Non-incremental boundary element discretization of parabolic models based on the use of the proper generalized decompositions

Gaël Bonithon, Pierre Joyot, Francisco Chinesta, Pierre Villon

▶ To cite this version:

Gaël Bonithon, Pierre Joyot, Francisco Chinesta, Pierre Villon. Non-incremental boundary element discretization of parabolic models based on the use of the proper generalized decompositions. Engineering Analysis with Boundary Elements, 2011, 35, pp.2-17. 10.1016/j.enganabound.2010.07.007 . hal-01532805

HAL Id: hal-01532805
https://hal.science/hal-01532805

Submitted on 3 Jun 2017

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

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution| 4.0 International License

Non-incremental boundary element discretization of parabolic models based on the use of the proper generalized decompositions

G. Bonithon [a,d], P. Joyot [a], F. Chinesta [b], P. Villon [c]

[a] ESTIA-Recherche, technopole izarbel, 64210 Bidart, France
[b] EADS Corporate Foundation International Chair, GEM CNRS-ECN, 1 rue de la Noë BP 92101, 44321 Nantes cedex 3, France
[c] UTC-Roberval, UMR 6253, 60200 Compiègne, France
[d] EPSILON Ingénierie, Imm. Technoparc 10, 10 rue Jean Bart, BP 97431, 31674 Labège cedex, France

There are many ways to solve space–time linear parabolic partial differential equations by using the boundary element method (BEM). In general, standard techniques make use of an incremental strategy. In this paper we propose a novel alternative of efficient non-incremental solution strategy for that kind of models. The proposed technique combines the use of the BEM with a proper generalized decomposition (PGD) that allows a space–time separated representation of the unknown field within a non-incremental integration scheme.

Keywords:
Proper generalized decomposition
BEM
Heat equation

1. Introduction

The boundary element method (BEM) allows efficient solution of partial differential equations whose kernel functions are known. The heat equation is one of these candidates when the thermal parameters are assumed constant (linear model). When the model involves large physical domains and time simulation intervals the amount of information that must be stored increases significantly. This drawback can be circumvented by using advanced strategies, as for example the multipoles technique [1]. Another possibility is to employ the LTBEM approach. The basic idea behind this method is to adopt the Laplace transform first to convert a time-dependent parabolic differential system into a time-independent elliptical boundary problem in the Laplace domain. However, once the differential system is solved in transformed space, the reconstruction of the solution in the time domain requires the application of the inverse Laplace transform. The solution accuracy depends on the choice of the inverse Laplace transform. Sutradhar [2] and Amado [3] provide recent results related to this method.

We propose in this paper an alternative radically different approach that leads to a separated solution of the space and time problems within a non-incremental integration strategy. The technique is based on the use of a space–time separated representation of the unknown field that, introduced in the residual weighting formulation, allows to define a separated solution of the resulting weak form. The spatial step can be then treated by invoking the standard BEM for solving the resulting steady state problem defined in the physical space. Then, the time problem that results in an ordinary first order differential equation is solved using any standard appropriate integration technique (e.g. backward finite differences).

In the case of the linear and transient heat equation here considered for the sake of simplicity, the PGD (proper generalized decomposition) leads to the solution of a series of steady state diffusion–reaction problems (accurately solved by using the BEM method) and a series of problems that consist of a simple time dependent ODE.

Separated representations were already applied for solving transient models in the context of finite element discretizations [4–10], but they never have been used in the BEM framework, and certainly in this context the main advantage is the possibility of defining non-incremental strategies as well as the possibility of avoiding the use of space–time kernels. In principle, this technique seems specially adapted for solving transient problems involving extremely small time steps.

We start summarizing the main ideas of the Proper Generalized Decomposition for the solution of linear parabolic problems. In the next section we will focus on the application of such technique in the context of the BEM. Finally, some numerical examples will be presented and discussed.

1.1. Motivating the use of separated space–time representations

Let $u(\mathbf{x}, t)$ be the solution of a certain transient model (in what follows $\mathbf{x} \in \Omega \subset \mathbb{R}^d$, $d=1,2,3$, and $t \in \mathcal{I} \subset \mathbb{R}^+$). We also assume that this field is known in a discrete manner, that is, at some points $\mathbf{x}_i$

(the nodes of a mesh or a grid) and at certain times t^p, where $i \in [1,\ldots,N_n]$ and $p \in [1,\ldots,P]$.

Now, we introduce the notation $u_i^p \equiv u(\mathbf{x}_i,t^p)$ and construct the matrix $\mathbf{Q}$ that contains the snapshots:

$$\mathbf{Q} = \begin{pmatrix} u_1^1 & u_1^2 & \cdots & u_1^P \\ u_2^1 & u_2^2 & \cdots & u_2^P \\ \vdots & \vdots & \ddots & \vdots \\ u_{N_n}^1 & u_{N_n}^2 & \cdots & u_{N_n}^P \end{pmatrix}. \tag{1}$$

The proper orthogonal decomposition (POD) of this discrete field consists in solving the eigenvalue problem:

$$(\mathbf{Q}\mathbf{Q}^T)\boldsymbol{\phi} = \lambda\boldsymbol{\phi} \tag{2}$$

that results in N_n coupled eigenvalue-eigenvector $(\lambda_i,\boldsymbol{\phi}_i)$, $i=1,\ldots,N_n$.

When the field evolves smoothly, the magnitude of the eigenvalues decreases very fast, fact that reveals that the evolution of the field can be approximated from a reduced number of modes (eigenvectors). Thus, if we define a cutoff value ε ($\varepsilon = 10^{-8} \times \lambda_1$ in practice, λ_1 being the highest eigenvalue) only a reduced number of modes are retained. Let R ($R \ll N_n$) be the number of modes retained, i.e. $\lambda_i \geq 10^{-8} \times \lambda_1$, $i = 1,\ldots,R$ and $\lambda_i < 10^{-8} \times \lambda_1, \forall i > R$ (the eigenvalues are assumed ordered). Thus, one could write:

$$u(\mathbf{x},t) \approx \sum_{i=1}^{i=R} \phi_i(\mathbf{x}) \cdot T_i(t) \equiv \sum_{i=1}^{i=R} X_i(\mathbf{x}) \cdot T_i(t), \tag{3}$$

where for the sake of clarity the space modes $\phi_i(\mathbf{x})$ will, from now on, be denoted as $X_i(\mathbf{x})$. Eq. (3) represents a natural separated representation (also known as finite sums decomposition).

These modes could now be used to solve other "similar" problems, that is, models involving slight changes in the boundary conditions, model parameters … [11–13]. In [13] the use of reduced basis was successfully combined with a BEM discretization. Other possibility is computing the reduced basis from the standard transient solution within a short time interval (with respect to the whole time interval in which the model is defined) and then solve the remaining part of the time interval by employing the reduced basis. Obviously, both strategies induce the introduction of an error whose evaluation, control and reduction is a challenging issue.

One possibility to construct an adaptive reduced approximation basis, that should be the best reduced approximation basis for the treated problem, consists in alternating a reduction step (based on the application of the proper orthogonal decomposition) and an enrichment stage to improve the quality of the reduced approximation basis in order to capture all the solution features. We proposed recently an enrichment technique based on the use of some Krylov's subspaces generated by the equation residual. This technique known as "a priori" model reduction was originally proposed in [14], widely described in [15] and successfully applied for solving complex fluid flows within the kinetic theory framework [16,7] and for speeding up boundary element discretization [13] and thermomechanical simulations [17]. However, some difficulties were noticed in the application of this strategy: (i) the enrichment based on the use of Krylov's subspaces is far to be optimal in a variety of models (e.g. the wave equation) and (ii) the incremental nature of the algorithm; … .

From the previous analysis we can conclude: (i) the transient solution of numerous models can be expressed using a very reduced number of products each one involving a function of time and a function of space and (ii) the functions involved in these functional products can be determined simultaneously by applying an appropriate algorithm.

In what follows we describe a possible strategy able to compute these separated functional couples. From our numerical experiments we noticed that this technique converged in the different models until now analyzed, and that the final decomposition is not so far to the one that results from the application of the proper orthogonal decomposition on the model solution, that is, the number of functional couples was quite similar, being the optimal decomposition of the one performed by applying the POD. However, at present we cannot prove this empirical observation for general models.

2. On the proper generalized decomposition: a survey

Some models encountered in science and engineering are sometimes defined in multidimensional spaces (as the ones involved in quantum mechanics or kinetic theory descriptions of materials, including complex fluids) that exhibit the terrific curse of dimensionality when usual mesh-based discretization techniques are applied. Other times, models involve transient fields, that even when they are defined in three-dimensional physical spaces, they must be solved in large time intervals using very small time steps.

In the first kind of models the difficulty is quite natural and the solution of such models needs new strategies. One possibility lies in the use of sparse grids [18]. However, as argued in [19], the use of sparse grid is restricted to models with moderate multidimensionality (up to 20). Another technique able to circumvent, or at least alleviate, the curse of dimensionality consists of using a separated representation of the unknown field (see [20,21] for some mathematical results on this topic). Basically, the separated representation of a generic function $u(x_1,\ldots,x_D)$ (also known as finite sums decomposition) writes:

$$u(x_1,\ldots,x_D) \approx \sum_{i=1}^{i=N} F_1^i(x_1) \times \cdots \times F_D^i(x_D). \tag{4}$$

This kind of representation is not new, it was widely employed in the last decades in the framework of quantum chemistry. In particular the Hartree–Fock (that involves a single product of functions) and post-Hartree–Fock approaches (as the MCSCF that involves a finite number of sums) made use of a separated representation of the wavefunction [22].

We proposed recently a technique able to construct, in a way completely transparent for the user, the separated representation of the unknown field involved in a partial differential equation. This technique, originally described and applied to multi-bead-spring FENE models of polymeric systems in [4], was extended to transient models of such complex fluids in [5]. Other more complex models (involving different couplings and non-linearities) based on the reptation theory of polymeric liquids were analyzed in [6] and in [7] separated representations were applied in the stochastic framework. An introductive overview on the application of separated representations in the multi-scale modeling of materials can be found in [8].

The present work will focus on the second kind of models described at the beginning of this section. For the sake of simplicity in what follows we consider models defined in moderate dimensions (d D, $d=1$, 2, 3, physical or conformation spaces) but whose solutions evolve in large time intervals. In this context, if one uses standard incremental time-discretizations, in the general case (models involving time-dependent parameters, non-linear models, …), 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, the simulation becomes inefficient. To illustrate this scenario, one could imagine the simple reaction–diffusion model that describes the degradation of plastic materials, where the characteristic time of the chemical reaction involved in the material degradation is of some microseconds and the one related to the diffusion of chemical substances (that also represents the material degradation characteristic time itself) is of the order of years. In this case standard incremental techniques must be replaced by other more efficient techniques.

Pierre Ladeveze proposed several years ago a powerful technique for addressing this kind of challenging models that he called the LATIN method [23]. The LATIN method integrates many ingredients leading to a robust, powerful, efficient and accurate discretization technique especially well adapted for treating transient multi-scale non-linear models. The two most outstanding ingredients are (i) the decoupling between a linear-global problem and a non-linear-local one, both defined in the whole space–time domain and (ii) a space–time separated representation of the model variables in order to accelerate the solution of the linear-global problem. The former separated representation was called by Ladeveze in the 1980s "radial approximation", and in our knowledge it was the first time that separated representations were applied in computational mechanics. The interested reader can refer to [24] and the references therein for a recent overview and the state of the art of the LATIN approach in multi-scale modeling.

2.1. Illustrating the discretization based on separated representations

In this section, we illustrate the discretization of partial differential equations using a separated representation of the unknown field.

Let us consider the generic partial differential equation

$$\mathcal{L}(u(\mathbf{x},t)) = f(\mathbf{x},t) \quad \text{in } \Omega \times \mathcal{I} = \Omega \times (0,T_{max}],$$

with homogeneous initial and boundary conditions, where $\Omega \subset \mathbb{R}^d, d \geq 1,\ T_{max} > 0$. The aim of the separated representation method is to compute N couples of functions $\{(X_i, T_i)\}_{i=1,\dots,N}$ such that $\{X_i\}_{i=1,\dots,N}$ and $\{T_i\}_{i=1,\dots,N}$ are defined, respectively, in Ω and $(0,T_{max}]$ and the solution u of this problem can be written in the separate form

$$u(\mathbf{x},t) \approx \sum_{i=1}^{N} T_i(t) \cdot X_i(\mathbf{x}). \tag{5}$$

We are illustrating the strategy for constructing these functional products in the case of an academic transient problem, the transient linear heat equation:

$$\frac{\partial u}{\partial t} - a\Delta u = f(\mathbf{x},t) \quad \text{in } \Omega \times (0,T_{max}], \tag{6}$$

with homogeneous initial and boundary conditions, where a is the diffusion coefficient. The weak formulation yields:

Find $u(\mathbf{x},t)$ such that

$$\int_0^{T_{max}} \int_\Omega u^\star \left(\frac{\partial u}{\partial t} - a\Delta u - f(\mathbf{x},t) \right) d\mathbf{x}\, dt = 0 \tag{7}$$

for all the functions $u^\star$ in an appropriate functional space.

We compute now the functions involved in the sum (5). We suppose that the set of functional couples $\{(X_i, T_i)\}_{i=1,\dots,n}$ with $0 \leq n < N$ are already known (they have been previously computed) and that at the present iteration we search the enrichment couple $(R(t), S(\mathbf{x}))$ by applying an alternating direction fixed point algorithm that after convergence will constitute the next functional couple (X_{n+1}, T_{n+1}). Hence, at the present iteration, n, we

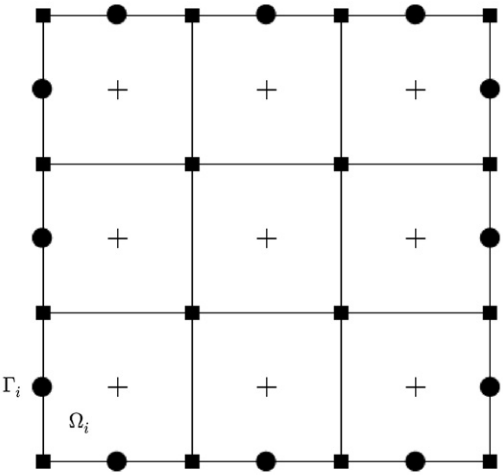


Fig. 1. Colocation nodes (•), MLS nodes (■) and integration points (+).

Table 1
Error $\overline{e}_n = 10^2 \times e_n$ as a function of the functional couples considered to define the approximation of the unknown function $u(\mathbf{x},t)$ and the discretization employed ($n_\tau = 256$).

n_Γ	4	8	16	32	64
$\overline{e}_1$	9.9640	8.9397	8.7820	8.7442	8.7519
$\overline{e}_2$	5.7327	3.4744	3.5755	3.5846	3.5515
$\overline{e}_3$	4.9489	1.7439	2.0731	2.1270	2.1819
$\overline{e}_4$	5.9839	2.2748	2.2101	2.1536	2.1793
$\overline{e}_5$	5.9498	1.1472	1.2022	1.2752	1.3111
$\overline{e}_6$	5.9477	1.7641	1.4213	1.3425	1.3634
$\overline{e}_7$	5.9476	1.3769	0.5473	0.4681	0.4195
$\overline{e}_8$	5.9476	1.5295	0.5433	0.4505	0.3590
$\overline{e}_9$	5.9474	1.4631	0.4443	0.2992	0.2761

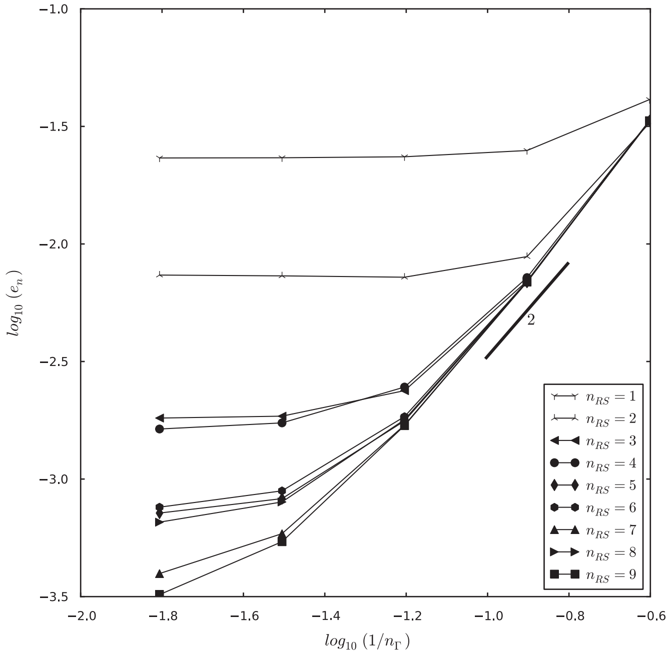


Fig. 2. Evolution of the error e_n versus the space discretization for different levels of approximation n.

assume the separated representation

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

The weighting function $u^\star$ is then assumed as

$$u^\star = S \cdot R^\star + R \cdot S^\star. \tag{9}$$

Introducing (8) and (9) into (7) it results

$$\int_0^{T_{max}} \int_\Omega (S \cdot R^\star + R \cdot S^\star) \cdot \left(S \cdot \frac{\partial R}{\partial t} - a \Delta S \cdot R \right) d\mathbf{x}\, dt$$
$$= \int_0^{T_{max}} \int_\Omega (S \cdot R^\star + R \cdot S^\star) \cdot \quad f(\mathbf{x},t) - \sum_{i=1}^{n} X_i \cdot \frac{\partial T_i}{\partial t} + a \sum_{i=1}^{n} \Delta X_i \cdot T_i \Bigg) d\mathbf{x}\, dt. \tag{10}$$

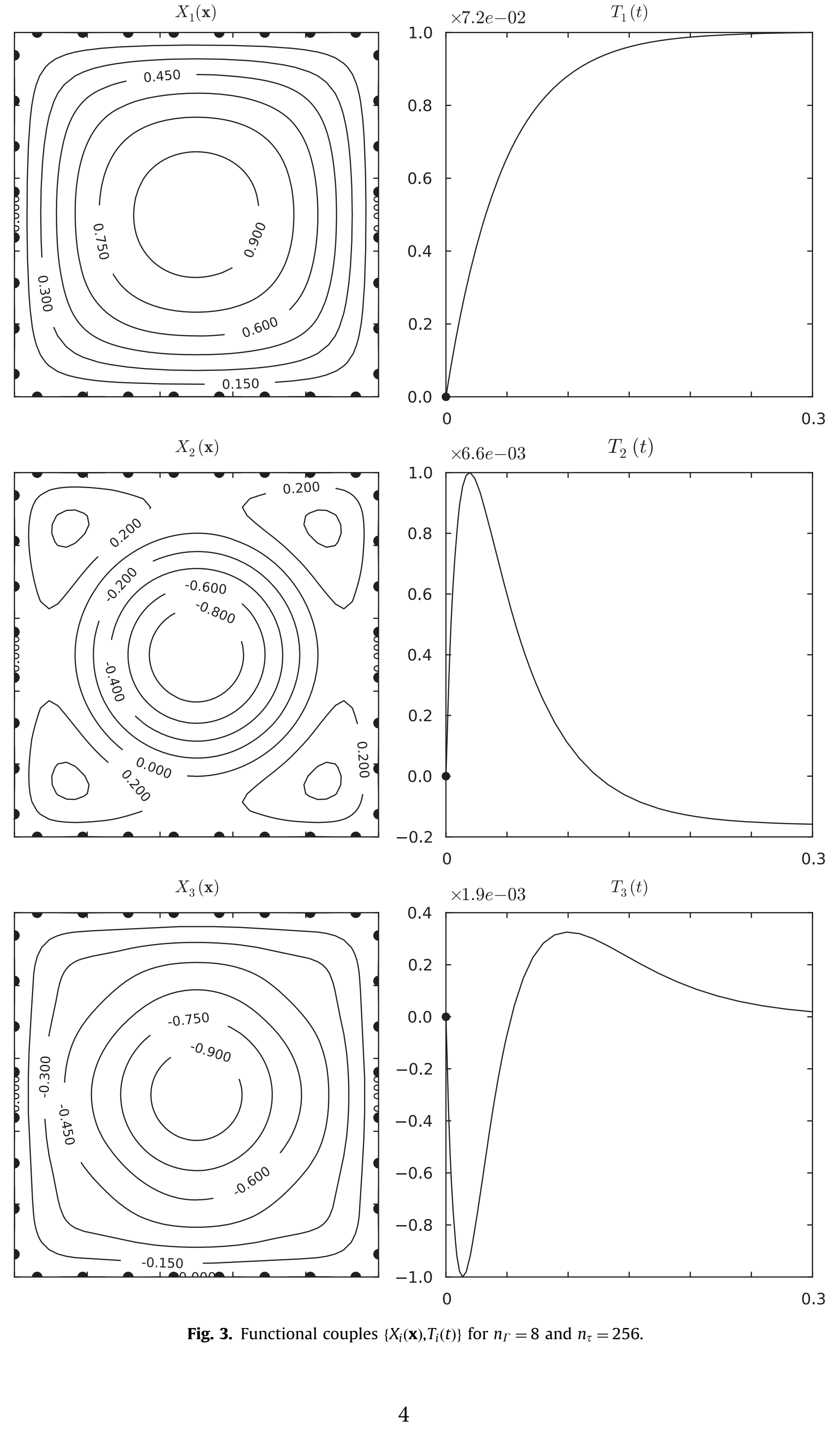


Fig. 3. Functional couples $\{X_i(\mathbf{x}),T_i(t)\}$ for $n_\Gamma = 8$ and $n_\tau = 256$.

We apply an alternating direction fixed point algorithm to compute the couple of functions (R, S):

- *Computing the function* $S(\mathbf{x})$:
  First, we suppose that R is known, implying that $R^\star$ vanishes in (9). Thus, Eq. (10) writes

$$\int_\Omega S^\star \cdot (\alpha_t S - a\beta_t \Delta S)\, d\mathbf{x} = \int_\Omega S^\star \cdot \left. \gamma_t(\mathbf{x}) - \sum_{i=1}^{n} \alpha_t^i X_i + a \sum_{i=1}^{n} \beta_t^i \Delta X_i \right) d\mathbf{x}, \tag{11}$$

  where

$$\begin{cases} \alpha_t = \int_0^{T_{max}} R(t) \cdot \dfrac{\partial R}{\partial t}(t)\, dt, \\ \alpha_t^i = \int_0^{T_{max}} R(t) \cdot \dfrac{\partial T_i}{\partial t}(t)\, dt, \\ \beta_t = \int_0^{T_{max}} R^2(t)\, dt, \\ \beta_t^i = \int_0^{T_{max}} R(t) \cdot T_i(t)\, dt, \\ \gamma_t(\mathbf{x}) = \int_0^{T_{max}} R(t) \cdot f(\mathbf{x},t)\, dt; \quad \forall \mathbf{x} \in \Omega. \end{cases} \tag{12}$$

  The weak formulation (11) is satisfied for all $S^\star$, therefore, we could come back to the associated strong formulation

$$\alpha_t S - a\beta_t \Delta S = \gamma_t - \sum_{i=1}^{n} \alpha_t^i X_i + a \sum_{i=1}^{n} \beta_t^i \Delta X_i \tag{13}$$

  that one could solve by using any appropriate discretization technique for computing the space function $S(\mathbf{x})$.
- *Computing the function* $R(t)$:
  From the function $S(\mathbf{x})$ just computed, we search $R(t)$. In this case $S^\star$ vanishes in (9) and Eq. (10) reduces to

$$\int_0^{T_{max}} \int_\Omega (S \cdot R^\star) \cdot \left(S \cdot \frac{\partial R}{\partial t} - a\Delta S \cdot R\right) d\mathbf{x}\, dt = \int_0^{T_{max}} \int_\Omega (S \cdot R^\star) \cdot \left. f(x,t) - \sum_{i=1}^{n} X_i \cdot \frac{\partial T_i}{\partial t} + a \sum_{i=1}^{n} \Delta X_i \cdot T_i \right) d\mathbf{x}\, dt, \tag{14}$$

  where all the spatial functions can be integrated in Ω. Thus, by using the following notations

$$\begin{cases} \alpha_x = \int_\Omega S(\mathbf{x}) \cdot \Delta S(\mathbf{x})\, d\mathbf{x}, \\ \alpha_x^i = \int_\Omega S(\mathbf{x}) \cdot \Delta X_i(\mathbf{x})\, d\mathbf{x}, \\ \beta_x = \int_\Omega S^2(\mathbf{x})\, d\mathbf{x}, \\ \beta_x^i = \int_\Omega S(\mathbf{x}) \cdot X_i(\mathbf{x})\, d\mathbf{x}, \\ \gamma_x(t) = \int_\Omega S(\mathbf{x}) \cdot f(\mathbf{x},t)\, d\mathbf{x}; \quad \forall t. \end{cases} \tag{15}$$

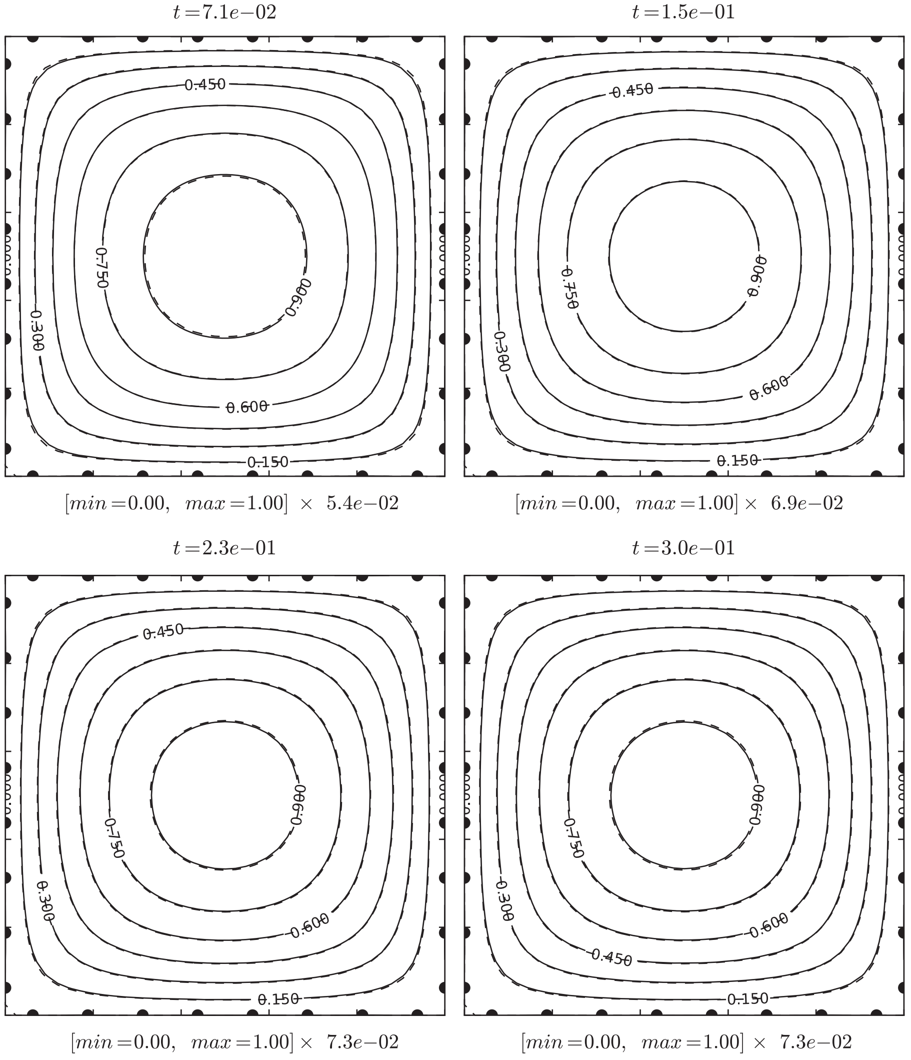


Fig. 4. $u(\mathbf{x},t)$ and $u^{ref}(\mathbf{x},t)$ (dashed line) for $t = \{0.071s, 0.15s, 0.23s, 0.3s\}$, $n_F = 8$ and $n_\tau = 256$.

Eq. (14) reads

$$\int_0^{T_{max}} R^\star \cdot \quad \beta_x \frac{\partial R}{\partial t} - a\alpha_x R - \gamma_x(t) + \sum_{i=1}^{n} \beta_x^i \frac{\partial T_i}{\partial t} - \sum_{i=1}^{n} a\alpha_x^i \cdot T_i \Bigg) dt = 0. \tag{16}$$

As Eq. (16) holds for all $S^\star$, we could come back to the strong formulation

$$\beta_x \frac{\partial R}{\partial t} = a \cdot \alpha_x \cdot R + \gamma_x(t) - \sum_{i=1}^{n} \beta_x^i \cdot \frac{\partial T_i}{\partial t} + \sum_{i=1}^{n} a \cdot \alpha_x^i \cdot T_i, \tag{17}$$

which is a first order ordinary differential equation that can be solved easily (even for extremely small times steps) from its initial condition.

These two steps must be repeated until convergence, that is, until verifying that both functions reach a fixed point. If we denote by $R^{(q)}(t)$ and $R^{(q-1)}(t)$ the computed functions $R(t)$ at the present and previous iteration, respectively, and the same for the space functions: $S^{(q)}(\mathbf{x})$ and $S^{(q-1)}(\mathbf{x})$, the stopping criterion used in this work writes:

$$e = \|R^{(q)}(t) \cdot S^{(q)}(\mathbf{x}) - R^{(q-1)}(t) \cdot S^{(q-1)}(\mathbf{x})\|_2 < 10^{-8}, \tag{18}$$

where 10^{-8} represents the square root of the machine precision.

We denote by Q_{n+1} the number of iterations for solving this non-linear problem to determine the enrichment couple of functions $X_{n+1}(\mathbf{x})$ and $T_{n+1}(t)$. After reaching convergence we write $X_{n+1}(\mathbf{x}) = S(\mathbf{x})$ and $T_{n+1}(t) = R(t)$. The enrichment procedure must continue until reaching the convergence of the enrichment global procedure at iteration N, when the separated representation of the unknown field writes:

$$u(\mathbf{x},t) \approx \sum_{i=1}^{N} X_i(\mathbf{x}) \cdot T_i(t). \tag{19}$$

In our numerical solutions the global stopping criterion was:

- For models whose exact solution u^{ref} was known:
$$E = \frac{\|u - u^{ref}\|_2}{\|u^{ref}\|_2} < \varepsilon. \tag{20}$$
- For models whose exact solution was not known:
$$E = \frac{\left\| \frac{\partial u}{\partial t} - a\Delta u - f(\mathbf{x},t) \right\|_2}{\|f(\mathbf{x},t)\|_2} < \varepsilon, \tag{21}$$

with $\varepsilon = 10^{-6}$ in all the simulations reported in this work.

Discussion: The just proposed strategy needs for the solution of about $N \times Q$ space and time problems (with $Q = (Q_1 + \cdots + Q_N/N$ and N the number of functional couples needed to approximate, up to the desired precision, the searched solution). Thus, one must compute $N \times Q$ dD problems, $d = 1, 2, 3$, whose complexity depends

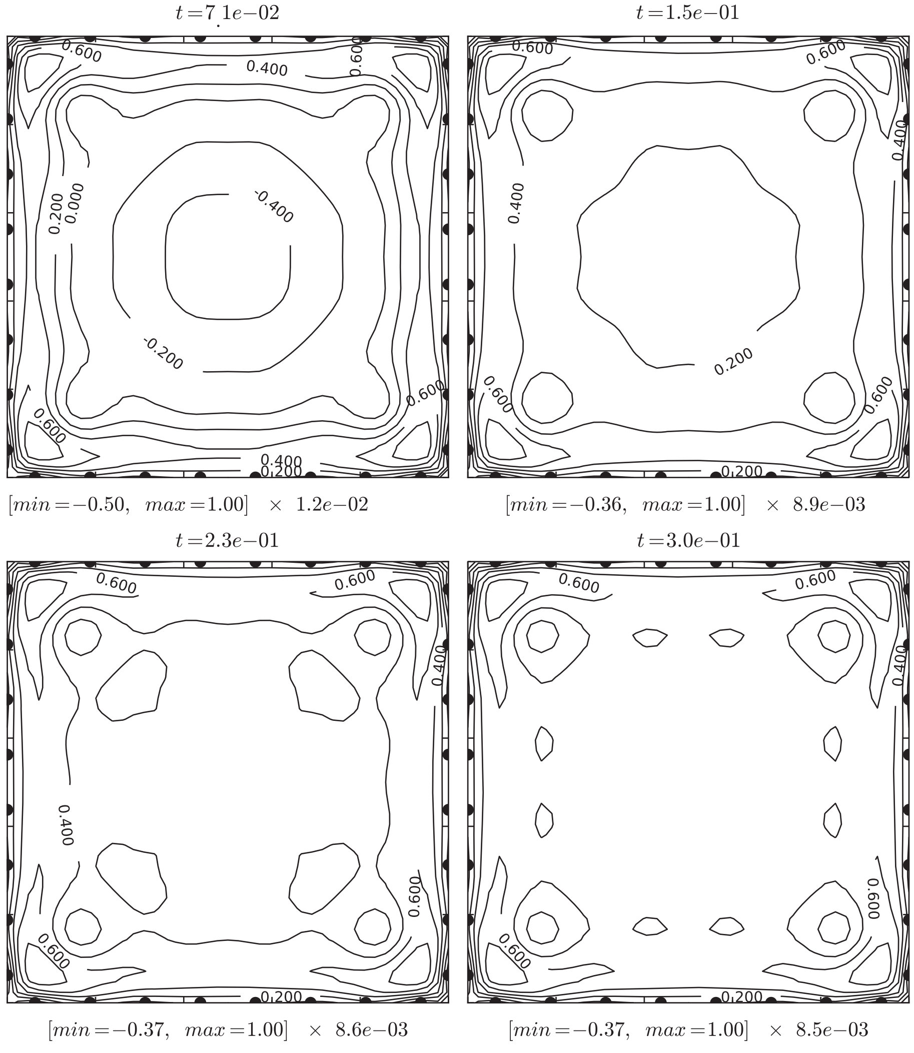


Fig. 5. $(u(\mathbf{x},t) - u^{ref}(\mathbf{x},t))/\|u^{ref}\|_\infty$ for $t = \{0.071s, 0.15s, 0.23s, 0.3s\}$, $n_\Gamma = 8$ and $n_\tau = 256$.

on the spatial mesh considered, and also $N \times Q$ 1D problems (defined in the time interval $\mathcal{I}$) that only need the solution of an ordinary differential equation from its initial condition. Obviously, even for extremely small time steps, the solution of these transient 1D problems does not introduce any difficulty.

If instead of applying the proper generalized decomposition just discussed, one performs a standard incremental solution, P dD models, $d=1$, 2, 3, must be solved (P being the number of time steps, i.e. $P=T_{max}/\Delta t$). The time step in incremental strategies has a direct impact on the convergence and stability of the numerical scheme.

In all the analyzed cases N and Q are of the order of tens that implies the solution of about hundred dD problems defined in Ω, instead the thousands (or even millions) needed for solving those models using standard incremental solvers.

A first comparison between both kind of approaches (the one based on the separated representation and the one based on standard incremental strategies) was presented in [25].

3. PGD based boundary element discretizations

The BEM is applied for solving Eq. (13). For this purpose that equation is rewritten as

$$-a\beta_t \Delta S = -\alpha_t S + \gamma_t - \sum_{i=1}^{n} \alpha_t^i X_i + a \sum_{i=1}^{n} \beta_t^i \Delta X_i \tag{22}$$

that defines a steady state elliptic problem with constant coefficients. It is important to note that β_t, $\alpha_t S$, α_t^i and β_t^i will be evaluated at each iteration of the fixed point algorithm from the solution computed at the previous iteration.

The computation of integrals involved in (15) requires the knowledge of the field S in the domain. This field is reconstructed by the BEM and approximated by using moving least squares [26].

As depicted in Fig. 1, the nodes inside Ω are used for defining the moving least square approximation of function $X_i(\mathbf{x})$ everywhere in Ω. They are distributed uniformly on a regular internal mesh consisting of $(n_\Gamma+1) \times (n_\Gamma+1)$ nodes. The Gauss points used for evaluating the domain integrals are located in the center of the internal cells defined by the moving least squares nodes in Ω. In the results that follow we use a single integration point by cell that are aligned with the collocation points.

The PGD algorithm involves $N \times Q$ times the solution of Eqs. (22) and (17), the computation of their associated integrals (12) and (15) and the reconstruction of S in Ω. It is important to note that matrix involved in the discretization are computed only once. Thus, the $N \times Q$ solutions only involve matrix vector products.

Thus, the solution complexity by using the PGD is close to the one related to the standard BEM solution of steady state models. However, the storage cost is slightly higher. Moreover, the cost associated with the computation of function R from Eq. (17) is simply negligible because it consists in the solution of a simple

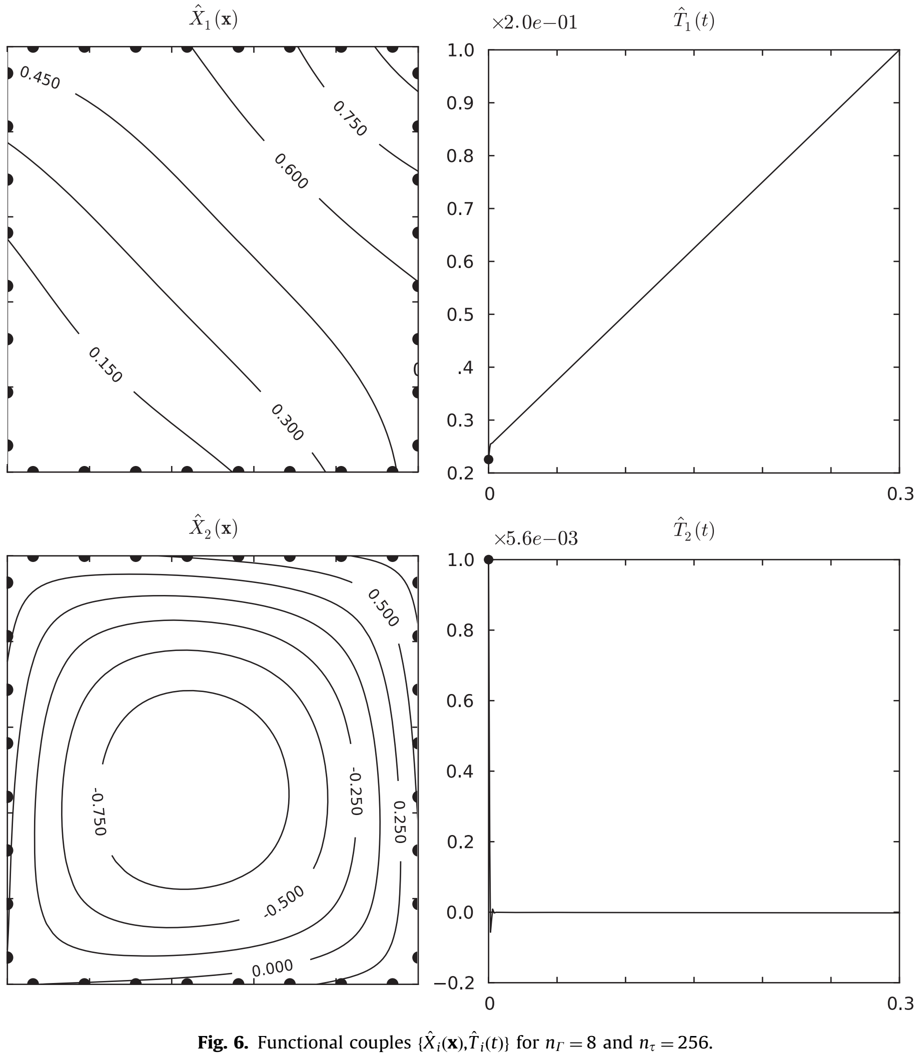


Fig. 6. Functional couples $\{\hat{X}_i(\mathbf{x}),\hat{T}_i(t)\}$ for $n_\Gamma=8$ and $n_\tau=256$.

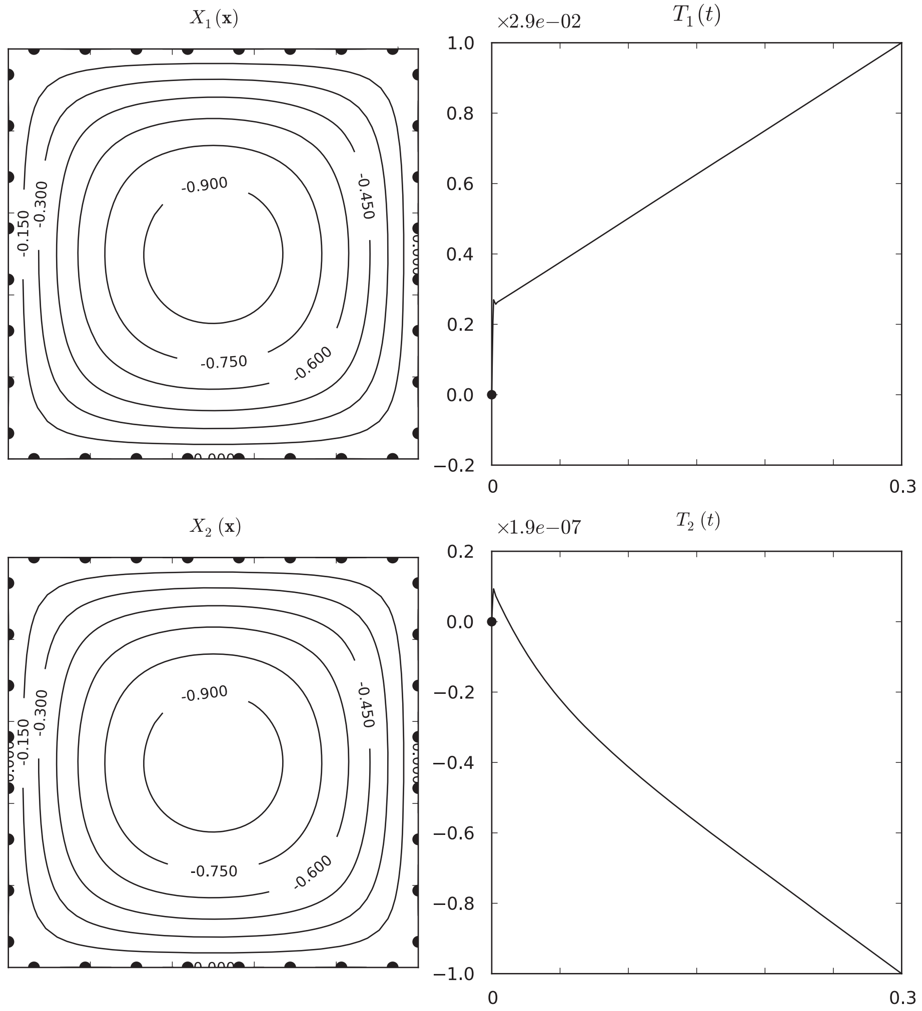


Fig. 7. Functional couples $\{X_i(\mathbf{x}),T_i(t)\}$ for $n_\Gamma = 8$ and $n_\tau = 256$.

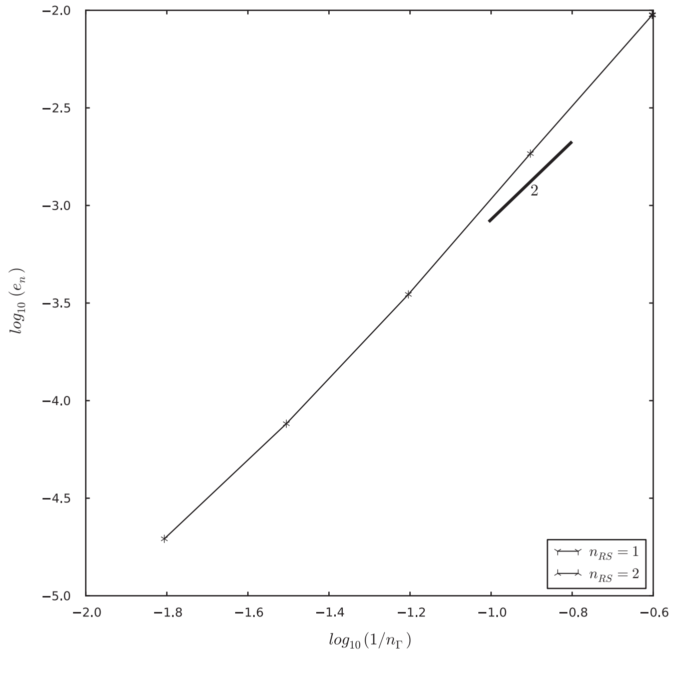


Fig. 8. Evolution of the error e_n versus the space discretization for different levels of approximation n.

ordinary differential equation. Even when the number of time steps (n_τ) is extremely large, its impact on the computing time is negligible because it only affects the solution of the ODE related to the solution of function R. The number of time steps must be large enough for ensuring an accurate evaluation of integrals (12).

4. Numerical results

4.1. Heat equation with source term

We considered heat equation in a simple rectangular domain $\Omega = (0,1) \times (0,1)$ with homogeneous boundary conditions and a time interval able to reach its steady state $\mathcal{I} = (0,0.3]$. The term source is set to a unit value $f(\mathbf{x},t) = 1$ and both the initial and boundary conditions vanish.

The exact solution writes [27]

$$u^{ref}(\mathbf{x},t) = \int_{\mathcal{I}} \int_0^1 \int_0^1 G(x,y,\bar{x},\bar{y},t-\bar{t})\, d\bar{x}\, d\bar{y}\, d\bar{t},$$

with

$$G(x,y,\bar{x},\bar{y},\tau) = 4 \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \sin(n\pi x)\sin(m\pi y) \times \sin(n\pi\bar{x})\sin(m\pi\bar{y})\exp(-\pi^2(n^2+m^2)\tau). \tag{23}$$

The domain boundary Γ consists of $n_\Gamma \times n_\Gamma$ segments Γ_i. The time interval $\mathcal{I}$ is discretized by using n_τ nodes, uniformly distributed.

The moving least square (MLS) approximation is built using a kernel function whose support radius is $3.5 \times 1/n_\Gamma$. We have verified that the use of cubic spline or exponential functions have not any incidence on the computed results. A quadratic consistency of the approximation is enforced by using a fully quadratic basis for defining the MLS approximation. The MLS approximation of the time functions is defined by using a support radius of $2.5 \times 1/n_{\mathcal{I}}$ and linear consistency.

The enforcement of the homogeneous boundary conditions is quite simple because it suffices for each functional couple that S vanishes on the domain boundary and R at t=0, conditions that are applied during the solution of Eqs. (22) and (17).

First we are analyzing the convergence rates as a function of the space discretization (i.e. n_Γ). For all the space meshes the time discretization (i.e. n_τ) is adapted in order to reach the maximum precision.

Table 1 and Fig. 2 show the evolution of the L^2 error in time and space as a function of the level of approximation, that is, as a function of the number of functional couples $X_i(\mathbf{x}) \cdot T_i(t)$ involved in the approximation of $u(\mathbf{x},t)$ for different meshes. This error is defined by:

$$e_n = \frac{\left\| \sum_{i=1}^{n} X_i(\mathbf{x}) \cdot T_i(t) - u^{ref}(\mathbf{x},t) \right\|_{L^2_{\Omega\times\mathcal{I}}}}{\|u^{ref}(\mathbf{x},t)\|_{L^2_{\Omega\times\mathcal{I}}}}.$$

We can notice that for a given number of functional couples the error e_n decreases when n_Γ increases, reaching an asymptotic value. For reducing the value of the error we must increase n_Γ as well as the number of functional couples $X_i(\mathbf{x}) \cdot T_i(t)$ involved in the functional approximation. In the case of the example here addressed we must consider 9 functional couples for reaching a quadratic convergence rate for $4 \le n_\Gamma \le 64$.

Fig. 3 depicts functions $\{X_1(\mathbf{x}),T_1(t)\}$, $\{X_2(\mathbf{x}),T_2(t)\}$, $\{X_3(\mathbf{x}),T_3(t)\}$ for $n_\Gamma = 8$ and $n_\tau = 256$.

Finally, Fig. 4 depicts the unknown field $u(\mathbf{x},t)$ and Fig. 5 its associated error $(u(\mathbf{x},t) - u^{ref}(\mathbf{x},t))/\|u^{ref}\|_\infty$ for $t = \{0.071s, 0.15s, 0.23s, 0.3s\}$.

4.2. *Heat equation with non-homogeneous initial and boundary conditions*

In what follows we consider the problem:

$$\mathcal{L}(u(\mathbf{x},t)) = f(\mathbf{x},t), \quad (\mathbf{x},t) \in \Omega \times \mathcal{I}, \tag{24}$$

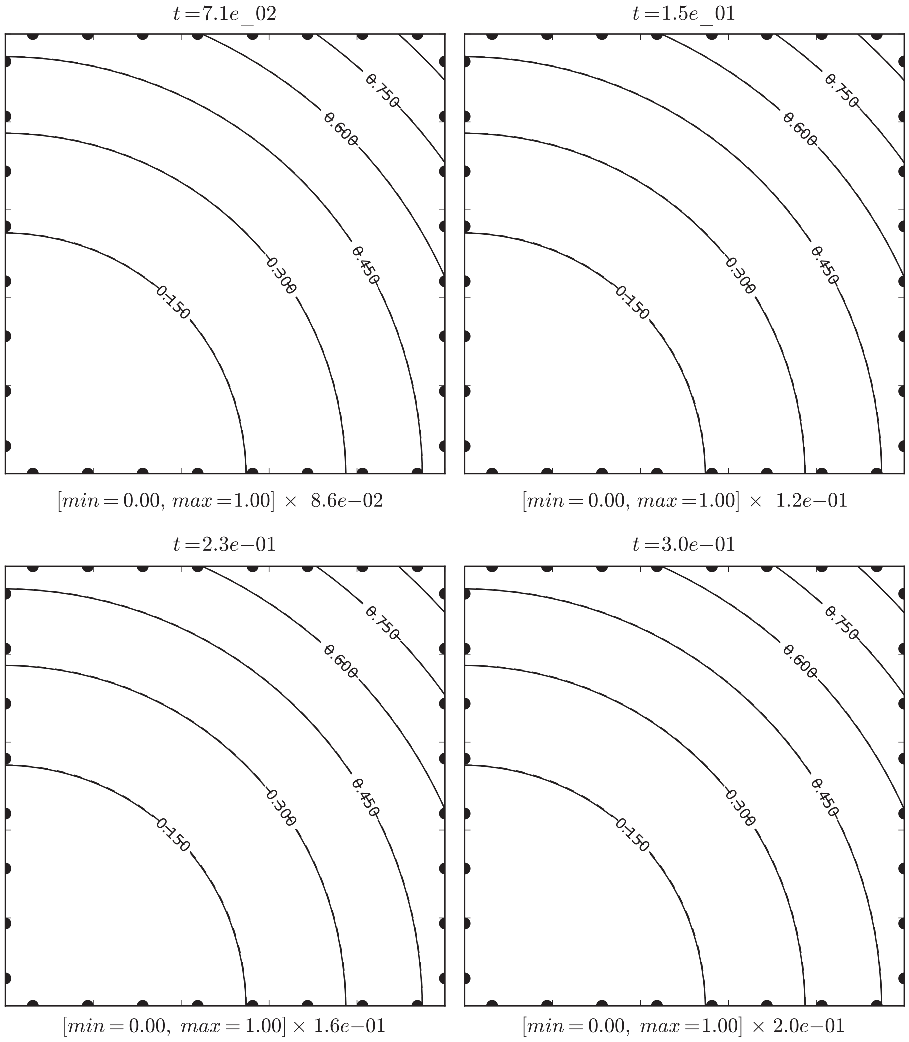


Fig. 9. $u(\mathbf{x},t)$ and $u^{ref}(\mathbf{x},t)$ (dashed line) for $t = \{0.071s, 0.15s, 0.23s, 0.3s\}$, $n_\Gamma = 8$ and $n_\tau = 256$.

$$u(\mathbf{x},t) = \hat{u}(\mathbf{x},t), \quad \mathbf{x} \in \Gamma_{\hat{u}}, \tag{25}$$

$$\frac{\partial u(\mathbf{x},t)}{\partial \mathbf{n}} = \hat{\mathbf{q}}(\mathbf{x},t), \quad \mathbf{x} \in \Gamma_{\hat{\mathbf{q}}}, \tag{26}$$

$$u(\mathbf{x},t=0) = u_0(\mathbf{x}), \quad \mathbf{x} \in \Omega. \tag{27}$$

The enforcement of the non-homogeneous initial and boundary conditions in the context of the PGD deserves some additional comments.

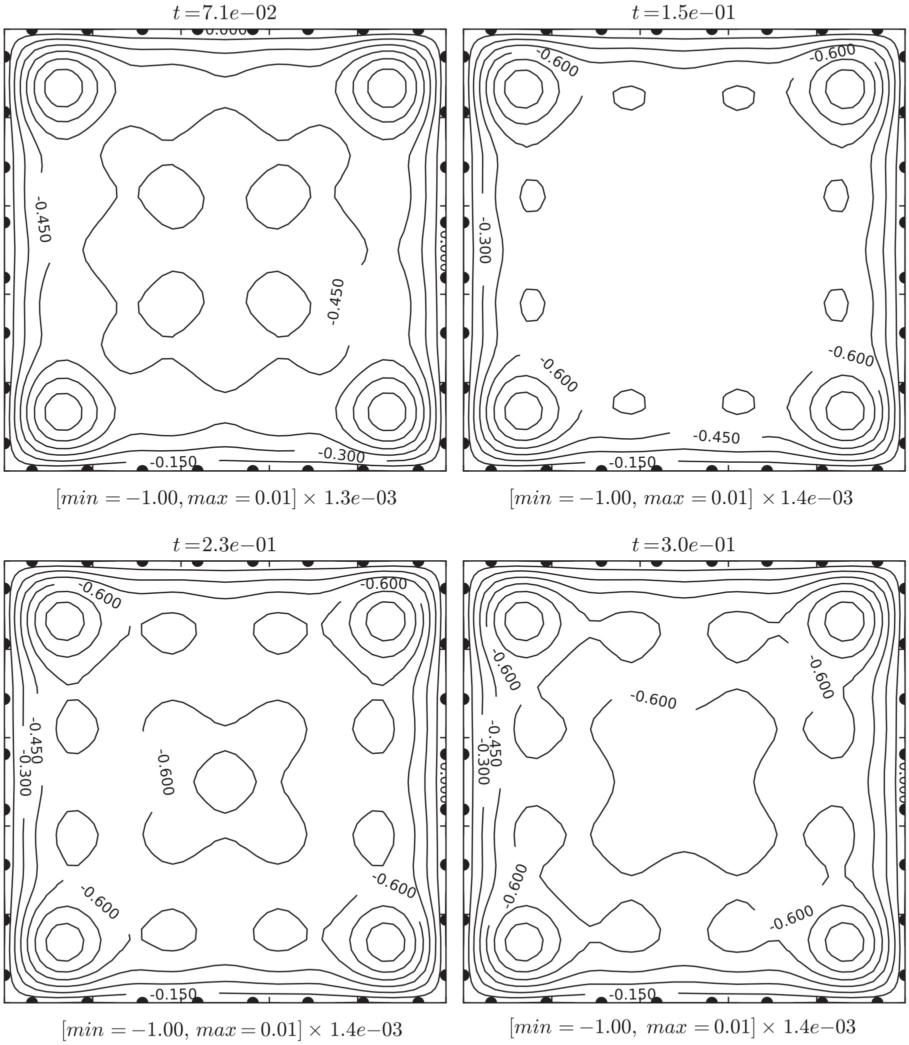


Fig. 10. $(u(\mathbf{x},t) - u^{ref}(\mathbf{x},t))/\|u^{ref}\|_\infty$ for $t = \{0.071s, 0.15s, 0.23s, 0.3s\}$, $n_\Gamma = 8$ and $n_\tau = 256$.

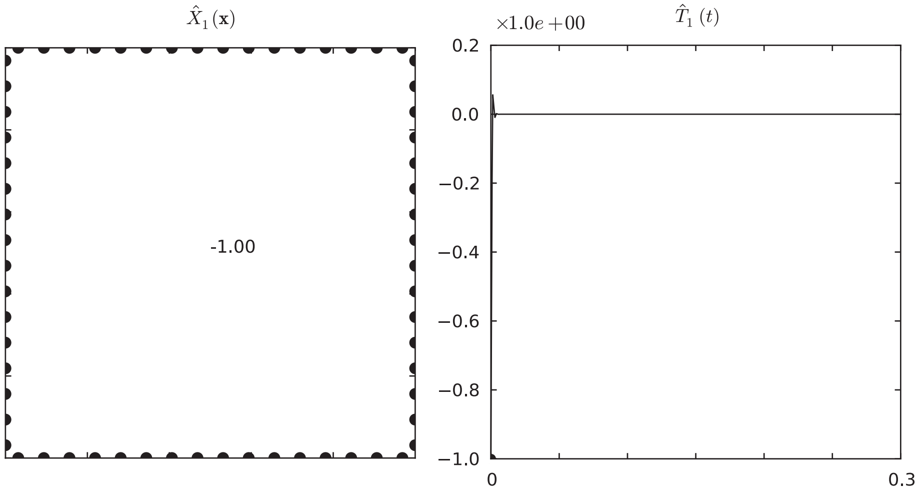


Fig. 11. Functional couples $\{\hat{X}_i(\mathbf{x}), \hat{T}_i(t)\}$ for $n_\Gamma = 16$ and $n_\tau = 256$.

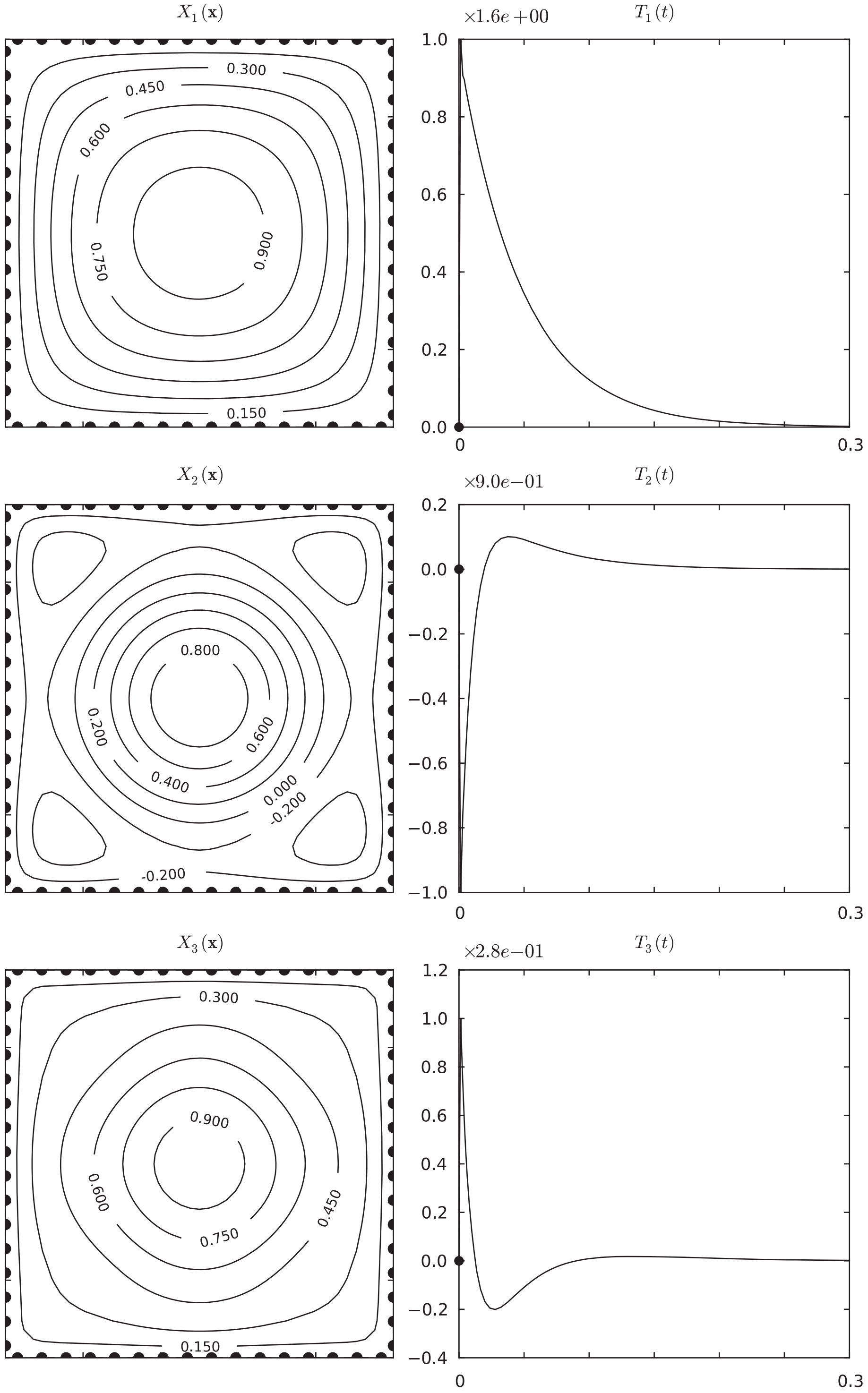


Fig. 12. Functional couples $\{X_i(\mathbf{x}),T_i(t)\}$ for $n_\Gamma = 16$ and $n_\tau = 256$.

We define $\hat{u}$ verifying the boundary conditions (25) and (26) as well as the initial condition (26). Introducing the change of variable $v = u - \hat{u}$, the solution of (24)–(27) is equivalent of finding v solution of

$$\mathcal{L}(v(\mathbf{x},t)) = f(\mathbf{x},t) - \mathcal{L}(\hat{u}(\mathbf{x},t)), \quad (\mathbf{x},t) \in \Omega \times \mathcal{I}, \tag{28}$$

with homogeneous initial and boundary conditions.

For solving efficiently Eq. (28) by using the PGD method we should give a separated representation of function $\hat{u}$:

$$\hat{u} = \sum_i \hat{X}_i(\mathbf{x})\hat{T}_i(t). \tag{29}$$

A procedure for generating such decomposition was given by [28] in the case of the Laplace equation. Here, we propose an alternative well adapted to the transient heat

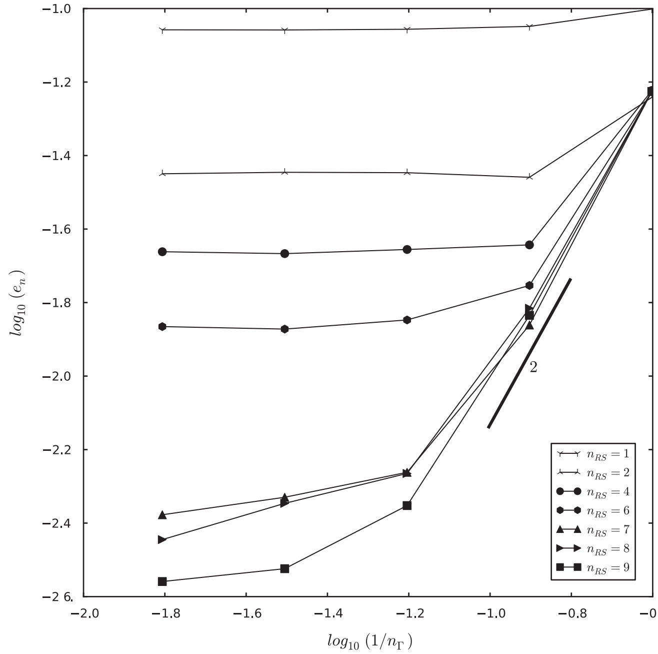


Fig. 13. Evolution of the error e_n versus the space discretization for different levels of approximation n.

equation. Inspired from the POD we define the matrix $C_{\hat{u}}$ containing the snapshots of function $\hat{u}$, whose each column contains the nodal values of $\hat{u}$ at time t_i. The first column of $C_{\hat{u}}$ being u_0.

We introduce the notation $\hat{u}^i = \hat{u}(\mathbf{x},t_i)$. Functions $\hat{u}^i$, verifying the boundary conditions expressed by (25) and (26), can be computed inside the domain and on $\Gamma_{\hat{\mathbf{q}}}$ by solving:

$$\Delta\hat{u}^i(\mathbf{x}) = 0, \quad \mathbf{x} \in \Omega, \tag{30}$$

$$\hat{u}^i(\mathbf{x}) = \hat{u}(\mathbf{x},t = t_i), \quad \mathbf{x} \in \Gamma_{\hat{u}}, \tag{31}$$

$$\partial\frac{\hat{u}^i(\mathbf{x})}{\partial\mathbf{n}} = \hat{\mathbf{q}}(\mathbf{x},t = t_i), \quad \mathbf{x} \in \Gamma_{\hat{\mathbf{q}}}. \tag{32}$$

Now, a singular value decomposition (SVD) of $C_{\hat{u}}$ allows computing the separated representation of $\hat{u}$ involving $\hat{X}_i$ and $\hat{T}_i$.

The number of columns, that is the snapshots t_i, to be considered in the construction of $C_{\hat{u}}$ must be large enough for describing accurately $\hat{u}$ but much lower than the time steps considered in the time discretization of the partial differential equation. The cost associated with the computation of functions $\hat{u}^i$ is quite reduced because the BEM discretization of Eq. (30) must be performed when one proceed using the PGD for computing function S.

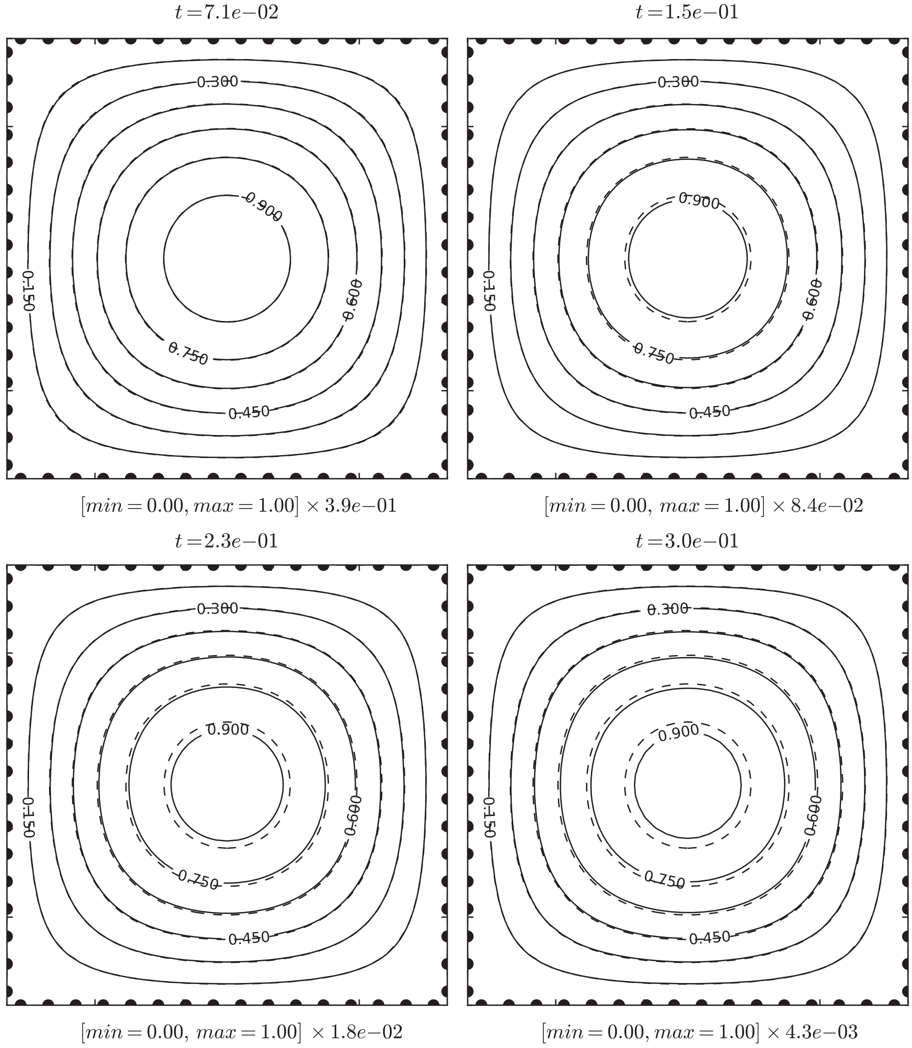


Fig. 14. $u(\mathbf{x},t)$ and $u^{ref}(\mathbf{x},t)$ (dashed line) for $t = \{0.071s,0.15s,0.23s,0.3s\}$, $n_\Gamma = 16$ and $n_\tau = 256$.

For illustrating the proposed approach we are considering the domain $\Omega=(0,1)\times(0,1)$ and the time interval $\mathcal{I}=(0,0.3]$. The source term writes $\frac{1}{4}(x^2+y^2)-(t-0.1)$. We enforce on the domain boundary the exact solution of the problem, i.e $u^{ref}(\mathbf{x},t)=\frac{1}{4}(x^2+y^2)(t-0.1)$.

Fig. 6 depicts the two most significant functional couples coming from the SVD decomposition of $C_{\hat{u}}$ that allow approximating $\hat{u}$.

The field v obtained by applying the PGD method is represented and accurately approximated by a single functional couple (as expected because the expression of the exact solution). Fig. 7 depicts the two functional couples.

Fig. 8 shows the evolution of the error e_n versus the space discretization for different levels of approximation n. A quadratic convergence is noticed.

Finally, Fig. 9 depicts the unknown field $u(\mathbf{x},t)$ and Fig. 10 its associated error $(u(\mathbf{x},t)-u^{ref}(\mathbf{x},t))/\|u^{ref}\|_\infty$ for $t=\{0.071s,0.15s,0.23s,0.3s\}$.

4.3. Heat equation with incompatible boundary and initial condition

We consider the heat equation in a simple rectangular domain $\Omega=(0,1)\times(0,1)$ and a time interval $\mathcal{I}=(0,0.3]$ with a null source term $f(\mathbf{x},t)=0$. The model is solved by assuming a homogeneous boundary condition and an initial condition taking a unit value everywhere except on the domain boundary on which it vanishes.

The exact solution writes [27]

$$u^{ref}(\mathbf{x},t)=\int_0^1\int_0^1 G(x,y,\overline{x},\overline{y},t)\,d\overline{x}\,d\overline{y},$$

with

$$\begin{aligned}G(x,y,\overline{x},\overline{y},t)= &\quad 1+2\sum_{n=1}^{\infty}\cos(n\pi x)\cos(n\pi\overline{x})\exp(-\pi^2n^2t)\Bigg)\\ &\times\ 1+2\sum_{m=1}^{\infty}\cos(m\pi y)\cos(m\pi\overline{y})\exp(-\pi^2m^2t)\Bigg).\end{aligned} \tag{33}$$

Fig. 11 depicts the functional couple arising from the SVD decomposition of $C_{\hat{u}}$, allowing defining $\hat{u}$.

Fig. 12 depicts functions $\{X_1(\mathbf{x}),T_1(t)\}$, $\{X_2(\mathbf{x}),T_2(t)\}$, $\{X_3(\mathbf{x}),T_3(t)\}$ for $n_\Gamma=16$ and $n_\tau=256$.

Fig. 13 shows the evolution of the error e_n versus the space discretization for different levels of approximation n. Again, a quadratic convergence is noticed.

The number of functional couples to be considered in the separated representation of the problem solution is now higher because the discontinuity involved in the initial condition.

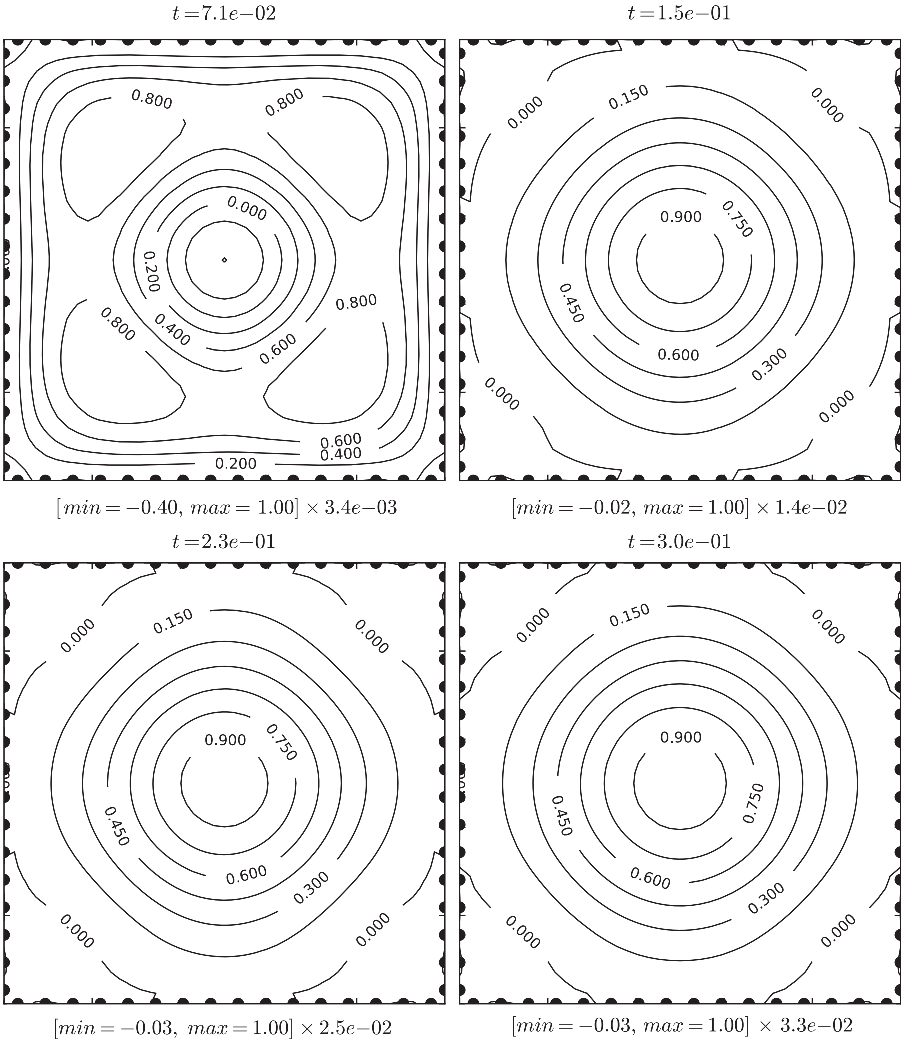
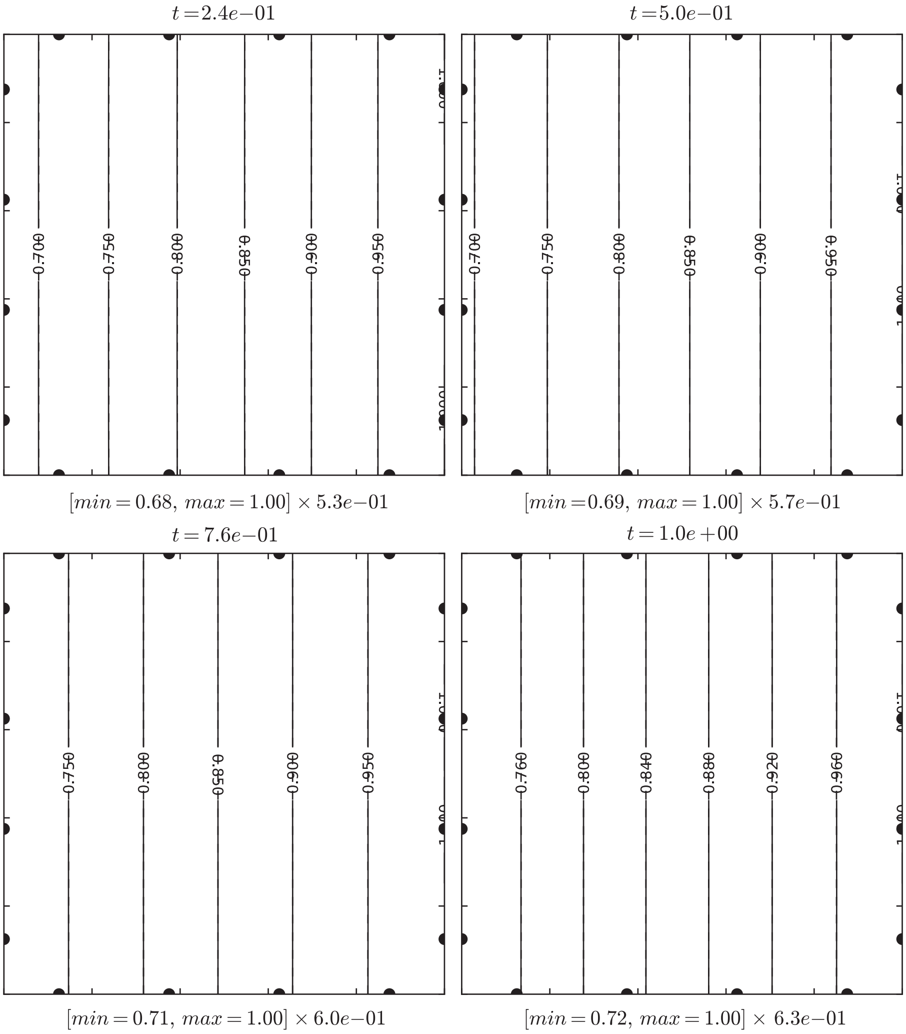


Fig. 15. $(u(\mathbf{x},t)-u^{ref}(\mathbf{x},t))/\|u^{ref}\|_\infty$ for $t=\{0.071s,0.15s,0.23s,0.3s\}$, $n_\Gamma=16$ and $n_\tau=256$.



$t = 2.4e{-}01$

$[min = 0.68,\ max = 1.00] \times 5.3e{-}01$

$t = 5.0e{-}01$

$[min = 0.69,\ max = 1.00] \times 5.7e{-}01$

$t = 7.6e{-}01$

$[min = 0.71,\ max = 1.00] \times 6.0e{-}01$

$t = 1.0e{+}00$

$[min = 0.72,\ max = 1.00] \times 6.3e{-}01$

Fig. 16. $u(\mathbf{x},t)$ and $u^{ref}(\mathbf{x},t)$ (dashed line) for $t = \{0.24s, 0.5s, 0.76s, 1s\}$, $n_\Gamma = 4$ and $n_\tau = 256$.

Finally, Fig. 14 depicts the unknown field $u(\mathbf{x},t)$ and Fig. 15 its associated error $(u(\mathbf{x},t) - u^{ref}(\mathbf{x},t))/\|u^{ref}\|_\infty$ for $t = \{0.071s, 0.15s, 0.23s, 0.3s\}$.

4.4. Heat equation with non-linear source term

We considered the heat equation with a non-linear source term in a simple rectangular domain $\Omega = (0,1) \times (0,1)$ and a time interval $\mathcal{I} = (0,1]$. The source term is set to $f(u) = u^2(1-u)$, and the initial and boundary conditions are chosen in order to agree with the known exact solution given by

$$u^{ref}(\mathbf{x},t) = \frac{e^{\eta(\mathbf{x},t)}}{2 + e^{\eta(\mathbf{x},t)}},$$

with $\eta(\mathbf{x},t) = 1/\sqrt{2}(x + t/\sqrt{2})$.

The strategy adopted for accounting for the non-linearity of f lies in expressing f in a separated form by suing as previously the POD. Because function f depends on u we should perform such decomposition at each iteration of PGD enrichment stage or even at the level of the fixed point algorithm. In the last case we should perform the decomposition of the source term $N \times Q$ times, instead the N decompositions involved by the former alternative. In any case, here we are only interested for evaluating the possibility of solving non-linear transient models, being the aspects related to the computational efficiency a work in progress.

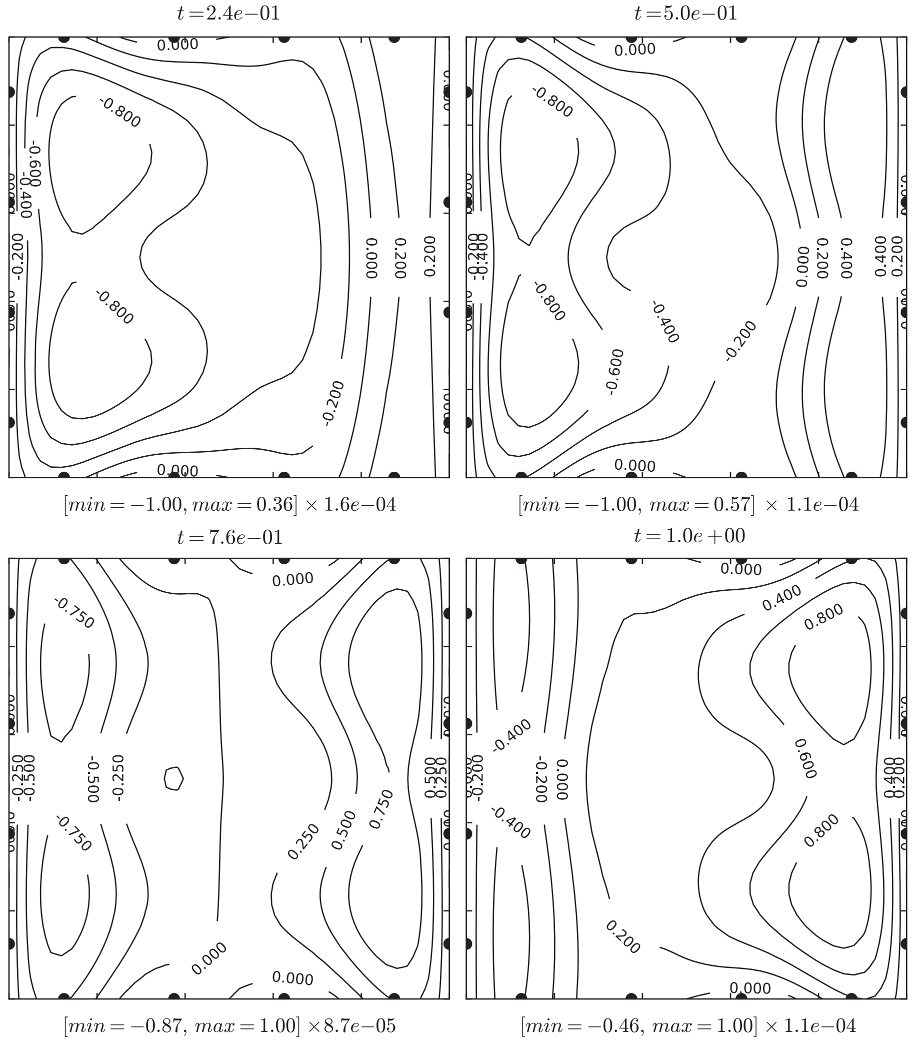
Fig. 16 depicts the unknown field $u(\mathbf{x},t)$ and Fig. 17 its associated error $(u(\mathbf{x},t) - u^{ref}(\mathbf{x},t))/\|u^{ref}\|_\infty$ for $t = \{0.24s, 0.5s, 0.76s, 1s\}$. From these results we can conclude ability of the PGD for solving complex transient models.

5. Conclusion

The strategy proposed in this paper allows to define a non-incremental boundary element method strategy for solving linear parabolic partial differential equations. Thus, instead of applying the standard incremental BEM using the mandatory space–time kernel, we only need to know the steady kernel associated to the steady-state diffusion–reaction problem that defines the space functions. The time functions defined in the whole time domain are computed by solving a ODE.

Obviously, by using an appropriate linearization the proper generalized decomposition based BEM could be extended for solving non-linear models.

In principle, even if the computed code is not optimized from a computing time point of view, the use of the proposed technique should alleviate the storage needs, in comparison with standard incremental strategies. Moreover, significant CPU time savings are expected, as were noticed and reported in some of our former works, in the finite element framework [25].



$t=2.4e-01$

$[min=-1.00, max=0.36]\times 1.6e-04$

$t=5.0e-01$

$[min=-1.00,\ max=0.57]\ \times\ 1.1e-04$

$t=7.6e-01$

$[min=-0.87,\ max=1.00]\ \times 8.7e-05$

$t=1.0e+00$

$[min=-0.46,\ max=1.00]\ \times\ 1.1e-04$

Fig. 17. $(u(\mathbf{x},t) - u^{ref}(\mathbf{x},t))/\|u^{ref}\|_\infty$ for $t=\{0.24s, 0.5s, 0.76s, 1s\}$, $n_\Gamma=4$ and $n_\tau=256$.

References

[1] Tausch J. A fast method for solving the heat equation by layer potentials. Journal of Computational Physics 2007;224(2):956–69, doi:10.1016/j.jcp.2006.11.001.

[2] Sutradhar A, Paulino GH, Gray LJ. Transient heat conduction in homogeneous and non-homogeneous materials by the Laplace transform Galerkin boundary element method. Engineering Analysis with Boundary Elements 2002;26(2):119–32, doi:10.1016/S0955-7997(01)00090-X.

[3] Amado JM, Tobar MJ, Ramil A, Yáñez A. Application of the Laplace transform dual reciprocity boundary element method in the modelling of laser heat treatments. Engineering Analysis with Boundary Elements 2005;29(2):126–35, doi:10.1016/j.enganabound.2004.12.005.

[4] 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(3):153–76, doi:10.1016/j.jnnfm.2006.07.007.

[5] 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 modelling of complex fluids: part ii: transient simulation using space–time separated representations. Journal of Non-Newtonian Fluid Mechanics 2007;144(2–3):98–121, doi:10.1016/j.jnnfm.2007.03.009.

[6] Mokdad B, Pruliere E, Ammar A, Chinesta F. On the simulation of kinetic theory models of complex fluids using the Fokker–Planck approach. Applied Rheology 2007;17(2):1–14.

[7] Chinesta F, Ammar A, Falco A, Laso M. On the reduction of stochastic kinetic theory models of complex fluids. Modeling Simulation in Materials Science Engineering 2007;15:639–52.

[8] Ammar A, Chinesta F, Joyot P. The nanometric and micrometric scales of the structure and mechanics of materials revisited: an introduction to the challenges of fully deterministic numerical descriptions. International Journal for Multiscale Computational Engineering 2008;6(3):191–213, doi:10.1615/IntJMultCompEng.v6.i3.20.

[9] Chinesta F, Ammar A, Cueto E. Proper generalized decomposition of multiscale models. International Journal for Numerical Methods in Engineering 2009, doi:10.1002/nme.2794.

[10] 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):1603–26, doi:10.1016/j.cma.2010.01.009.

[11] Park HM, Cho DH. The use of the Karhunen–Loève decomposition for the modelling of distributed parameter systems. Chemical Engineering Science 1996;51:81–98.

[12] Maday Y, Ronquist EM. The reduced basis element method: application to a thermal fin problem. SIAM Journal on Scientific Computing 2005;26(1):240–58, doi:10.1137/S1064827502419932.

[13] Ryckelynck D, Hermanns L, Chinesta F, Alarcón E. An efficient ‘‘a priori’’ model reduction for boundary element models. Engineering Analysis with Boundary Elements 2005;29(8):796–801, doi:10.1016/j.enganabound.2005.04.003.

[14] Ryckelynck D. A priori hyperreduction method: an adaptive approach. Journal of Computational Physics 2005;202(1):346–66, doi:10.1016/j.jcp.2004.07.015.

[15] Ryckelynck D, Chinesta F, Cueto E, Ammar A. On the a priori model reduction: overview and recent developments. Archives of Computational Methods in Engineering 2006;13(1):91–128, doi:10.1007/BF02905932.

[16] Ammar A, Ryckelynck D, Chinesta F, Keunings R. On the reduction of kinetic theory models related to finitely extensible dumbbells. Journal of Non-Newtonian Fluid Mechanics 2006;134(1–3):136–47. Second annual European rheology conference. doi:10.1016/j.jnnfm.2006.01.007.

[17] Chinesta F, Ammar A, Lemarchand F, Beauchene P, Boust F. Alleviating mesh constraints: model reduction, parallel time integration and high resolution homogenization. Computer Methods in Applied Mechanics and Engineering 2008;197(5):400–13. Enriched simulation methods and related topics. doi:10.1016/j.cma.2007.07.022.

[18] Bungartz H-J, Griebel M. Sparse grids. Acta Numerica 2004;13:147–269, doi:10.1017/S0962492904000182.

[19] Achdou Y, Pironneau O. Computational methods for option pricing (Frontiers in applied mathematics). Society for Industrial and Applied Mathematic; 2005.

[20] Rassias TM, Simsa J. Finite sums decompositions in mathematical analysis. John Wiley & Sons Inc; 1995.

[21] Beylkin G, Mohlenkamp MJ. Algorithms for numerical analysis in high dimensions. SIAM Journal on Scientific Computing 2005;26(6):2133–59, doi:10.1137/040604959.

[22] Cancès E, Defranceschi M, Kutzelnigg W, Bris CL, Maday Y. Computational quantum chemistry: a primer. In: Handbook of numerical analysis, vol. 10. Elsevier; 2003. p. 3–270.

[23] Ladevèze P. Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculation, Mechanical engineering series. Springer; 1999.

[24] Ladevèze P, Passieux JC, Neron D. The Latin multiscale computational method and the proper orthogonal decomposition. Computer Methods in Applied Mechanics Engineering 2010;199(21–22):1287–96, doi:10.1016/j.cma.2009.06.023.

[25] Ammar A, Normandin M, Daim F, Gonzalez D, Cueto E, Chinesta F. Nonincremental strategies based on separated representations: applications in computational rheology. Communications in Mathematical Sciences 2010;8(3):671–95.

[26] Nayroles B, Touzot G, Villon P. Generalizing the finite element method: diffuse approximation and diffuse elements. Computational Mechanics 1992;10:307–18.

[27] Carslaw HS, Jaeger JC. Conduction of heat in solids. 2nd ed. USA: Oxford University Press; 1986.

[28] González D, Ammar A, Chinesta F, Cueto E. Recent advances on the use of separated representations. International Journal for Numerical Methods in Engineering 2007;81(5):637–59, doi:10.1002/nme.2710.