrand: Princeton, NJ, 1963.

20. Lorenz EN. Empirical orthogonal functions and statistical weather prediction. *Scientific Report Number 1, Statistical Forecasting Project*, MIT, Department of Meteorology, MIT, Cambridge, Massachussets, 1956.