On the space-time separated representation of integral linear viscoelastic models

Représentation séparée espace-temps pour des comportements viscoélastiques linaires intégraux

Amine Ammar[a,b,*], Ali Zghal[a], Franck Morel[b], Francisco Chinesta[c]

[a] *UMSSDT, ENSIT, Université de Tunis, 5, avenue Taha-Hussien, Montfleury 1008, Tunis, Tunisia*
[b] *Arts et Métiers ParisTech, 2, bd du Ronceray, BP 93525, 49035 Angers cedex 01, France*
[c] *GEM, UMR CNRS–Centrale Nantes, 1, rue de la Noe, BP 92101, 44321 Nantes cedex 3, France*

Keywords:
PGD
Viscoelasticity
Integro-differential models
Fatigue

Mots-clés :
PGD
Viscoélasticité
Modèle integro-differentiel
Fatigue

ABSTRACT

The analysis of materials mechanical behavior involves many computational challenges. In this work, we are addressing the transient simulation of the mechanical behavior when the time of interest is much larger than the characteristic time of the mechanical response. This situation is encountered in many applications, as for example in the simulation of materials aging, or in structural analysis when small-amplitude oscillatory loads are applied during a long period, as it occurs for example when characterizing viscoelastic behaviors by calculating the complex modulus or when addressing fatigue simulations. Moreover, in the case of viscoelastic behaviors, the constitutive equation is many times expressed in an integral form avoiding the necessity of using internal variables, fact that results in an integro-differential model. In order to efficiently simulate such a model, we explore in this work the use of a space-time separated representation.

RÉSUMÉ

L'analyse du comportement mécanique des matériaux entraîne de nombreuses difficultés du point de vue numérique. Dans ce travail, nous allons nous focaliser sur l'une d'entre elles, celle associée à la simulation transitoire du comportement mécanique quand l'intervalle temporel d'intérêt est substantiellement plus long que le temps caractéristique associé à la réponse mécanique. Cette situation est fréquemment retrouvée dans la caractérisation rhéologique des matériaux viscoélastiques (pour la détermination du module complexe) ainsi que quand on s'attaque à la simulation de la fatigue. De plus, dans le cas des matriaux viscoélastiques, le comportement est généralement décrit par une loi de comportement intégrale qui évite le besoin d'utiliser des variables internes, donnant lieu

* Corresponding author.
E-mail addresses: Amine.Ammar@ensam.eu (A. Ammar), Ali.Zghal@gmail.com (A. Zghal), Franck.Morel@ensam.eu (F. Morel), Francisco.Chinesta@ec-nantes.fr (F. Chinesta).

à un modèle mécanique integro-différentiel. Pour une résolution efficace, nous analysons ici l'utilisation d'une représentation séparée en espace-temps.

1. Introduction

The present work focuses on the efficient treatment of models involving transient fields that must be solved in large time intervals using very small time steps. In this context, if one uses standard incremental time-discretizations, in the general case (models involving time-dependent parameters, non-linear models, etc.), one must solve at least a linear system at each time step. When the time step becomes too small as a consequence of the stability requirements, and the simulation time interval is large enough, standard incremental simulations become inefficient. They must be replaced with other more efficient techniques.

Model order reduction—MOR—techniques consider reduced bases on which the solution is projected. As such bases involve in general few functions, compared with the standard approximation bases in which an interpolation function is attached to each node of a mesh, one must consider a reduced discrete model whose solution can be in many cases solved in real time.

There are three main strategies based on MOR. The first one concerns the so-called Proper Orthogonal Decomposition—POD—that proceeds by extracting the most significant functions involved in the model's solution. For that purpose, the high-fidelity model is solved by using a standard discretization technique and different snapshots are extracted (solution at different times). Then, by applying the proper orthogonal decomposition, the most significant modes are identified and then used for projecting the solution to "similar" problems. By similar problems, we understand models involving slightly different parameters, boundary conditions, geometries, than the ones involved in the original model that served for extracting the reduced basis. There is an extensive literature regarding this issue. The interested readers can refer to [1–12] and the numerous references therein. The extraction of the reduced basis is the key point when using POD-based model order reduction, as well as its adaptivity when addressing scenarios far from the ones considered in the construction of the reduced basis [13,14].

Another family of model order reduction techniques well adapted to the solution to parametric models lies in the used of reduced bases—RB—constructed by combining a greedy algorithm and an a priori error indicator driving the exploration of the parametric space. Thus, RB techniques need for some amount off-line work, but then the reduced basis can be used online for solving different models with a perfect control of the solution accuracy because of the availability of error bounds. When the error is inadmissible, the reduced basis can be enriched by invoking again the same greedy algorithm. The interested readers can refer to [15–18] and the references therein.

Many years ago, P. Ladeveze proposed the use of space-time separated representations, at the heart of the third kind of MOR strategies here addressed and that was coined as Proper Generalized Decomposition—PGD. He introduced the space-time separated representation as one of the main bricks composing the LATIN method, a powerful nonlinear solver. The interested reader can refer to [19–24] and the valuable references therein.

When using space-time separated representations, the approximation of a transient field $u(\mathbf{x}, t)$, $\mathbf{x} \in \Omega \subset \mathcal{R}^D$, $D = 1, 2, 3$ and $t \in \mathcal{I} = (0, \mathcal{T}] \subset \mathcal{R}$, is expressed as

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

The constructor of such a separated representation consists of a double iteration loop: the first associated with the calculation of each term $(X_n(\mathbf{x}) \cdot T_n(t))$, $\forall n \in [1, \cdots, N]$, of the finite sum (1), and the other for solving the nonlinear problem related to the calculation of each couple of functions ($X_n(\mathbf{x})$ and $T_n(t)$) because both being unknown the problem results nonlinear. The numerical algorithm was deeply reported in our former works, but for the sake of completeness it has been summarized in Appendix A.

An additional advantage of separated representations is that they can be applied to the solution to problems defined in highly dimensional spaces because they allow circumventing the so-called curse of dimensionality. Thus, we applied such kind of separated representations for solving models involving many conformational coordinates encountered in quantum chemistry, kinetic theory descriptions of materials or cell signaling processes [25–29]. Moreover, we proposed adding model parameters as extra-coordinates for constructing parametric solutions that can be seen as computational vademecums from which we can perform, in real time, optimization, inverse analysis, and simulation-based control [30–35].

The interested reader can also refer to the recent reviews [36–39] and the references therein.

1.1. Non-incremental versus incremental time integrations

It is useful to reflect on the considerable difference between the above PGD strategy and traditional, incremental time integration schemes.

Indeed, the PGD allows for a non-incremental solution to time-dependent problems. Let $\mathcal{Q}_n$ denote the number of non-linear iterations required to compute the new term $X_n(x) \cdot T_n(t)$ at enrichment step n. Then, the entire PGD procedure to obtain the N-term approximation (1) involves the solution to a total of $\mathcal{Q} = (\mathcal{Q}_1 + \cdots + \mathcal{Q}_N)$ decoupled, boundary and initial value problems. The BVPs are defined over the space domain Ω, and their computational complexity scales with the mesh used to discretize them. The IVPs are defined over the time interval $\mathcal{I}$, and their complexity is usually negligible compared to that of the BVPs, even when extremely small time steps are used for their discretization.

This is vastly different from a standard, incremental solution procedure. If P is the total number of time steps for the complete simulation, i.e. $P = \mathcal{T}/\Delta t$, an incremental procedure involves the solution to a BVP in Ω at each time step, i.e. a total of P BVPs. This can be a very large number indeed, as the time step Δt must be chosen small enough to guarantee the stability of the numerical scheme.

Numerical experiments with the PGD show that the $\mathcal{Q}_n$s rarely exceed ten, while N is a few tens. Thus, the complexity of the complete PGD solution is a few hundreds of BVP solutions in Ω. This is in many applications several orders of magnitude less than the total of P BVPs that must be solved using a standard incremental procedure.

This and other related advantages in using space-time separated representations were analyzed in [26,40,41] and [42].

1.2. Separating the physical space

Sometimes, the domain Ω, assumed to be three-dimensional, can be fully or partially separated, and consequently it can be expressed as $\Omega = \Omega_x \times \Omega_y \times \Omega_z$ or $\Omega = \Omega_{xy} \times \Omega_z$, respectively. The first decomposition is related to hexahedral domains, whereas the second one is related to plate, beams or extruded domains. Both were widely considered in [43,37,44–47]. We consider below the approximations related to both scenarios.

(i) The spatial domain Ω is partially separable. In this case Eq. (1) can be rewritten as:

$$u(\mathbf{x}, z, t) = \sum_{i=1}^{N} X_i(\mathbf{x}) \cdot Z_i(z) \cdot T_i(t) \tag{2}$$

where $\mathbf{x} = (x, y) \in \Omega_{xy}$, $z \in \Omega_z$ and $t \in \mathcal{I}$.

Thus, iteration p of the alternating direction strategy at a given enrichment step n consists in the following three tasks, employing the notation introduced in Appendix A:

(a) solve in Ω_{xy} a two-dimensional BVP to obtain the function X_n^p,
(b) solve in Ω_z a one-dimensional BVP to obtain the function Z_n^p,
(c) solve in $\mathcal{I}$ a one-dimensional IVP to obtain the function T_n^p.

We can repeat our discussion regarding the complexity of this PGD non-incremental strategy versus standard incremental schemes. Clearly, what will dominate the cost of the PGD procedure is the total of $\mathcal{Q}$ two-dimensional BVPs to be solved in Ω_{xy}. The BVPs in Ω_z and IVPs in $\mathcal{I}$ being one-dimensional, their complexity is comparatively negligible. Thus, the computational cost of the PGD simulation will be orders of magnitude smaller than that of a standard incremental procedure, which requires the solution to a three-dimensional BVP at each time step.

(ii) The spatial domain $\Omega_{\mathbf{x}}$ is fully separable. In this case, Eq. (1) can be rewritten as:

$$u(x, y, z, t) = \sum_{i=1}^{N} X_i(x) \cdot Y_i(y) \cdot Z_i(z) \cdot T_i(t) \tag{3}$$

Iteration p of the alternating direction strategy at a given enrichment step n consists in the following four tasks:

(a) solve in Ω_x a one-dimensional BVP to obtain the function X_n^p,
(b) solve in Ω_y a one-dimensional BVP to obtain the function Y_n^p,
(c) solve in Ω_z a one-dimensional BVP to obtain the function Z_n^p,
(d) solve in $\mathcal{I}$ a one-dimensional IVP to obtain the function T_n^p.

The cost savings provided by the PGD are potentially phenomenal when the spatial domain is fully separable. Indeed, the complexity of the PGD simulation now scales with the one-dimensional meshes used to solve the BVPs in Ω_x, Ω_y and Ω_z, regardless of the time step used in the solution to the decoupled IVPs in $\mathcal{I}$. The computational cost is thus orders of magnitude smaller than that of a standard incremental procedure, which requires the solution to a three-dimensional BVP at each time step.

Even when the domain is not fully separable, a fully separated representation could be considered by using appropriate geometrical mappings or by immersing the non-separable domain into a fully separable one. The interested reader can refer to [48] and [49].

After this short introduction in Section 2, we define the integro-differential viscoelastic model within the small transformations framework whose space discretization will be carried out in Section 3. In Section 4, the space-time separated representation will be introduced and its construction will be considered in detail in Section 5. Finally, in Section 6, we address some numerical examples for verifying the proposed strategy and to prove its ability to address efficiently complex scenarios.

2. Linear viscoelastic integral model

The mechanical model is defined in the domain Ω whose boundary $\partial\Omega \equiv \Gamma$ is decomposed into Γ_D and Γ_N in which velocities and tractions are prescribed respectively.

We consider the standard momentum balance equation neglecting the inertia and mass terms:

$$\nabla \cdot \mathbf{\Sigma} = \mathbf{0} \tag{4}$$

where $\mathbf{\Sigma}$ is the standard Cauchy's stress tensor.

The boundary conditions write:

$$\begin{cases} \mathbf{v}(\mathbf{x} \in \Gamma_\mathrm{D}, t \in \mathcal{I}) = \mathbf{v}_g(\mathbf{x} \in \Gamma_\mathrm{D}, t \in \mathcal{I}) \\ \mathbf{\Sigma}(\mathbf{x} \in \Gamma_\mathrm{N}, t \in \mathcal{I}) \cdot \mathbf{n}(\mathbf{x} \in \Gamma_\mathrm{N}) = \mathbf{t}_g(\mathbf{x} \in \Gamma_\mathrm{N}, t \in \mathcal{I}) \end{cases} \tag{5}$$

where $\mathbf{n}$ is the unit outwards vector defined on the boundary Γ_N, $\mathbf{v}_g$ the prescribed velocities on Γ_D and $\mathbf{t}_g$ the applied tractions on Γ_N. It was assumed that the mechanical problem is linear implying both a linear constitutive law and small displacements and strains. Thus, we assume that domain Ω remains unchanged all along the time and then unaffected by the kinematics induced by the applied boundary conditions.

The weak form related to the momentum balance at each time t consists in looking for the velocity field $\mathbf{v} \in \mathcal{V}$, with $\mathcal{V} = \left\{ \mathbf{v}(\mathbf{x}, t) \in \left(\mathcal{H}^1(\Omega)\right)^3, \mathbf{v}(\mathbf{x} \in \Gamma_\mathrm{D}, t \in \mathcal{I}) = \mathbf{v}_g(\mathbf{x} \in \Gamma_\mathrm{D}, t \in \mathcal{I}) \right\}$ such that

$$\int_\Omega \mathbf{D}^* : \mathbf{\Sigma} \, \mathrm{d}\mathbf{x} = \int_{\Gamma_\mathrm{N}} \mathbf{v}^* \cdot \mathbf{t}_g \, \mathrm{d}\mathbf{x}, \quad \forall \mathbf{v}^* \in \mathcal{V}^* \tag{6}$$

with $\mathcal{V}^* = \left\{ \mathbf{v}^*(\mathbf{x}, t) \in \left(\mathcal{H}^1(\Omega)\right)^3, \mathbf{v}^*(\mathbf{x} \in \Gamma_\mathrm{D}, t \in \mathcal{I}) = \mathbf{0} \right\}$.

In Eq. (6), $\mathbf{D}$ is the usual rate of strain tensor and we use $\mathbf{\Sigma}$ instead of the usual $\boldsymbol{\sigma}$ notation for the stress tensor because in what follows $\boldsymbol{\sigma}$ will refer to the vector form of the stress tensor.

Using vector notations, integral (6) writes

$$\int_\Omega \mathbf{d}^* \cdot \boldsymbol{\sigma} \, \mathrm{d}\mathbf{x} = \int_{\Gamma_\mathrm{N}} \mathbf{v}^* \cdot \mathbf{t}_g \, \mathrm{d}\mathbf{x} \tag{7}$$

where $\mathbf{d}$ is the vector form of the rate of strain tensor $\mathbf{D}$.

The constitutive equation here considered consists of the standard viscoelastic integral form

$$\mathbf{\Sigma} = \int_{-\infty}^{t} \lambda(t-\tau) \, Tr\left(\mathbf{D}(\tau)\right) \cdot \mathbf{I} \, \mathrm{d}\tau + \int_{-\infty}^{t} 2\mu(t-\tau) \, \mathbf{D}(\tau) \, \mathrm{d}\tau \tag{8}$$

where $Tr()$ refers to the trace operator and λ and μ are two memory functions.

Even if here we only address the simplest viscoelastic constitutive model, all the developments can be extended to generalized viscoelastic models involving several relaxation times.

By using vector notations, the constitutive equation can be written as

$$\boldsymbol{\sigma} = \int_{-\infty}^{t} \mathbf{C}(t-\tau) \cdot \mathbf{d}(\tau) \, \mathrm{d}\tau \tag{9}$$

being $\mathbf{d}$ the vector form of the strain rate tensor $\mathbf{D}$. In plane strain, with

$$\boldsymbol{\sigma} = \begin{pmatrix} \mathbf{\Sigma}_{11} \\ \mathbf{\Sigma}_{22} \\ \mathbf{\Sigma}_{12} \end{pmatrix} \tag{10}$$

and

$$\mathbf{d} = \begin{pmatrix} \mathbf{D}_{11} \\ \mathbf{D}_{22} \\ 2\mathbf{D}_{12} \end{pmatrix} \tag{11}$$

the expression of $\mathbf{C}(t-\tau)$ writes:

$$\mathbf{C}(t-\tau) = \lambda(t-\tau)\begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} + \mu(t-\tau)\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \lambda(t-\tau)\mathbf{G}_\lambda + \mu(t-\tau)\mathbf{G}_\mu \tag{12}$$

The vector form of the strain rate tensor reads:

$$\mathbf{d} = \begin{pmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{v}_x \\ \mathbf{v}_y \end{pmatrix} \tag{13}$$

where $\mathbf{v}_x$ and $\mathbf{v}_y$ are the velocity vector components: $\mathbf{v}^{\mathrm{T}} = (\mathbf{v}_x, \mathbf{v}_y)$.

3. Space discretization

We can assume a standard finite element approximation of the velocity field, involving a mesh $\mathcal{M}$ consisting in $\mathcal{N}$ nodes with coordinates $\mathbf{X}_i$, $i = 1, 2, \cdots, \mathcal{N}$. Thus, if $N_i(\mathbf{x})$ denotes the shape function related to node $\mathbf{X}_i$, that by construction verifies the Kroenecker delta property $N_i(\mathbf{X}_j) = \delta_{ij}$, the velocity field van be written as

$$\begin{cases} \mathbf{v}_x = \sum_{i=1}^{\mathcal{N}} N_i(\mathbf{x})\, \mathbf{v}_x(\mathbf{X}_i) = \mathbf{N}^{\mathrm{T}} \cdot \mathbf{V}_x \\ \mathbf{v}_y = \sum_{i=1}^{\mathcal{N}} N_i(\mathbf{x})\, \mathbf{v}_y(\mathbf{X}_i) = \mathbf{N}^{\mathrm{T}} \cdot \mathbf{V}_y \end{cases} \tag{14}$$

where $\mathbf{V}_x$ and $\mathbf{V}_y$ are the vectors that contain the nodal velocity components $\mathbf{v}_x(\mathbf{X}_i)$ and $\mathbf{v}_y(\mathbf{X}_i)$ $(i = 1, 2, \cdots, \mathcal{N})$ respectively and $\mathbf{N}$ the vector containing the different shape functions.

This approximation can be written in a more compact form according to:

$$\mathbf{v} = \begin{pmatrix} \mathbf{N}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0}^{\mathrm{T}} & \mathbf{N}^{\mathrm{T}} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{V}_x \\ \mathbf{V}_y \end{pmatrix} = \mathbf{M} \cdot \mathbf{V} \tag{15}$$

where $\mathbf{0}^{\mathrm{T}}$ is the row vector of size $\mathcal{N}$ with null entries.

Thus, the vector form of the rate of strain $\mathbf{d}$ reads:

$$\mathbf{d} = \begin{pmatrix} \frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial x} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0}^{\mathrm{T}} & \frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial y} \\ \frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial y} & \frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial x} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{V}_x \\ \mathbf{V}_y \end{pmatrix} = \mathbf{B} \cdot \mathbf{V} \tag{16}$$

Now, coming back to the weak form (7), its left member results

$$\int_\Omega \mathbf{d}^* \cdot \boldsymbol{\sigma}\, \mathrm{d}\mathbf{x} = \mathbf{V}^{*\mathrm{T}}(t) \cdot \left\{ \int_{-\infty}^{t} \left\{ \lambda(t-\tau)\mathbf{K}_\lambda + \mu(t-\tau)\mathbf{K}_\mu \right\} \cdot \mathbf{V}(\tau)\, \mathrm{d}\tau \right\} \tag{17}$$

with

$$\begin{cases} \mathbf{K}_\lambda = \int_\Omega \mathbf{B}^{\mathrm{T}} \cdot \mathbf{G}_\lambda \cdot \mathbf{B}\, \mathrm{d}\mathbf{x} \\ \mathbf{K}_\mu = \int_\Omega \mathbf{B}^{\mathrm{T}} \cdot \mathbf{G}_\mu \cdot \mathbf{B}\, \mathrm{d}\mathbf{x} \end{cases} \tag{18}$$

On the other hand, the right-hand-side member of Eq. (6) writes:

$$\int_{\Gamma_{\mathrm{N}}} \mathbf{v}^*(\mathbf{x},t)\cdot\mathbf{t}_g(\mathbf{x},t)\,\mathrm{d}\mathbf{x} = \mathbf{V}^{*\mathrm{T}}\cdot\left\{\int_{\Gamma_{\mathrm{N}}} \mathbf{M}^{\mathrm{T}}\cdot\mathbf{M}\,\mathrm{d}\mathbf{x}\right\}\cdot\mathbf{f}(t) = \mathbf{V}^{*\mathrm{T}}(t)\cdot\mathbf{F}(t) \tag{19}$$

Thus finally, after discretizing in space, the problem reads:

$$\mathbf{V}^{*\mathrm{T}}(t)\cdot\left\{\int_{-\infty}^{t}\left\{\lambda(t-\tau)\mathbf{K}_{\lambda}+\mu(t-\tau)\mathbf{K}_{\mu}\right\}\cdot\mathbf{V}(\tau)\,\mathrm{d}\tau\right\} = \mathbf{V}^{*\mathrm{T}}(t)\cdot\mathbf{F}(t) \tag{20}$$

which leads to the linear system:

$$\int_{-\infty}^{t}\left\{\lambda(t-\tau)\mathbf{K}_{\lambda}+\mu(t-\tau)\mathbf{K}_{\mu}\right\}\cdot\mathbf{V}(\tau)\,\mathrm{d}\tau = \mathbf{F}(t) \tag{21}$$

complemented with the Dirichlet boundary conditions applying on Γ_{D}.

Eq. (21) can be rewritten as

$$\mathbf{K}_{\lambda}\cdot\int_{-\infty}^{t}\lambda(t-\tau)\mathbf{V}(\tau)\,\mathrm{d}\tau + \mathbf{K}_{\mu}\cdot\int_{-\infty}^{t}\mu(t-\tau)\mathbf{V}(\tau)\,\mathrm{d}\tau = \mathbf{F}(t) \tag{22}$$

4. Space-time separated representation

Now, we consider Eq. (22) and assume that both the applied traction $\mathbf{F}(t)$ and the velocity field $\mathbf{V}(\tau)$ can be written in a separated form, respectively:

$$\mathbf{F}(t) \approx \sum_{i=1}^{N_F}\mathbf{S}_i\,\mathcal{S}_i(t) \tag{23}$$

and

$$\mathbf{V}(t) \approx \sum_{i=1}^{N_V}\mathbf{X}_i\,\mathcal{X}_i(t) \tag{24}$$

Thus, Eq. (22) results:

$$\sum_{i=1}^{N_V}\left\{\mathbf{K}_{\lambda}\cdot\mathbf{X}_i\cdot\int_{-\infty}^{t}\lambda(t-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau + \mathbf{K}_{\mu}\cdot\mathbf{X}_i\cdot\int_{-\infty}^{t}\mu(t-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau\right\} = \sum_{i=1}^{N_F}\mathbf{S}_i\,\mathcal{S}_i(t) \tag{25}$$

The time integrals can be approximated by using an adequate numerical quadrature. If we assume that $\mathbf{F}(t)$ and $\mathbf{V}(t)$ vanish at $t \le 0$, and consider discrete times $t_n = n\Delta t$, then we can write:

$$\begin{cases} \displaystyle\int_{0}^{t_1} g(t)\,\mathrm{d}t \approx g(t_1)\Delta t \\ \displaystyle\int_{0}^{t_2} g(t)\,\mathrm{d}t \approx g(t_1)\Delta t + g(t_2)\Delta t \\ \vdots \\ \displaystyle\int_{0}^{t_n} g(t)\,\mathrm{d}t \approx \sum_{i=1}^{i=n} g(t_i)\Delta t \end{cases} \tag{26}$$

that applied to the integrals in Eq. (25), for example those involving $\mathcal{X}_i(\tau)$, results,

$$\begin{cases} \int\limits_0^{t_1} \lambda(t_1-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \approx \lambda(t_0)\mathcal{X}_i(t_1)\Delta t \\ \int\limits_0^{t_2} \lambda(t_2-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \approx \lambda(t_0)\mathcal{X}_i(t_2)\Delta t + \lambda(t_1)\mathcal{X}_i(t_1)\Delta t \\ \vdots \\ \int\limits_0^{t_n} \lambda(t_n-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \approx \sum\limits_{j=1}^{n} \lambda(t_n-t_j)\mathcal{X}_i(t_j)\Delta t \\ \vdots \end{cases} \tag{27}$$

whose matrix form reads:

$$\begin{pmatrix} \int\limits_0^{t_1} \lambda(t_1-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \\ \int\limits_0^{t_2} \lambda(t_2-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \\ \vdots \\ \int\limits_0^{t_P} \lambda(t_P-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \end{pmatrix} = \Delta t \begin{pmatrix} \lambda(t_0) & 0 & \cdots & 0 \\ \lambda(t_1) & \lambda(t_0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \lambda(t_P) & \lambda(t_{P-1}) & \cdots & \lambda(t_0) \end{pmatrix} \cdot \begin{pmatrix} \mathcal{X}_i(t_1) \\ \mathcal{X}_i(t_2) \\ \vdots \\ \mathcal{X}_i(t_P) \end{pmatrix} = \Delta t\, \mathbf{L}_\lambda \cdot \mathbb{X}_i \tag{28}$$

with $P\Delta t = \mathcal{T}$.

Considering now the integral involving the memory function $\mu(t-\tau)$ and using the same quadrature, it results:

$$\begin{pmatrix} \int\limits_0^{t_1} \mu(t_1-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \\ \int\limits_0^{t_2} \mu(t_2-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \\ \vdots \\ \int\limits_0^{t_P} \mu(t_P-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \end{pmatrix} = \Delta t \begin{pmatrix} \mu(t_0) & 0 & \cdots & 0 \\ \mu(t_1) & \mu(t_0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mu(t_P) & \mu(t_{P-1}) & \cdots & \mu(t_0) \end{pmatrix} \cdot \begin{pmatrix} \mathcal{X}_i(t_1) \\ \mathcal{X}_i(t_2) \\ \vdots \\ \mathcal{X}_i(t_P) \end{pmatrix} = \Delta t\, \mathbf{L}_\mu \cdot \mathbb{X}_i \tag{29}$$

For evanescent memory, functions $\lambda(t_m)$ and $\mu(t_m)$ vanish up to a certain value n, and consequently only m diagonals of $\mathbf{L}_\lambda$ and $\mathbf{L}_\mu$ must be computed.

5. Separated representation constructor

We consider the previous discrete form (25)

$$\sum_{i=1}^{N_\mathrm{V}} \left\{ \mathbf{K}_\lambda \cdot \mathbf{X}_i \cdot \int\limits_{-\infty}^{t} \lambda(t-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau + \mathbf{K}_\mu \cdot \mathbf{X}_i \cdot \int\limits_{-\infty}^{t} \mu(t-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \right\} = \sum_{i=1}^{N_F} \mathbf{S}_i\, \mathcal{S}_i(t) \tag{30}$$

and assume that at present iteration we already computed the $q-1$ first terms of the finite sum (24), with $q-1 < N_\mathrm{V}$, leading to the $(q-1)$-approximate:

$$\mathbf{V}^{q-1}(t) = \sum_{i=1}^{q-1} \mathbf{X}_i\, \mathcal{X}_i(t) \tag{31}$$

At the present iteration, we look for the q-approximate of $\mathbf{V}(t)$ that can be written as

$$\mathbf{V}^q(t) = \sum_{i=1}^{q} \mathbf{X}_i \, \mathcal{X}_i(t) = \mathbf{V}^{q-1} + \mathbf{X}_q \mathcal{X}_q(t) \tag{32}$$

Now, in order to apply the rationale described in Appendix A, we consider the test function

$$\mathbf{V}_q^* = \mathbf{X}^* \mathcal{X}_q(t) + \mathbf{X}_q \mathcal{X}^*(t) \tag{33}$$

and from (25) the extended weak form:

$$\int_0^{\mathcal{T}} \left(\mathbf{X}^* \mathcal{X}_q(t) + \mathbf{X}_q \mathcal{X}^*(t) \right) \cdot \left\{ \sum_{i=1}^{q} \left\{ \mathbf{K}_\lambda \cdot \mathbf{X}_i \cdot \int_{-\infty}^{t} \lambda(t-\tau)\mathcal{X}_i(\tau)\, \mathrm{d}\tau + \mathbf{K}_\mu \cdot \mathbf{X}_i \cdot \int_{-\infty}^{t} \mu(t-\tau)\mathcal{X}_i(\tau)\, \mathrm{d}\tau \right\} - \sum_{i=1}^{N_F} \mathbf{S}_i \, \mathcal{S}_i(t) \right\} \mathrm{d}t = 0 \tag{34}$$

that can be rewritten under the form:

$$\begin{aligned} &\int_0^{\mathcal{T}} \left(\mathbf{X}^* \mathcal{X}_q(t) + \mathbf{X}_q \mathcal{X}^*(t) \right) \cdot \left\{ \mathbf{K}_\lambda \cdot \mathbf{X}_q \cdot \int_{-\infty}^{t} \lambda(t-\tau)\mathcal{X}_q(\tau)\, \mathrm{d}\tau + \mathbf{K}_\mu \cdot \mathbf{X}_q \cdot \int_{-\infty}^{t} \mu(t-\tau)\mathcal{X}_q(\tau)\, \mathrm{d}\tau \right\} \mathrm{d}t \\ &= - \int_0^{\mathcal{T}} \left(\mathbf{X}^* \mathcal{X}_q(t) + \mathbf{X}_q \mathcal{X}^*(t) \right) \\ &\quad \cdot \left\{ \sum_{i=1}^{q-1} \left\{ \mathbf{K}_\lambda \cdot \mathbf{X}_i \cdot \int_{-\infty}^{t} \lambda(t-\tau)\mathcal{X}_i(\tau)\, \mathrm{d}\tau + \mathbf{K}_\mu \cdot \mathbf{X}_i \cdot \int_{-\infty}^{t} \mu(t-\tau)\mathcal{X}_i(\tau)\, \mathrm{d}\tau \right\} - \sum_{i=1}^{N_F} \mathbf{S}_i \, \mathcal{S}_i(t) \right\} \mathrm{d}t \end{aligned} \tag{35}$$

that contains the unknown fields in the left-hand-side member and the known (already computed) fields in the right-hand-side one.

Now, as described in Appendix A, for computing the couple of unknown functions $\mathbf{X}_q$ and $\mathcal{X}_q(t)$, we are considering again an alternated directions fixed point strategy that computed $\mathbf{X}_q$ by assuming $\mathcal{X}_q(t)$ known (it is randomly chosen at the beginning of the process), and then updating $\mathcal{X}_q(t)$ from the just calculated $\mathbf{X}_q$. The process continue until reaching convergence, that is, the fixed point.

In what follow we are developing both steps.

5.1. Calculation of $\mathbf{X}_q$

When calculating $\mathbf{X}_q$, $\mathcal{X}_q(t)$ is assumed known ($\mathcal{X}^*(t) = 0$ in Eq. (35)), and with it all functions depending on time. Thus, all time integrals can be performed, leading to a linear problem for calculating the unknown vector $\mathbf{X}_q$.

The first integral in Eq. (35) concerns

$$\int_0^{\mathcal{T}} \mathcal{X}_q(t) \left\{ \int_{-\infty}^{t} \lambda(t-\tau)\mathcal{X}_q(\tau)\, \mathrm{d}\tau \right\} \mathrm{d}t \tag{36}$$

that using the notation previously introduced results

$$\alpha_q^\lambda = \int_0^{\mathcal{T}} \mathcal{X}_q(t) \left\{ \int_{-\infty}^{t} \lambda(t-\tau)\mathcal{X}_q(\tau)\, \mathrm{d}\tau \right\} \mathrm{d}t = \Delta t^2 \, \mathbb{X}_q^{\mathrm{T}} \cdot \mathbf{L}_\lambda \cdot \mathbb{X}_q \tag{37}$$

Similarly, we can define:

$$\alpha_q^\mu = \int_0^{\mathcal{T}} \mathcal{X}_q(t) \left\{ \int_{-\infty}^{t} \mu(t-\tau)\mathcal{X}_q(\tau)\, \mathrm{d}\tau \right\} \mathrm{d}t = \Delta t^2 \, \mathbb{X}_q^{\mathrm{T}} \cdot \mathbf{L}_\mu \cdot \mathbb{X}_q, \tag{38}$$

$$\alpha_{q,i}^\lambda = \int_0^{\mathcal{T}} \mathcal{X}_q(t) \left\{ \int_{-\infty}^{t} \lambda(t-\tau)\mathcal{X}_i(\tau)\, \mathrm{d}\tau \right\} \mathrm{d}t = \Delta t^2 \, \mathbb{X}_q^{\mathrm{T}} \cdot \mathbf{L}_\lambda \cdot \mathbb{X}_i, \tag{39}$$

$$\alpha^{\mu}_{q,i} = \int_{0}^{\mathcal{T}} \mathcal{X}_q(t) \left\{ \int_{-\infty}^{t} \mu(t-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \right\} \mathrm{d}t = \Delta t^2\, \mathbb{X}_q^{\mathrm{T}} \cdot \mathbf{L}_{\mu} \cdot \mathbb{X}_i \tag{40}$$

$\forall i \in [1, 2, \cdots, q-1]$; and

$$\beta_{q,i} = \int_{0}^{\mathcal{T}} \mathcal{X}_q(t) \cdot \mathcal{S}_i(t)\,\mathrm{d}t \tag{41}$$

$\forall i \in [1, 2, \cdots, N_F]$; from with Eq. (35) reduced to:

$$\mathbf{X}^* \cdot \left\{ \alpha^{\lambda}_{q} \mathbf{K}_{\lambda} \cdot \mathbf{X}_q + \alpha^{\mu}_{q} \mathbf{K}_{\mu} \cdot \mathbf{X}_q \right\} = \mathbf{X}^* \cdot \left\{ \sum_{i=1}^{q-1} \left\{ \alpha^{\lambda}_{q,i} \mathbf{K}_{\lambda} \cdot \mathbf{X}_i + \alpha^{\mu}_{q,i} \mathbf{K}_{\mu} \cdot \mathbf{X}_i \right\} - \sum_{i=1}^{N_F} \beta_{q,i} \mathbf{S}_i \right\} \tag{42}$$

or its associated linear system

$$\left\{ \alpha^{\lambda}_{q} \mathbf{K}_{\lambda} \cdot \mathbf{X}_q + \alpha^{\mu}_{q} \mathbf{K}_{\mu} \cdot \mathbf{X}_q \right\} = \left\{ \sum_{i=1}^{q-1} \left\{ \alpha^{\lambda}_{q,i} \mathbf{K}_{\lambda} \cdot \mathbf{X}_i + \alpha^{\mu}_{q,i} \mathbf{K}_{\mu} \cdot \mathbf{X}_i \right\} - \sum_{i=1}^{N_F} \beta_{q,i} \mathbf{S}_i \right\} \tag{43}$$

that can be solved for calculating $\mathbf{X}_q$

$$\left\{ \alpha^{\lambda}_{q} \mathbf{K}_{\lambda} + \alpha^{\mu}_{q} \mathbf{K}_{\mu} \right\} \cdot \mathbf{X}_q = \left\{ \sum_{i=1}^{q-1} \left\{ \alpha^{\lambda}_{q,i} \mathbf{K}_{\lambda} \cdot \mathbf{X}_i + \alpha^{\mu}_{q,i} \mathbf{K}_{\mu} \cdot \mathbf{X}_i \right\} - \sum_{i=1}^{N_F} \beta_{q,i} \mathbf{S}_i \right\} \tag{44}$$

or

$$\mathbf{X}_q = \left\{ \alpha^{\lambda}_{q} \mathbf{K}_{\lambda} + \alpha^{\mu}_{q} \mathbf{K}_{\mu} \right\}^{-1} \cdot \left\{ \sum_{i=1}^{q-1} \left\{ \alpha^{\lambda}_{q,i} \mathbf{K}_{\lambda} \cdot \mathbf{X}_i + \alpha^{\mu}_{q,i} \mathbf{K}_{\mu} \cdot \mathbf{X}_i \right\} - \sum_{i=1}^{N_F} \beta_{q,i} \mathbf{S}_i \right\} \tag{45}$$

5.2. Calculation of $\mathcal{X}_q(t)$

When calculating $\mathcal{X}_q(t)$, $\mathbf{X}_q$ is assumed known ($\mathbf{X}^* = \mathbf{0}$ in Eq. (35)). Thus, all matrix products in Eq. (35) can be calculated, from which the next scalars result:

$$\gamma^{\lambda}_{q} = \mathbf{X}_q^{\mathrm{T}} \cdot \mathbf{K}_{\lambda} \cdot \mathbf{X}_q \tag{46}$$

$$\gamma^{\mu}_{q} = \mathbf{X}_q^{\mathrm{T}} \cdot \mathbf{K}_{\mu} \cdot \mathbf{X}_q \tag{47}$$

$$\gamma^{\lambda}_{q,i} = \mathbf{X}_q^{\mathrm{T}} \cdot \mathbf{K}_{\lambda} \cdot \mathbf{X}_i \tag{48}$$

$$\gamma^{\mu}_{q,i} = \mathbf{X}_q^{\mathrm{T}} \cdot \mathbf{K}_{\mu} \cdot \mathbf{X}_i \tag{49}$$

$\forall i \in [1, 2, \cdots, q-1]$; and

$$\delta_{q,i} = \mathbf{X}_q^{\mathrm{T}} \cdot \mathbf{S}_i \tag{50}$$

$\forall i \in [1, 2, \cdots, N_F]$.

By using previous notation, Eq. (35) reduces to:

$$\begin{aligned} &\int_{0}^{\mathcal{T}} \mathcal{X}^*(t) \left\{ \gamma^{\lambda}_{q} \int_{-\infty}^{t} \lambda(t-\tau)\mathcal{X}_q(\tau)\,\mathrm{d}\tau + \gamma^{\mu}_{q} \int_{-\infty}^{t} \mu(t-\tau)\mathcal{X}_q(\tau)\,\mathrm{d}\tau \right\} \mathrm{d}t \\ &\quad = -\int_{0}^{\mathcal{T}} \mathcal{X}^*(t) \left\{ \sum_{i=1}^{q-1} \left\{ \gamma^{\lambda}_{q,i} \int_{-\infty}^{t} \lambda(t-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau + \gamma^{\mu}_{q,i} \int_{-\infty}^{t} \mu(t-\tau)\mathcal{X}_i(\tau)\,\mathrm{d}\tau \right\} - \sum_{i=1}^{N_F} \delta_{q,i} \mathcal{S}_i(t) \right\} \mathrm{d}t \end{aligned} \tag{51}$$

or

$$\Delta t^2\, \mathbb{X}^{*\mathrm{T}} \cdot \left\{ \gamma^{\lambda}_{q} \mathbf{L}_{\lambda} + \gamma^{\mu}_{q} \mathbf{L}_{\mu} \right\} \cdot \mathbb{X}_q = -\mathbb{X}^{*\mathrm{T}} \cdot \left\{ \Delta t^2 \sum_{i=1}^{q-1} \left\{ \gamma^{\lambda}_{q,i} \mathbf{L}_{\lambda} + \gamma^{\mu}_{q,i} \mathbf{L}_{\mu} \right\} \cdot \mathbb{X}_i - \Delta t \sum_{i=1}^{N_F} \delta_{q,i} \mathbb{S}_i(t) \right\} \tag{52}$$

where $\mathbb{S}_i$ is the vector that contains the value of $\mathcal{S}_i(t)$ at times $n \cdot \Delta t$, $n \in [1, 2, \cdots, P]$. Thus the strong form related to (52) results

$$\left\{\gamma_q^\lambda \mathbf{L}_\lambda + \gamma_q^\mu \mathbf{L}_\mu\right\} \cdot \mathbb{X}_q = \left\{\sum_{i=1}^{q-1} \left\{\gamma_{q,i}^\lambda \mathbf{L}_\lambda + \gamma_{q,i}^\mu \mathbf{L}_\mu\right\} \cdot \mathbb{X}_i - \frac{1}{\Delta t} \sum_{i=1}^{N_F} \delta_{q,i} \mathbb{S}_i(t)\right\} \tag{53}$$

from which it finally results:

$$\mathbb{X}_q = \left\{\gamma_q^\lambda \mathbf{L}_\lambda + \gamma_q^\mu \mathbf{L}_\mu\right\}^{-1} \cdot \left\{\sum_{i=1}^{q-1} \left\{\gamma_{q,i}^\lambda \mathbf{L}_\lambda + \gamma_{q,i}^\mu \mathbf{L}_\mu\right\} \cdot \mathbb{X}_i - \frac{1}{\Delta t} \sum_{i=1}^{N_F} \delta_{q,i} \mathbb{S}_i(t)\right\} \tag{54}$$

5.3. Separated representation constructor overwiew

- Assuming at iteration $q \geq 1$ vectors $\mathbf{X}_i$ and $\mathcal{X}_i(t)$, $i \in [1, 2, \cdots, q-1]$, known
- while $\|\mathbf{V}^{q-1}(t) - \mathbf{V}^{q-2}(t)\| > \epsilon$ calculate $\mathbf{V}^q(t) = \mathbf{V}^{q-1}(t) + \mathbf{X}_q \mathcal{X}_q(t)$ by solving until reaching the fixed point the two problems below:
  - calculate $\mathbf{X}_q$ from Eq. (45)

$$\mathbf{X}_q = \left\{\alpha_q^\lambda \mathbf{K}_\lambda + \alpha_q^\mu \mathbf{K}_\mu\right\}^{-1} \cdot \left\{\sum_{i=1}^{q-1} \left\{\alpha_{q,i}^\lambda \mathbf{K}_\lambda \cdot \mathbf{X}_i + \alpha_{q,i}^\mu \mathbf{K}_\mu \cdot \mathbf{X}_i\right\} - \sum_{i=1}^{N_F} \beta_{q,i} \mathbf{S}_i\right\} \tag{55}$$

  - calculate $\mathcal{X}_q(t)$ from Eq. (54)

$$\mathbb{X}_q = \left\{\gamma_q^\lambda \mathbf{L}_\lambda + \gamma_q^\mu \mathbf{L}_\mu\right\}^{-1} \cdot \left\{\sum_{i=1}^{q-1} \left\{\gamma_{q,i}^\lambda \mathbf{L}_\lambda + \gamma_{q,i}^\mu \mathbf{L}_\mu\right\} \cdot \mathbb{X}_i - \frac{1}{\Delta t} \sum_{i=1}^{N_F} \delta_{q,i} \mathbb{S}_i(t)\right\} \tag{56}$$

6. Numerical results

In this section, we are first verifying the proposed strategy by solving a quite simple problem and then addressing a more complex problem close to the one found in assembled systems involving elastomers. As we are considering here linear behaviors, it is expected that after a certain time the response becomes steady harmonic, with a certain phase angle with respect to the applied load. Thus, simulations in the linear case do not need to cover the entire life period, but only the transient regime.

6.1. Strategy verification

For strategy verification, we consider the plane deformation quasi-incompressible viscoelastic model in $\Omega = (0, L) \times (0, H)$, with $L = 1$ and $H = 1$; and $\mathcal{I} = (0, \mathcal{T}]$, with $\mathcal{T} = 0.25$ (all units in the metric system).

A harmonic traction is applied to the upper boundary $y = H$ given by $\mathbf{t}_g(x, y = H, t) = (\sin(\omega t), 0)^{\mathrm{T}}$, with $\omega = 2\pi$. The lateral sides are free, that is $\mathbf{t}_g(x = 0, y, t) = \mathbf{t}_g(x = L, y, t) = \mathbf{0}$. On the lower boundary, the displacement and velocities are enforced to zero, that is $\mathbf{v}(x, y = 0, t) = \mathbf{0}$.

We considered the viscoelastic law given by the Maxwell's model (assuming small displacements and strains)

$$\theta \frac{\mathrm{d}\mathbf{\Sigma}}{\mathrm{d}t} + \mathbf{\Sigma} = 2\, G\, \theta \mathbf{D} \tag{57}$$

where G denotes the shear modulus and θ the relaxation time.

The integral counterpart of the Maxwell model (57) reads:

$$\mathbf{\Sigma}(t) = \int_0^t 2\, G\, \mathrm{e}^{-\frac{t-\tau}{\theta}} \mathbf{D}\, \mathrm{d}\tau \tag{58}$$

Using the notation introduced in the previous sections we consider:

$$\begin{cases} \lambda(t) = \Lambda\, \mathrm{e}^{-\frac{t}{\theta}} \\ \mu(t) = 2\, G\, \mathrm{e}^{-\frac{t}{\theta}} \end{cases} \tag{59}$$

In the numerical tests carried out, we considered Λ large enough for ensuring the model incompressibility and $2\, G = 0.3356$.

Using the strategy described in the previous section we computed the velocity field related to the applied load, and the displacement was obtained by integrating the calculated velocity. For Maxwell's model it is well known that the tangent of the phase angle (angle between the applied load and the resulting displacement), $\tan(\varphi)$, is related to the relaxation time and the applied frequency from:

$$\tan(\varphi)=\frac{1}{\omega\theta} \tag{60}$$

Thus, it follows from Eq. (60) that the knowledge of the phase angle φ allows identifying the relaxation time θ. To check it, we solved the just-presented model for three different values of the relaxation time: $\theta_1=0.05$, $\theta_2=\frac{1}{2\pi}$, and $\theta_3=2$. By solving the three viscoelastic problems, we obtained the three associated displacement fields $\mathbf{u}_i(\mathbf{x},t)$, $i=1,2,3$. Now, the post-treatment of the obtained results allows calculating the three phase angles φ_i, $i=1,2,3$ and from them the three relaxation times that were in perfect agreement with the ones that were chosen for performing the calculation.

6.2. Analysis of a rigid–viscoelastic joining

In the present analysis, we consider again a square domain $\Omega=(-L,L)\times(-H,H)$, with $L=3$ and $H=3$, containing a circular hole $\mathcal{H}(\mathcal{C},R)$ centered at $\mathcal{C}=\mathbf{0}$ and of radius $R=1$. The system was analyzed in the time interval $\mathcal{I}=(0,\mathcal{T}]$, with $\mathcal{T}=20$. The velocity was prescribed on the domain boundary $\Gamma\equiv\partial\Omega$, consisting of the external boundary Γ_e and of the internal one (hole boundary) $\Gamma_i\equiv\partial\mathcal{H}$, $\Gamma=\Gamma_e\cup\Gamma_i$:

$$\begin{cases}\mathbf{v}_g(\mathbf{x}\in\Gamma_e,t)=(\sin(0.1\pi t^2),0)^{\mathrm{T}}\\ \mathbf{v}_g(\mathbf{x}\in\Gamma_i,t)=\mathbf{0}\end{cases} \tag{61}$$

The behavior law was given by

$$\begin{cases}\lambda(t)=\Lambda\left(a_1\,\mathrm{e}^{-\frac{t}{b_1}}+a_2\,\mathrm{e}^{-\frac{t}{b_2}}\right)\\ \mu(t)=\Upsilon\left(c_1\,\mathrm{e}^{-\frac{t}{c_1}}+c_2\,\mathrm{e}^{-\frac{t}{d_2}}\right)\end{cases} \tag{62}$$

with $a_1=a_2=c_1=c_2=1$, $b_1=5$, $b_2=0.1$, $d_1=10$, $d_2=0.5$, $\Lambda=\frac{E\nu}{(1+\nu)(1-2\nu)}$, $\Upsilon=\frac{E}{2(1+\nu)}$, $E=1$ and $\nu=0.3$.

The solution $\mathbf{V}(t)$ involves only six modes for the prescribed precision ($\mathbf{X}_i$, $\mathcal{X}_i$), $i=1,\cdots,6$, whose four most significant are depicted in Fig. 1. The time-associated functions $\mathcal{X}_i(t)$, $i=1,\cdots,4$, are depicted in Fig. 2. In Fig. 3 the applied displacement and the associated traction are represented. From this figure, it can be noticed that when the frequency of the applied displacement increases, the tension amplitude decreases and the phase angle increases, as expected for viscoelastic behaviors.

7. Conclusions

In this work, we extended the domain of applicability of space-time separated representations to integro-differential models describing viscoelastic behaviors. The advantages in using such decomposition follow from the fact that space and time are discretized independently and then a fine resolution of both discretizations can be considered, without affecting the global efficiency of the coupled model. Depending on the analyzed case, the speeding up can reach some orders of magnitude.

Here we used the most direct formulation that only involves kinematic degrees of freedom (velocities); however, a mixed formulation (stress velocity) as in the LATIN method (see [19]) could be envisaged in order to separate the global linear problem from the local one that depends on the history despite of its linearity.

Another appealing possibility in using such kind of separated representations is the fact of introducing some model parameter as extra-coordinate in order to calculate a general parametric solution to the transient integer-differential model. This possibility, and the consideration of nonlinear viscoelastic behaviors, constitute some work in progress.

Acknowledgements

Francisco Chinesta acknowledges the support of the Institut Universitaire de France (IUF).

Appendix A. Space-time separated representation constructor

For the sake of simplicity, we consider here the one-dimensional problem of computing the field $u(x,t)$ governed by

$$\frac{\partial u}{\partial t}-k\frac{\partial^2 u}{\partial x^2}=f \tag{63}$$

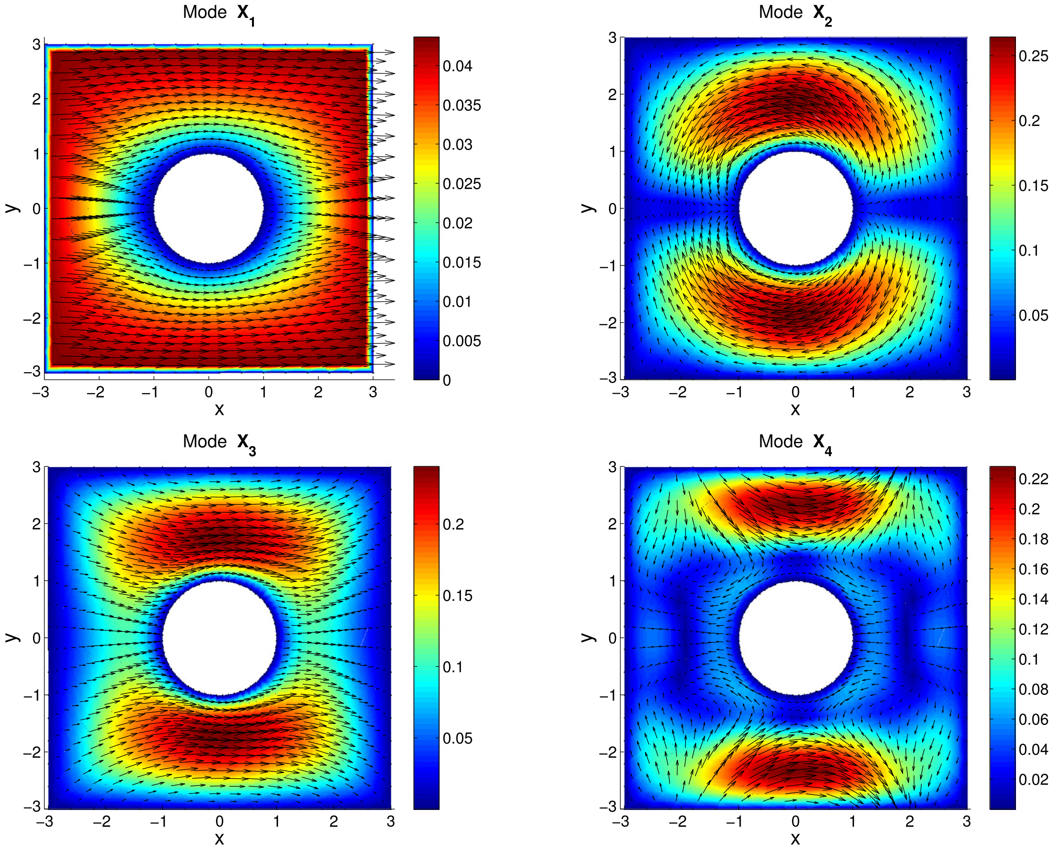

Fig. 1. (Color online.) Four most significant modes $\mathbf{X}_i$: (top-left) $\mathbf{X}_1$; (top-right) $\mathbf{X}_2$; (down-left) $\mathbf{X}_3$ and (down-right) $\mathbf{X}_4$.

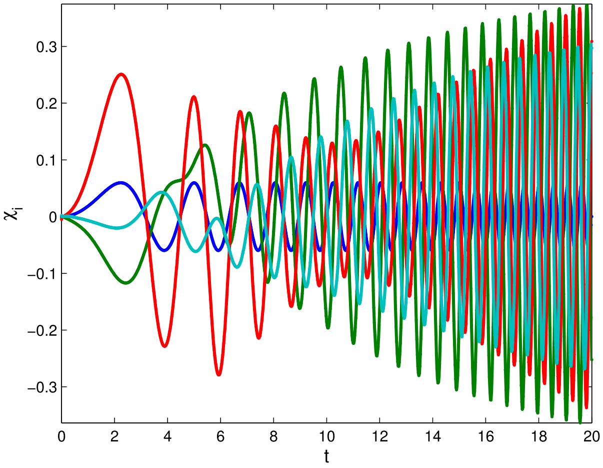

Fig. 2. (Color online.) Four most significant modes $\mathcal{X}_i(t)$, $i = 1, \cdots, 4$.

in the space-time domain $\Omega = \Omega_x \times \Omega_t = (0, L) \times (0, \tau]$. The diffusivity k and source term f are assumed to be constant. We specify homogeneous initial and boundary conditions, i.e. $u(x, t = 0) = u(x = 0, t) = u(x = L, t) = 0$. More complex scenarios were addressed in [50].

The weighted residual form of (63) reads

$$\int\limits_{\Omega_x \times \Omega_t} u^* \left(\frac{\partial u}{\partial t} - k \frac{\partial^2 u}{\partial x^2} - f \right) \, dx \, dt \; = \; 0 \tag{64}$$

for all suitable test functions u^*.

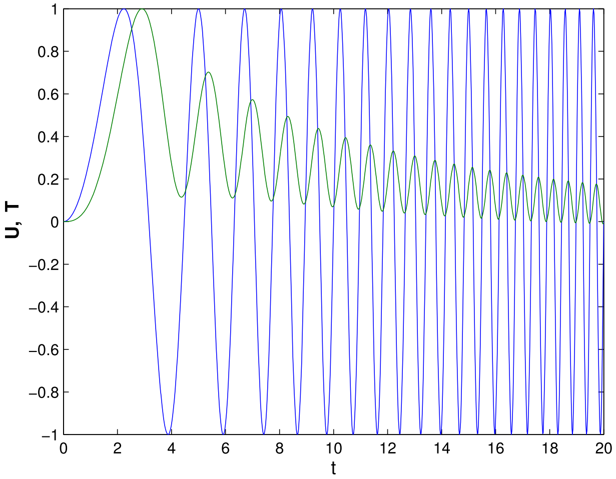

Fig. 3. (Color online.) Applied displacement (blue curve) versus its associated tension (green curve).

Our objective is to obtain a PGD approximate solution in the separated form

$$u(x,t) \approx \sum_{i=1}^{N} X_i(x) \cdot T_i(t) \tag{65}$$

We do so by computing each term of the expansion at each step of an enrichment process, until a suitable stopping criterion is met.

A.1. Progressive construction of the separated representation

At enrichment step n, the $n-1$ first terms of the PGD approximation (65) are known:

$$u^{n-1}(x,t) = \sum_{i=1}^{n-1} X_i(x) \cdot T_i(t) \tag{66}$$

We now wish to compute the next term $X_n(x) \cdot T_n(t)$ to get the enriched PGD solution

$$u^{n}(x,t) = u^{n-1}(x,t) + X_n(x) \cdot T_n(t) = \sum_{i=1}^{n-1} X_i(x) \cdot T_i(t) + X_n(x) \cdot T_n(t) \tag{67}$$

One must thus solve a non-linear problem for the unknown functions $X_n(x)$ and $T_n(t)$ by means of a suitable iterative scheme. We rely on the simple but robust alternating direction scheme.

At enrichment step n, the PGD approximation $u^{n,p}$ obtained at iteration p is given by

$$u^{n,p}(x,t) = u^{n-1}(x,t) + X_n^p(x) \cdot T_n^p(t) \tag{68}$$

Starting from an arbitrary initial guess $T_n^0(t)$, the alternating direction strategy computes $X_n^p(x)$ from $T_n^{p-1}(t)$, and then $T_n^p(t)$ from $X_n^p(x)$. These non-linear iterations proceed until reaching a fixed point within a user-specified tolerance ϵ, i.e.

$$\| X_n^p(x) \cdot Y_n^p(y) - X_n^{p-1}(x) \cdot Y_n^{p-1}(y) \| < \epsilon \tag{69}$$

where $\| \cdot \|$ is a suitable norm.

The enrichment step n thus ends with the assignments $X_n(x) \leftarrow X_n^p(x)$ and $T_n(t) \leftarrow T_n^p(t)$.

The enrichment process itself stops when an appropriate measure of error $\mathcal{E}(n)$ becomes small enough, i.e. $\mathcal{E}(n) < \tilde{\epsilon}$. One can apply the stopping criteria discussed in [51,52].

Let us look at one particular alternating direction iteration at a given enrichment step.

A.2. Alternating direction strategy

Each iteration of the alternating direction scheme consists in the following two steps.

- Calculating $X_n^p(x)$ from $T_n^{p-1}(t)$.
  At this stage, the approximation is given by

$$u^n(x,t) = \sum_{i=1}^{n-1} X_i(x) \cdot T_i(t) + X_n^p(x) \cdot T_n^{p-1}(t) \tag{70}$$

  where all functions but $X_n^p(x)$ are known.
  The simplest choice for the weight function u^* in (64) is

$$u^*(x,t) = X_n^*(x) \cdot T_n^{p-1}(t) \tag{71}$$

  which amounts to consider a Galerkin formulation of the diffusion problem.
  Introducing (70) and (71) into (64), we obtain

$$\begin{aligned}&\int\limits_{\Omega_x \times \Omega_t} X_n^* \cdot T_n^{p-1} \cdot \left(X_n^p \cdot \frac{\mathrm{d}T_n^{p-1}}{\mathrm{d}t} - k\, \frac{\mathrm{d}^2 X_n^p}{\mathrm{d}x^2} \cdot T_n^{p-1} \right) \mathrm{d}x\, \mathrm{d}t \\ &\quad = - \int\limits_{\Omega_x \times \Omega_t} X_n^* \cdot T_n^{p-1} \cdot \sum_{i=1}^{n-1} \left(X_i \cdot \frac{\mathrm{d}T_i}{\mathrm{d}t} - k\, \frac{\mathrm{d}^2 X_i}{\mathrm{d}x^2} \cdot T_i \right) \mathrm{d}x\, \mathrm{d}t + \int\limits_{\Omega_x \times \Omega_t} X_n^* \cdot T_n^{p-1} \cdot f \, \mathrm{d}x\, \mathrm{d}t\end{aligned} \tag{72}$$

  As all functions of time t are known, we can evaluate the following integrals:

$$\begin{cases} \alpha^x = \displaystyle\int\limits_{\Omega_t} \left(T_n^{p-1}(t) \right)^2 \mathrm{d}t \\ \beta^x = \displaystyle\int\limits_{\Omega_t} T_n^{p-1}(t) \cdot \frac{\mathrm{d}T_n^{p-1}(t)}{\mathrm{d}t} \mathrm{d}t \\ \gamma_i^x = \displaystyle\int\limits_{\Omega_t} T_n^{p-1}(t) \cdot T_i(t) \, \mathrm{d}t \\ \delta_i^x = \displaystyle\int\limits_{\Omega_t} T_n^{p-1}(t) \cdot \frac{\mathrm{d}T_i(t)}{\mathrm{d}t} \mathrm{d}t \\ \xi^x = \displaystyle\int\limits_{\Omega_t} T_n^{p-1}(t) \cdot f \, \mathrm{d}t \end{cases} \tag{73}$$

  Eq. (72) then takes the form

$$\int\limits_{\Omega_x} X_n^* \cdot \left(-k \cdot \alpha^x \cdot \frac{\mathrm{d}^2 X_n^p}{\mathrm{d}x^2} + \beta^x \cdot X_n^p \right) \mathrm{d}x = \int\limits_{\Omega_x} X_n^* \cdot \sum_{i=1}^{n-1} \left(k \cdot \gamma_i^x \cdot \frac{\mathrm{d}^2 X_i}{\mathrm{d}x^2} - \delta_i^x \cdot X_i \right) \mathrm{d}x \; + \int\limits_{\Omega_x} X_n^* \cdot \xi^x \, \mathrm{d}x \tag{74}$$

  This defines a one-dimensional boundary value problem (BVP), which is readily solved by means of a standard finite-element method to obtain an approximation of the function X_n^p. As another option, one can go back to the associated strong form

$$-k \cdot \alpha^x \cdot \frac{\mathrm{d}^2 X_n^p}{\mathrm{d}x^2} + \beta^x \cdot X_n^p \; = \sum_{i=1}^{n-1} \left(k \cdot \gamma_i^x \cdot \frac{\mathrm{d}^2 X_i}{\mathrm{d}x^2} - \delta_i^x \cdot X_i \right) \; + \; \xi^x \tag{75}$$

  and then solve it using any suitable numerical method, such as finite differences for example. The strong form (75) is a second-order differential equation for X_n^p due to the fact that the original diffusion equation (63) involves a second-order x-derivative of the unknown field u.
  The homogeneous Dirichlet boundary conditions $X_n^p(x=0) = X_n^p(x=L) = 0$ are readily specified with either weak or strong formulations.
- Calculating $T_n^p(t)$ from the just-computed $X_n^p(x)$.
  The procedure mirrors what we have just done. It suffices to exchange the roles played by the relevant functions of x and t.

The current PGD approximation reads

$$u^n(x,t) = \sum_{i=1}^{n-1} X_i(x) \cdot T_i(t) + X_n^p(x) \cdot T_n^p(t) \tag{76}$$

where all functions are known except $T_n^p(t)$.
With the Galerkin weight function

$$u^*(x,t) = X_n^p(x) \cdot T_n^*(t) \tag{77}$$

the weighted residual form (64) becomes

$$\begin{aligned} &\int\limits_{\Omega_x \times \Omega_t} X_n^p \cdot T_n^* \cdot \left(X_n^p \cdot \frac{\mathrm{d}T_n^p}{\mathrm{d}t} - k \frac{\mathrm{d}^2 X_n^p}{\mathrm{d}x^2} \cdot T_n^p \right) \mathrm{d}x\, \mathrm{d}t \\ &\quad = - \int\limits_{\Omega_x \times \Omega_t} X_n^p \cdot T_n^* \cdot \sum_{i=1}^{n-1} \left(X_i \cdot \frac{\mathrm{d}T_i}{\mathrm{d}t} - k \frac{\mathrm{d}^2 X_i}{\mathrm{d}x^2} \cdot T_i \right) \mathrm{d}x\, \mathrm{d}t + \int\limits_{\Omega_x \times \Omega_t} X_n^p \cdot T_n^* \cdot f \, \mathrm{d}x\, \mathrm{d}t \end{aligned} \tag{78}$$

Since all functions of x are known, we can perform the following integrals

$$\begin{cases} \alpha^t = \int\limits_{\Omega_x} \left(X_n^p(x) \right)^2 \, \mathrm{d}x \\ \beta^t = \int\limits_{\Omega_x} X_n^p(x) \cdot \frac{\mathrm{d}^2 X_n^p(x)}{\mathrm{d}x^2} \, \mathrm{d}x \\ \gamma_i^t = \int\limits_{\Omega_x} X_n^p(x) \cdot X_i(x) \, \mathrm{d}x \\ \delta_i^t = \int\limits_{\Omega_x} X_n^p(x) \cdot \frac{\mathrm{d}^2 X_i(x)}{\mathrm{d}x^2} \, \mathrm{d}x \\ \xi^t = \int\limits_{\Omega_x} X_n^p(x) \cdot f \, \mathrm{d}x \end{cases} \tag{79}$$

Eq. (78) then becomes

$$\int\limits_{\Omega_t} T_n^* \cdot \left(\alpha^t \cdot \frac{\mathrm{d}T_n^p}{\mathrm{d}t} - k \cdot \beta^t \cdot T_n^p \right) \mathrm{d}t = \int\limits_{\Omega_t} T_n^* \cdot \sum_{i=1}^{n-1} \left(-\gamma_i^t \cdot \frac{\mathrm{d}T_i}{\mathrm{d}t} + k \cdot \delta_i^t \cdot T_i \right) \mathrm{d}t \; + \int\limits_{\Omega_t} T_n^* \cdot \xi^t \, \mathrm{d}t \tag{80}$$

We have thus obtained an initial value problem (IVP) for the function T_n^p. The weighted residual form (80) can be solved by means of any stabilized finite-element scheme (e.g., discontinuous Galerkin's scheme). The associated strong form reads:

$$\alpha^t \cdot \frac{\mathrm{d}T_n^p}{\mathrm{d}t} - k \cdot \beta^t \cdot T_n^p \; = \sum_{i=1}^{n-1} \left(-\gamma_i^t \cdot \frac{\mathrm{d}T_i}{\mathrm{d}t} + k \cdot \delta_i^t \cdot T_i \right) \; + \; \xi^t \tag{81}$$

Since the original diffusion equation involves a first-order derivative of u with respect to t, we have thus obtained a first-order ordinary differential equation for T_n^p. Any classical numerical technique can be used to solve it. The initial condition $T_n^p(t = 0) = 0$ is readily specified with either weak or strong form.

References

[1] H.M. Park, D.H. Cho, The use of the Karhunen–Loève decomposition for the modelling of distributed parameter systems, Chem. Eng. Sci. 51 (1996) 81–98.

[2] Y. Maday, E.M. Ronquist, The reduced basis element method: application to a thermal fin problem, SIAM J. Sci. Comput. 26 (1) (2004) 240–258.

[3] R.A. Bialecki, A.J. Kassab, A. Fic, Proper orthogonal decomposition and modal analysis for acceleration of transient FEM thermal analysis, Int. J. Numer. Methods Eng. 62 (2005) 774–797.

[4] J. Burkardt, M. Gunzburger, H.-Ch. Lee, POD and CVT-based reduced-order modeling of Navier–Stokes flows, Comput. Methods Appl. Mech. Eng. 196 (2006) 337–355.

[5] M.D. Gunzburger, J.S. Peterson, J.N. Shadid, Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data, Comput. Methods Appl. Mech. Eng. 196 (2007) 1030–1047.

[6] A. Ammar, D. Ryckelynck, F. Chinesta, R. Keunings, On the reduction of kinetic theory models related to finitely extensible dumbbells, J. Non-Newton. Fluid Mech. 134 (2006) 136–147.

[7] S. Niroomandi, I. Alfaro, E. Cueto, F. Chinesta, Real-time deformable models of non-linear tissues by model reduction techniques, Comput. Methods Programs Biomed. 91 (2008) 223–231.

[8] S. Niroomandi, I. Alfaro, E. Cueto, F. Chinesta, Model order reduction for hyperelastic materials, Int. J. Numer. Methods Biomed. Eng. 81 (9) (2010) 1180–1206.

[9] S. Niroomandi, I. Alfaro, E. Cueto, F. Chinesta, Accounting for large deformations in real-time simulations of soft tissues based on reduced order models, Comput. Methods Programs Biomed. 105 (2012) 1–12.

[10] S. Niroomandi, I. Alfaro, D. Gonzalez, E. Cueto, F. Chinesta, Real time simulation of surgery by reduced order modelling and X-FEM techniques, Int. J. Numer. Methods Biomed. Eng. 28 (5) (2012) 574–588.

[11] A. Ammar, E. Pruliere, F. Chinesta, M. Laso, Reduced numerical modeling of flows involving liquid–crystalline polymeres, J. Non-Newton. Fluid Mech. 160 (2009) 140–156.

[12] F. Schmidt, N. Pirc, M. Mongeau, F. Chinesta, Efficient mould cooling optimization by using model reduction, Int. J. Material Form. 4 (1) (2011) 71–82.

[13] D. Ryckelynck, L. Hermanns, F. Chinesta, E. Alarcon, An efficient a priori model reduction for boundary element models, Eng. Anal. Bound. Elem. 29 (2005) 796–801.

[14] D. Ryckelynck, F. Chinesta, E. Cueto, A. Ammar, On the a priori model reduction: overview and recent developments, Arch. Comput. Methods Eng. 13 (1) (2006) 91–128.

[15] Y. Maday, E.M. Ronquist, A reduced-basis element method, C. R. Acad. Sci. Paris, Ser. I 335 (2002) 195–200.

[16] Y. Maday, A.T. Patera, G. Turinici, A priori convergence theory for reduced-basis approximations of single-parametric elliptic partial differential equations, J. Sci. Comput. 17 (1–4) (2002) 437–446.

[17] K. Veroy, A. Patera, Certified real-time solution of the parametrized steady incompressible Navier–Stokes equations: rigorous reduced-basis a posteriori error bounds, Int. J. Numer. Methods Fluids 47 (2005) 773–788.

[18] G. Rozza, D.B.P. Huynh, A.T. Patera, Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations—application to transport and continuum mechanics, Arch. Comput. Methods Eng. 15 (3) (2008) 229–275.

[19] P. Ladevèze, The large time increment method for the analyze of structures with nonlinear constitutive relation described by internal variables, C. R. Acad. Sci. Paris, Ser. I 309 (1989) 1095–1099.

[20] P. Ladevèze, A. Nouy, A multiscale computational method with time and space homogenization, C. R. Mecanique 330 (10) (2002) 683–689.

[21] P. Ladevèze, A. Nouy, O. Loiseau, A multiscale computational approach for contact problems, Comput. Methods Appl. Mech. Eng. 191 (43) (2002) 4869–4891.

[22] P. Ladevèze, A. Nouy, On a multiscale computational strategy with time and space homogenization for structural mechanics, Comput. Methods Appl. Mech. Eng. 192 (28–30) (2003) 3061–3087.

[23] P. Ladevèze, D. Néron, P. Gosselet, On a mixed and multiscale domain decomposition method, Comput. Methods Appl. Mech. Eng. 96 (2007) 1526–1540.

[24] P. Ladevèze, J.-C. Passieux, D. Néron, The Latin multiscale computational method and the proper generalized decomposition, Comput. Methods Appl. Mech. Eng. 199 (21–22) (2010) 1287–1296.

[25] 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 (2006) 153–176.

[26] 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. Part II: Transient simulation using space-time separated representation, J. Non-Newton. Fluid Mech. 144 (2007) 98–121.

[27] A. Ammar, F. Chinesta, 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.

[28] A. Ammar, E. Cueto, F. Chinesta, Reduction of the chemical master equation for gene regulatory networks using proper generalized decompositions, Int. J. Numer. Methods Biomed. Eng. 28 (9) (2012) 960–973.

[29] H. Lamari, A. Ammar, A. Leygue, F. Chinesta, On the solution of the multidimensional Langer's equation by using the proper generalized decomposition method for modeling phase transitions, Model. Simul. Mater. Sci. Eng. 20 (1) (2012) 015007.

[30] A. Ammar, M. Normandin, F. Chinesta, Solving parametric complex fluids models in rheometric flows, J. Non-Newton. Fluid Mech. 165 (2010) 1588–1601.

[31] E. Pruliere, F. Chinesta, A. Ammar, On the deterministic solution of multidimensional parametric models by using the proper generalized decomposition, Math. Comput. Simul. 81 (2010) 791–810.

[32] Ch. Ghnatios, F. Chinesta, E. Cueto, A. Leygue, P. Breitkopf, P. Villon, Methodological approach to efficient modeling and optimization of thermal processes taking place in a die: application to pultrusion, Composites, Part A, Appl. Sci. Manuf. 42 (2011) 1169–1178.

[33] Ch. Ghnatios, F. Masson, A. Huerta, E. Cueto, A. Leygue, F. Chinesta, Proper generalized decomposition based dynamic data-driven control of thermal processes, Comput. Methods Appl. Mech. Eng. 213 (2012) 29–41.

[34] D. Gonzalez, F. Masson, F. Poulhaon, A. Leygue, E. Cueto, F. Chinesta, Proper generalized decomposition based dynamic data-driven inverse identification, Math. Comput. Simul. 82 (9) (2012) 1677–1695.

[35] A. Ammar, E. Cueto, F. Chinesta, Non-incremental PGD solution of parametric uncoupled models defined in evolving domains, Int. J. Numer. Methods Eng. 93 (8) (2013) 887–904.

[36] F. Chinesta, A. Ammar, E. Cueto, Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models, Arch. Comput. Methods Eng. 17 (4) (2010) 327–350.

[37] F. Chinesta, A. Ammar, A. Leygue, R. Keunings, An overview of the proper generalized decomposition with applications in computational rheology, J. Non-Newton. Fluid Mech. 166 (2011) 578–592.

[38] F. Chinesta, P. Ladeveze, E. Cueto, A short review in model order reduction based on proper generalized decomposition, Arch. Comput. Methods Eng. 18 (2011) 395–404.

[39] F. Chinesta, A. Leygue, F. Bordeu, J.V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, A. Huerta, Parametric PGD based computational vademecum for efficient design, optimization and control, Arch. Comput. Methods Eng. 20 (1) (2013) 31–59.

[40] A. Ammar, M. Normandin, F. Daim, D. Gonzalez, E. Cueto, F. Chinesta, Non-incremental strategies based on separated representations: applications in computational rheology, Commun. Math. Sci. 8 (3) (2010) 671–695.

[41] F. Chinesta, A. Ammar, E. Cueto, Proper generalized decomposition of multiscale models, Int. J. Numer. Methods Biomed. Eng. 83 (8–9) (2010) 1114–1132.

[42] A. Ammar, F. Chinesta, E. Cueto, M. Doblare, Proper generalized decomposition of time-multiscale models, Int. J. Numer. Methods Biomed. Eng. 90 (5) (2012) 569–596.

[43] H. Lamari, A. Ammar, P. Cartraud, G. Legrain, F. Jacquemin, F. Chinesta, Routes for efficient computational homogenization of non-linear materials using the proper generalized decomposition, Arch. Comput. Methods Eng. 17 (4) (2010) 373–391.

[44] B. Bognet, A. Leygue, F. Chinesta, A. Poitou, F. Bordeu, Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity, Comput. Methods Appl. Mech. Eng. 201 (2012) 1–12.

[45] F. Chinesta, A. Leygue, B. Bognet, Ch. Ghnatios, F. Poulhaon, F. Bordeu, A. Barasinski, A. Poitou, S. Chatel, S. Maison-Le-Poec, First steps towards an advanced simulation of composites manufacturing by automated tape placement, Int. J. Material Form., http://dx.doi.org/10.1007/s12289-012-1112-9.

[46] A. Leygue, F. Chinesta, M. Beringhier, T.L. Nguyen, J.C. Grandidier, F. Pasavento, B. Schrefler, Towards a framework for non-linear thermal models in shell domains, Int. J. Numer. Methods Heat Fluid Flow 23 (1) (2013) 55–73.

[47] B. Bognet, A. Leygue, F. Chinesta, Separated representations of 3D elastic solutions in shell geometries, Adv. Model. Simul. Eng. Sci. 1 (2014) 4, www.amses-journal.com/content/1/1/4.

[48] D. Gonzalez, A. Ammar, F. Chinesta, E. Cueto, Recent advances in the use of separated representations, Int. J. Numer. Methods Biomed. Eng. 81 (5) (2010) 637–659.

[49] C. Ghnatios, G. Xu, M. Visonneau, A. Leygue, F. Chinesta, On the space separated representation when addressing the solution of PDE in complex domains, AIMS J., submitted for publication.

[50] F. Chinesta, R. Keunings, A. Leygue, The proper generalized decomposition for advanced numerical simulations. A primer, Springer briefs, Springer, 2014.

[51] A. Ammar, F. Chinesta, P. Diez, A. Huerta, An error estimator for separated representations of highly multidimensional models, Comput. Methods Appl. Mech. Eng. 199 (2010) 1872–1880.

[52] P. Ladevèze, L. Chamoin, On the verification of model reduction methods based on the proper generalized decomposition, Comput. Methods Appl. Mech. Eng. 200 (2011) 2032–2047.