# Parametric PGD model used with orthogonal polynomials to assess efficiently the building's envelope thermal performance

Marie-Hélène Azam, Julien Berger, Sihem Guernouti, Philippe Poullain, Marjorie Musy

# Parametric PGD model used with orthogonal polynomials to assess efficiently the building's envelope thermal performance

Marie-Hélène Azam[1,2,*], Julien Berger[3], Sihem Guernouti[1,2], Philippe Poullain[1], Marjorie Musy[1,2]

January 6, 2021

[1] *Université de Nantes, GeM UMR 6183 CNRS/Université de Nantes/Centrale Nantes, F-44600 Saint Nazaire, France*

[2] *Cerema, Equipe de Recherche BPE, F-44000 Nantes, France*

[3] *Université de La Rochelle, LaSIE UMR 7356 CNRS, F-17000, La Rochelle, France*

[*] *Corresponding author: Azam Marie-Hélène, marie-helene.azam@univ-nantes.fr, IUT de Saint-Nazaire, Département Génie Civil, 58 rue Michel Ange, F-44600 Saint Nazaire*

**Abstract:** Estimating the temperature field of a building envelope could be a time-consuming task. The use of a reduced-order method is then proposed: the Proper Generalized Decomposition method. The solution of the transient heat equation is then re-written as a function of its parameters: the boundary conditions, the initial condition, etc. To avoid a tremendous number of parameters, the initial condition is parameterized. This is usually done by using the Proper Orthogonal Decomposition method to provide an optimal basis. Building this basis requires data and a learning strategy. As an alternative, the use of orthogonal polynomials (Chebyshev, Legendre) is here proposed.

**Key words :** Heat transfer; Model order reduction methods; POD; PGD; Approximation Basis; Orthogonal Polynomials.

**Highlights**

- Chebyshev and Legendre polynomials are used to approximate the initial condition
- Performance of Chebyshev and Legendre polynomials are compared to the POD basis
- Each basis combined with the PGD model is compared to laboratory measurements
- The influence of four different parameters on the accuracy of the basis is studied
- For each approximation basis, CPU calculation times are evaluated and compared

## 1 Introduction

Modeling the thermal behavior of a building or a group of buildings is a challenging task. It implies that several physical phenomena have to be taken into account: short and long-wave radiative heat balance, sensible and latent heat flow transported by outdoor air movement and conductive heat transfer through the materials. All those heat fluxes may vary over time and through space and thus lead to complex and non-uniform boundary conditions.

To quantify the global heat loss of a building envelope, the balance between the outdoor thermo-radiative heat fluxes and the indoor ones must be estimated. One way of solving the global problem is to split the problem into several sub-problems relative to (i) the outside thermo-radiative balance, (ii) the inside thermo-radiative balance, and (iii) the heat transfer through the envelope [1]. Each problem is solved using a numerical model for each set of governing equations. The whole energy model represents then the aggregation of those several sub-models through a coupling procedure also called co-simulation [2]. Each numerical model exchanges parameters (*i.e.* surface energy balance or surface temperature in the case of a thermal problem) with the other models during the simulation process.

This combination of models results in a large computation complexity and we need to reduce the computational times.

We focus here on the problem of the building envelope. To solve the global problem, for each element of the building (walls, floor, *etc.*), the temperature field needs to be computed. For that purpose, the transient heat transfer equation needs to be solved for the previously described boundary conditions (indoor and outdoor thermo-radiative balance). Usually, a classical numerical model is then used based on finite difference, finite element, or finite volume. Those methods provide an accurate solution but for a high computation cost.

To reduce the computational time keeping an accurate solution, the use of model order reduction methods is currently investigated. The main idea is to replace the detailed and time-consuming model with a reduced-order model. For that purpose, we investigate the use of the Proper Generalized Decomposition (PGD) method.

Applied to urban soil heat transfer modeling, this model reduction method has shown its efficiency [3]. A cut computational cost of 80% was observed for a mean surface temperature error below $0.52°C$. Applied to building wall heat transfer modeling, the PGD parametric model computes the solution 100 times faster than a classical numerical method [4].

To reduce the numerical complexity of the problem, the solution is decomposed as a function of parameters like the boundary conditions, or the initial condition. The efficiency of the PGD method relies on the number of parameters used. To obtain a minimum number of parameters, some of them are usually combined through approximation. Those approximations are done by the projection of the field of interest on an approximation basis.

Selecting the right approximation basis that will guarantee the model final accuracy with a minimum number of parameters is a challenging task. The purpose of this article is then to overcome this obstacle. We investigate here the use of a polynomial basis like Chebyshev or Legendre. Out of the approximation theory [5], those basis have proven to be very efficient at solving partial differential equations using spectral methods [6, 7].

The use of a polynomial basis is compared to the use of a classical reduced-order basis obtain through the Proper Orthogonal Decomposition (POD) method. For that purpose, section 2 presents each basis and their combination with the PGD method. Each combination is compared both on its accuracy and computation time. The global methodology applied is explained in section 3. To evaluate the basis in several situations, two case studies are presented. The first case study, presented in section 4 is a theoretical application. In section 5, the models are then applied to a practical case with realistic boundary conditions. The results of the models

will be confronted with laboratory measurements. For each case study, the influence of several parameters on the accuracy of the approximation basis is investigated: the number of modes in the approximation basis, the number of modes in the PGD model and the discretization.

# 2 Materials and Methods

## 2.1 Physical problem of heat transfer in building wall

The physical problem studied is defined to be as close as possible to problems usually solved by building energy models (e.g. EnergyPlus [8]). It involves transient one-dimensional heat conduction through a wall without volumetric heat dissipation for a time interval $\Omega_\tau$ with $t \in [0, \tau]$ and space interval $\Omega_x$with $x \in [0, L]$:

$$c \frac{\partial u(x,t)}{\partial t} = \frac{\partial}{\partial x}\left( k \frac{\partial u(x,t)}{\partial x} \right), \tag{1}$$

On each side of the wall, a FOURIER boundary condition is assumed. On $x = 0$, the boundary condition can be described by the following equation:

$$-k \frac{\partial u(x,t)}{\partial x} = q(t) - h_{out}\left( u(x,t) - u_{out}(t) \right), \qquad x = 0, \tag{2}$$

The surface energy balance depends on a net radiative heat flux, noted q, and a sensible heat flux. The last one is calculated from the outdoor air temperature $u_{out}$ varying over time and from a convective heat transfer coefficient $h_{out}$.

On $x = L$, the boundary condition can be described by the following equation:

$$k \frac{\partial u(x,t)}{\partial x} = -h_{in}\left( u(x,t) - u_{in}(t) \right), \qquad x = L. \tag{3}$$

As it is illustrated with the outside boundary condition, another radiative heat flux could have been added to the inside boundary condition to complexify the mathematical model proposed here. The net radiative heat flux is neglected on that side of the wall. To support this hypothesis, the error due to this simplification of the mathematical model is studied in the appendix A. Note that this assumption will not have an impact on the results presented because the same mathematical model is used for all the numerical models developed.

The sensible heat flux is calculated from the indoor air temperature $u_{in}$ that varies over time and from a convective heat transfer coefficient $h_{in}$.

The initial temperature is uniform:

$$u(x,t) = u_0, \qquad t = 0. \tag{4}$$

Equation (1) can be written in a dimensionless form as:

$$\frac{\partial u(x,t)}{\partial t} = Fo \frac{\partial^2 u(x,t)}{\partial x^2}, \tag{5}$$

for a time interval $\Omega_\Gamma = \left[0, \Gamma\right]$ and space interval $\Omega_x = \left[0, 1\right]$, and the boundary condition as:

$$\frac{\partial u(x,t)}{\partial x} = Bi_{out}\left(u - u_{out}\right) - q, \qquad x = 0, \tag{6a}$$

$$\frac{\partial u(x,t)}{\partial x} = -Bi_{in}\left(u - u_{in}\right), \qquad x = 1. \tag{6b}$$

The initial condition becomes:

$$u = 0, \qquad t = 0. \tag{7}$$

Where the dimensionless quantities are defined as:

$$u := \frac{\mathrm{u} - \mathrm{u}_0}{\mathrm{u}_0}; \quad t := \frac{\mathrm{t}}{\mathrm{t}_{\mathrm{ref}}}; \quad x = \frac{\mathrm{x}}{\mathrm{L}}; \quad Bi_{in} := \frac{\mathrm{h_{in}.L}}{\mathrm{k}}; \quad Bi_{out} := \frac{\mathrm{h_{out}.L}}{\mathrm{k}}; \quad Fo := \frac{\mathrm{k}\,.\,\mathrm{t_{ref}}}{\mathrm{c\,L^2}} = 1$$

$$\mathrm{t_{ref}} := \frac{\mathrm{c\,L^2}}{\mathrm{k}}; \quad u_{in} := -1 + \frac{\mathrm{u_{in}}}{\mathrm{u_0}}; \quad u_{out} := -1 + \frac{\mathrm{u_{out}}}{\mathrm{u_0}}; \quad q := \frac{\mathrm{q}\,.\,\mathrm{L}}{\mathrm{k}\,.\,\mathrm{u_0}}; \quad \Gamma = \frac{\tau}{\mathrm{t_{ref}}}$$

## 2.2 The related boundary value problem in the context of co-simulation

The physical problem involves the partial differential equation (PDE) Eq. (5) together with the boundary (Eqs. (6a) and (6b)) and initial conditions (Eq. (7)). As presented on Figure 1, it is solved in the context of co-simulation (or coupling) with other numerical models (models 1 and 2), by solving the radiative heat balance and the air transfer around the building walls.

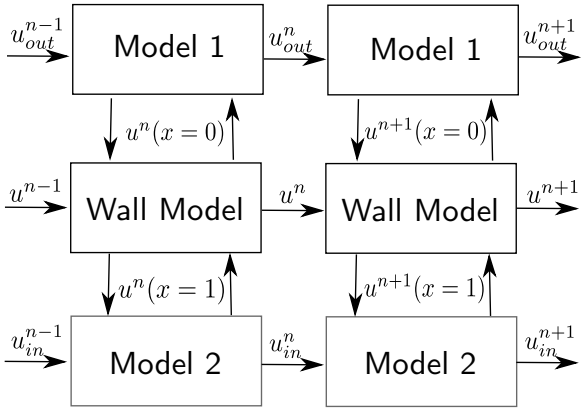



Figure 1: Co-simulation process

In this context, the initial boundary value problem Eq. (5) is semi-discretized along the time line [9]. The time discretization parameter is denoted by $\Delta t$, corresponding to the time step of coupling between the numerical models of the co-simulation. The discrete values of functions $u(x, t)$ is written as $u^n := u(x, t^n)$ with $n = 1, \ldots, N_t$. Thus, using an implicit approach, Eq. (5) becomes:

$$u^{n+1} = u^n + \Delta t \cdot Fo \cdot \frac{\partial^2 u^{n+1}}{\partial x^2}, \tag{8}$$

By introducing $y \equiv u^{n+1}$, Eq. (8) can be reformulated as:

$$y \; - \; a \cdot \frac{\partial^2 y}{\partial x^2} \; = \; b(x)\,. \tag{9}$$

Here, $y$ is the unknown of our boundary value problem and depends on the space coordinate $x$. The coefficient $a \overset{\text{def}}{:=} \Delta t \cdot Fo$ depends on the properties of the material composing the wall and on the co-simulation time step. The coefficient $b \overset{\text{def}}{:=} u^n$ is qualified as the source term of the boundary value problem, depending on the space coordinate $x$. It also varies at each time step of the co-simulation. The boundary conditions Eqs. (6a) and (6b) are also transformed:

$$\frac{\partial y}{\partial x} = Bi_{out} \cdot y \; - \; b_{out}\,, \qquad x \; = \; 0\,, \tag{10a}$$

$$\frac{\partial y}{\partial x} = -\,Bi_{in} \cdot y \; + \; b_{in}\,, \qquad x \; = \; 1\,, \tag{10b}$$

where the coefficients $b_{out}$ and $b_{in}$ are:

$$b_{out} = -\,Bi_{out} \cdot u_{out}(t^n) \; - \; q(t^n)\,, \qquad b_{in} \; = \; Bi_{in} \cdot u_{in}(t^n)\,.$$

Both are constants given at each time step $\Delta t$ of the co-simulation by model 1 and 2.

## 2.3 Formulation of the parametric problem

The boundary value problem Eq. (9) together with the boundary conditions (10) are the main interest to build a reduced-order model. Several solvers exist to solve such a problem. A brief overview can be consulted in [10]. These numerical models are used to compute a solution $y(x)$ only depending on the space coordinate.

However, it is a challenging problem to build a solution depending on the space coordinate and on extra-parameters such as the source term $b$ and the coefficients $b_{out}$ and $b_{in}$. It requires to solve a so-called parametric problem. The use of the PGD methods gives the opportunity to decompose the solution of a problem as a function of any parameters to generate a parametric model.

Taking into account the source term as a parameter is another challenging task. Indeed, once discretized in space, the source term is made of discrete values: one information per piece of the mesh. It implies inputting as many parameters in the parametric model as the number of pieces of the mesh. To avoid this large number of involved parameters, the source term is approximated by its projection on an approximation basis with a lower rank:

$$b\big(x\big) \; = \; \sum_{j \, = \, 1}^{\mathcal{N}} \Psi_j\big(x\big)\,\zeta_j \tag{11}$$

where $\Psi_j$ is the approximation basis, $\zeta_j$ are the coefficients of the projection and $\mathcal{N}$ the number of modes in the basis. So, the solution of Eq. (9) is searched as:

$$u \; : \; \Big[\,0\,,\,1\,\Big] \times \Omega_{b_{out}} \times \Omega_{b_{in}} \times \Omega_{\zeta_j} \longrightarrow \mathbb{R}\,,$$

$$\big(x\,,\,b_{out}\,,\,b_{in}\,,\,\zeta_j\big) \longmapsto u\,\big(x\,,\,b_{out}\,,\,b_{in}\,,\,\zeta_j\big)\,.$$

The sets $\Omega_{b_{out}}$, $\Omega_{b_{in}}$ and $\Omega_{\zeta_j}$ are the domain of variations of the coordinates $b_{out}$, $b_{in}$ and $\zeta_j$, respectively.

They are defined such as:

$$\Omega_{b_{out}} = \left[ b_{out}^{-}, b_{out}^{+} \right], \qquad \Omega_{b_{in}} = \left[ b_{in}^{-}, b_{in}^{+} \right], \qquad \Omega_{\zeta_j} = \left[ \zeta_j^{-}, \zeta_j^{+} \right], \qquad j \in \left[ 0, \mathcal{N} \right].$$

Their respective discretization parameters are denoted by $\Delta_{b_{out}}$, $\Delta_{b_{in}}$ and $\Delta_{\zeta_j}$.

## 2.4 Approximation basis

In the literature, several parameterizations have been studied. Chinesta *et al.* (2013 [11]) and Gonzalez *et al.* (2012 [12]) proposed to use the nodal values corresponding to the piece-wise linear finite element approximation of the problem. However, according to Gonzalez *et al.* (2014 [13]), this method leads to a large number of degrees of freedom: their model is made of one parameter per nodal values. That is why they proposed to use the POD to provide a suitable parameterization of the initial condition with the lowest number of degrees of freedom [13]. More information can be found on this method applied to convective heat transfer in [14] and solid dynamics in [13, 15].

One of the main drawbacks of this method is that a learning process is needed. It has an impact on the accuracy of the reduced-order basis. For this reason, the data-set used must be representative of the problem (boundary values, initial conditions, materials used).

According to Gonzalez *et al.* (2014 [13]) the initial condition could be interpolated by piece-wise polynomials. However, for the specific field of solid dynamics, this approach is not the best choice, considering the behavior of the system. Another solution proposed by Poulhaon *et al.* (2012, [16]) is to use an auxiliary mesh much coarser than the one used for the solution of the problem. A projection is made from the fine to the coarse mesh using the least square method. This method is purely mathematical and does not take into account physical considerations such as energy conservation or heat flux conservation. According to Poulhaon *et al.* (2012, [16]) it should be completed by a mathematical tool to take into account the physics of the studied phenomenon.

Conforming to [17], one important feature for the choice of an approximation basis is the *sparsity*. It ensures that the chosen basis has the required regularity to represent the solution. Spectral basis, such as polynomial or trigonometric functions, guarantee sparsity. For such functions, the values of the coefficients decrease exponentially with the order of approximation [18]. However, the basis is full (because a spectral basis is not interpolative [17]). It implies that the computational cost needed to determine the coefficients becomes impractical for large systems.

Based on the literature review, two methods are here compared: the use of a polynomials basis and the use of a POD basis to approximate the temperature profile. Details on how to build each approximation basis used are given in the appendix B.

As stated by the Weierstrass approximation theorem, every continuous function on a bounded interval can be approximated by a polynomial to a certain accuracy [5]. Several functions with polynomial basis can then be used according to the studied problem. The most simple polynomial basis is the monomial one. As described by Peyret (2013 [19]), if a periodic problem is studied, the Fourier method should be used. Yet, this method is not suitable for non-periodic problems, because of the Gibbs phenomenon. In this case, orthogonal polynomials such as Chebyshev or Legendre polynomials should be used.

Considering the numerous polynomial basis, the first difficulty is to select the right basis for the considered problem. As the field of interest is a non-periodic, smooth function, FOURIER or LAURENT polynomials shall not be studied here.

According to Trefethen (2013, [5]), the monomial basis is comfortable but should never be used to approximate a function. If we compare the condition number for inversion of the three basis, the CHEBYSHEV and LEGENDRE polynomials basis have a smaller condition number than the monomial one. If the condition number of a matrix is large, the matrix is close to being singular. The condition number reveals that the projection of the field of interest on the monomial basis will be sensitive to numerical round-off errors and perturbations in the input data. Moreover, monomial basis do not meet *sparsity* condition as its coefficients increase with the order. Therefore, this basis should not be used here to parameterize the initial condition.

According to Trefethen (2013, [5]), LEGENDRE points and polynomials are neither better than CHEBYSHEV ones for approximating functions, nor worse. The main advantage to use CHEBYSHEV over LEGENDRE points center around the use of FFT (Fast FOURIER Transform). This function can be used to get the coefficients from the point values or the reverse. But this property is not used here. Both polynomials basis will be compared.

The CHEBYSHEV and LEGENDRE polynomials are part of the family of orthogonal polynomials. They are calculated respectively at the CHEBYSHEV and LEGENDRE points. Special attention must be given to the spatial domain of the problem. The points define a non-uniform mesh for a space interval $[-1, 1]$. Thus, a change of variable must be performed to transform the dimensionless spatial domain $[0, 1]$ to $x \in [-1, 1]$.

## 2.5 Proper Generalized Decomposition method

Several MOR methods can be used to solve a parametric problem. One of them is the Proper Generalized Decomposition Method (PGD). It is an *a priori* MOR method based on the separation of variables. It does not reduce the system of equations itself but the whole parametric problem. Any variable can then be defined as an extra-parameter of the model [17].

With spectral methods [20], the PGD method is one of the unique methods that allows to create a complete parametric model without knowing *a priori* the solution of the problem.

The PGD is used to propose an accurate parametric solution of the formulated BVP problem. The method approximates the solution as a finite sum of separable functions. As presented in Section 2.3, the parametric model involves three parameters: the space, the boundary condition and the source term. Applying the PGD method, the solution is sought as the sum of $\mathcal{M}$ functional products involving each function as follows:

$$y \;=\; \sum_{m\,=\,1}^{\mathcal{M}} X_m\left(x\right) E_m\left(b_{in}\right) F_m\left(b_{out}\right) \prod_{j\,=\,1}^{\mathcal{N}} G_m^j\left(\zeta^j\right) \tag{12}$$

where $X$, $E$, $F$, and $G$ designate the functions of the parameters. Each function is defined over a domain : $\Omega_x \;=\; \left[-1, 1\right]$, $\Omega_{b_{in}} \;=\; \left[b_{in}^-, b_{in}^+\right]$, $\Omega_{b_{out}} \;=\; \left[b_{out}^-, b_{out}^+\right]$ and $\Omega_{\zeta_j} \;=\; \left[\zeta_j^-, \zeta_j^+\right]$.

The following weak form of the ODE is used with the test function $y^*$ (Galerkin formulation):

$$\int_{\Omega_x \times \Omega_{b_{in}} \times \Omega_{b_{out}} \times \Omega_{\zeta_j}} y^* . \left( y \; - \; a \, \frac{\partial^2 y}{\partial x^2} \; - \; \sum_{j \, = \, 1}^{\mathcal{N}} \Psi_j \left( x \right) \zeta_j \right) \mathrm{d}x . \mathrm{d}b_{in} . \mathrm{d}b_{out} . \mathrm{d}\zeta_j \; = \; 0 \tag{13}$$

The weak form of the ODE is regarded as an optimization problem. It leads to a nonlinear optimization problem due to the functional product of the subspaces. It can be solved with an iterative procedure that features two nested loops: the alternating direction strategy and the enrichment process [21]. The calculation of the unknowns is performed alternatively along each dimension until convergence [22]. In this way, the algorithm splits the high dimensional problem into a series of low dimensional ones. The complexity of the problem then grows linearly with the number of parameters. Each function $X_m$, $E_m$, $F_m$ and $G_m^j$ is first randomly initialized and then solved by iterations. The alternating directions process stops once a fixed point is reached. The criterion $\tilde{\epsilon}$ used to make this determination is defined by the user [11]. Once this criterion is reached, the new functions are added to the previous one in the PGD basis. The enrichment process of the PGD basis stops when the $\epsilon$ criterion, defined by the user, is reached [11]. Details on the alternating directions strategy equations and algorithms for a similar problem can be found on [3]. For further details on the method and its developments, the interested reader may refer to [11, 15].

Each function ($X_m$, $E_m$, $F_m$, $G_m^j$) defined previously depends on a continuous variable. To solve the parametric problem with the previous algorithm, the continuous variables need to be discretized. For that purpose, the continuous variable is projected on a mesh. The continuous variable is then described by a vector. The finer the mesh of discretization of each parameter, the closer the discrete value to the continuous one. But as a results, the number of elements in the vectors used to describe the parameter increases.

According to Leon *et al.* (2018, [23]), the final accuracy of a PGD model depends on the number of terms $\mathcal{M}$ in the final sum, on the number of parameters/vectors ( $x$, $b_{in}$, $b_{out}$, $\zeta_j$) and the discretization of those parameters. However, by increasing the number of elements in the mesh of discretization for each parameter, we increase the complexity of the problem. In the case of a PGD model, this complexity grows linearly with the number of parameters [17, 22]. As a comparison, the complexity of a grid-based discretization (finite element, finite difference) grows exponentially with the number of mesh elements. The number of elements on each vector is a matter of CPU time and space to save the PGD parametric model. As the purpose of building a PGD parametric model is to decrease the calculation time (compared to a classical model: finite difference, finite element) the number of elements in each vector should be then optimized.

For the spatial parameter and the boundary condition, the methodology to defined the discretization is classical, no special interrogation arises. However, each mode of the approximation basis also needs to be discretized. Several questions can arise for the coefficients $\zeta_j$ of the source term approximation. Spectral basis such as Chebyshev or Legendre guarantee sparsity. When this condition is met, the order of magnitude of the coefficients $\zeta_j$ decreases exponentially with the order of approximation [18]. The discretization of each coefficient needs to fit the order of magnitude of each mode. To simplify our study and only use one parameter to define the discretization of each coefficient of the basis, we propose to use dimensionless numbers for the coefficients $\zeta_j$ defined as $\overline{\zeta_j}$.

$$\overline{\zeta_j} = \frac{\zeta_j - min\left(\zeta_j\right)}{max\left(\zeta_j\right) - min\left(\zeta_j\right)} \tag{14}$$

where $\overline{\zeta_j} \in [0,1]$ and $\zeta_j \in [min\,(\zeta_j)\,, max\,(\zeta_j)]$.

## 2.6 *Offline / online* strategy

The use of the PGD method, to solve the parametric problem, features an offline-online strategy. During the offline stage, the model is built for the set of parameters. It is then used online combined with other models. Online, the use of the model requires no more than reading the unknown value in an abacus.

As previously described, one of the parameters of the problem consists of the source term $b$. Taking into account the source term as a parameter is a challenging task. Indeed, once discretized in space, the source term is made of discrete values: one information per piece of the mesh. It implies inputting as many parameters in the PGD parametric model as the number of pieces of the mesh, plus the boundary conditions and spatial coordinates. The PGD method has shown success for problems up to dimension 100. However, the efficiency of a parametric model depends on the number of involved parameters [13, 16].

To avoid this large number of involved parameters, one can gather some of them. Considering the source term, this is usually done by using an approximation basis. The temperature field is projected on an approximation basis of a smaller size. The use of different approximation basis is investigated in this work: Chebyshev, Legendre polynomials and the POD reduced basis.

The PGD method is combined with the approximation basis to build a PGD parametric model for the previous presented physical problem. Each step of the *offline / online* strategy is described in Figure 2.

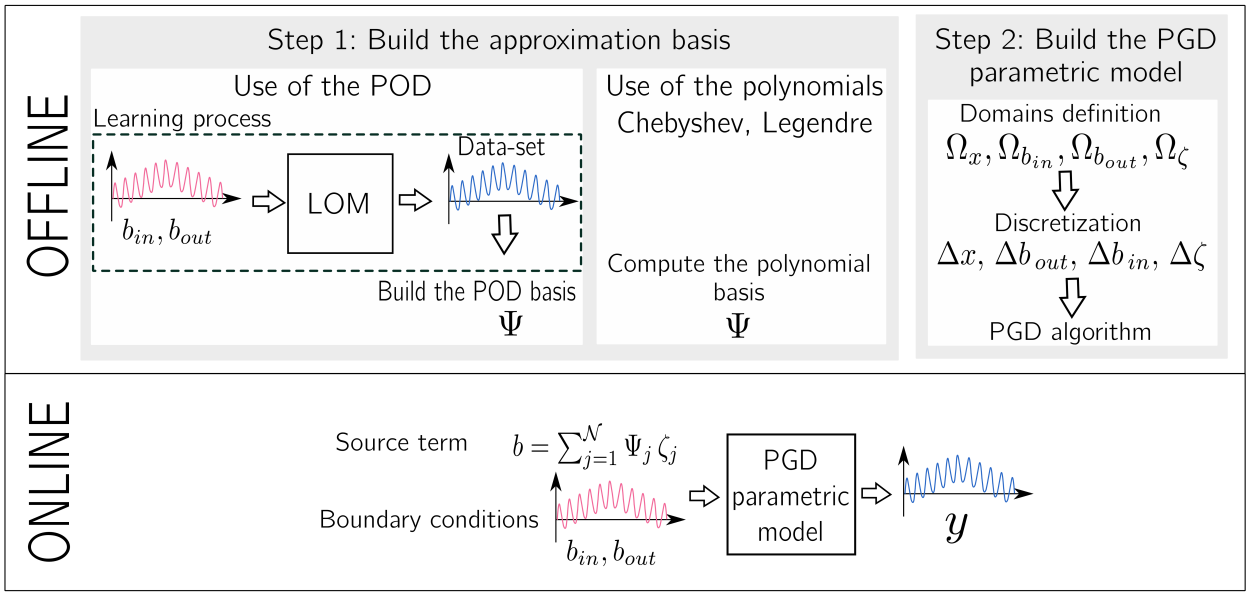



Figure 2: *Offline / online* strategy

The first step of the *offline* phase consists of building an approximation basis. In the case of Chebyshev, Legendre basis, it is made of the polynomials. In the case of the POD reduced basis, a learning process is required. The POD basis is built on a data-set. The latter can be provided from available measurements

or from another model defined as a Large Original Model (LOM). To get an accurate approximation basis, the learning process needs to be representative of all future modeled combinations. In the specific case of a building energy model, the basis should be representative of every material and climate data that could be used. The learning process needs a large amount of data and could be very time-consuming.

The approximation basis $\Psi$ aims at representing the source term in a minimum number of parameters called modes. For that purpose, the approximation basis is truncated. A number of modes in the approximation basis, $\mathcal{N}$, is defined to achieve the desired approximation accuracy. Note that this number has a direct influence on the number of parameters used in the PGD parametric model and its accuracy.

Then, all the parameters of the model (the mesh, the boundary conditions, and the approximation basis modes) are converted into parameter vectors. The discretization ($\Delta x$, $\Delta b_{out}$, $\Delta b_{in}$, $\Delta \zeta$) selected for each vector has an impact on the accuracy of the PGD parametric model.

Finally, as a last step of the *offline* phase, the parametric problem can be solved with the PGD algorithm. The PGD parametric model is built for a number of PGD modes $\mathcal{M}$. This parameter also influences the accuracy of the parametric model.

Once the PGD parametric model has been built, it can be applied for any value within the previously defined intervals, *online*. The source term $b$ is projected on the approximation basis $\Psi$ to identify the parameters $\zeta^j$. Afterwards, the PGD modes are computed for the defined parameters $x$, $b_{in}$, $b_{out}$ and $\zeta^j$. The evaluation of the solution demands no more than reading a look-up table [17].

# 3 Methodology

The purpose of this article is to overcome the obstacle of parameterizing the initial condition of a PGD parametric model. It is then necessary to quantify and compare the accuracy of each approximation basis in the framework of their combination with the PGD. The proposed study will therefore cover several issues:

1. the accuracy of the approximation basis for a given number of modes $\mathcal{N}$,
2. the discretization of each of the parameters vectors,
3. the number of PGD modes $\mathcal{M}$.

For the use of the POD approximation basis, a supplementary issue has to be added: the efficiency of the learning process.

## 3.1 Methods assessment's procedure

To evaluate the approximation basis in several situations, two case studies are presented. The first case study is a theoretical application. It is used to study the influence of the three first issues cited previously.

The built basis are then applied to a practical case with realistic boundary conditions. The results of the models will be confronted with laboratory measurements. The influence of the learning period is studied through this second case study.

They may seem simple and we could have considered more complicated case studies. However, the parametric model would have been more complex. It would then have been more complicated to identify the influence of the studied parameters on the final error of the model.

For each case study, the global methodology consists of two main steps. First, the approximation of the source term is evaluated to study the behavior of the basis alone. Then the PGD parametric model is evaluated to verify if the basis have the same behavior once applied in the PGD framework. The performance of the three basis is compared with regards to the model errors and CPU time. The chosen indicators are presented hereafter.

## 3.2 Error indicator of the model

For each step of the assessment procedure, the error indicator chosen is the $\ell_\infty$ norm. It is computed as the Root Mean Square Error between two spatial profiles. Only the maximum of the previous function is observed. This Section describes the errors calculated for each of the three parameters studied in this paper.

### 3.2.1 Evaluation of the source term approximation

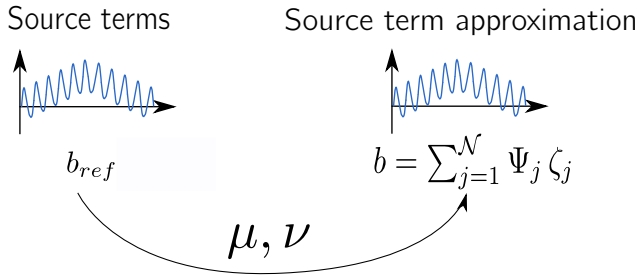



Figure 3: Evaluation of the source term approximation

For each metrics introduced hereafter, Figure 3 summarizes the methodology. First, the performance of each basis to approximate the source terms is evaluated by projecting the source terms (actual $b$ and reference $b_{ref}$) on the different basis and by then calculating the errors $\mu$ as follows:

$$\mu : (\mathcal{N}, \Psi) \mapsto \max_t \left( \sqrt{\frac{1}{\mathtt{N_x}} \sum_0^{\mathtt{N_x}} \left[ b_{ref} - \sum_{j=1}^{\mathcal{N}} \Psi_j \zeta_j \right]^2} \right) \tag{15}$$

where $\mathtt{N_x}$ is the number of elements over the axis. The reference source term (noted $b_{ref}$ ) outcomes from the reference solution calculation at each point of the spatial mesh and for each studied time step. The error is calculated for each approximation basis $\Psi$. The influence of the parameter $\mathcal{N}$ on the accuracy of the basis $\mu$ will be studied.

To integrate the approximation basis into the PGD framework, each parameter of the model has to be discretized. The error due to this discretization noted $\nu$ is evaluated for each approximation mode. The error is calculated as follows for the PGD variable $\zeta$:

$$\nu : (\mathcal{N}, \Psi, \zeta, \Delta\zeta) \mapsto \max_t \left( \sqrt{\frac{1}{\mathtt{N_x}} \sum_0^{\mathtt{N_x}} \left[ b_{ref} - \sum_{j=1}^{\mathcal{N}} \Psi_j \overline{\zeta_j} \right]^2} \right) \tag{16}$$

with $\overline{\zeta}$ the dimensionless coefficients.

### 3.2.2 Evaluation of the PGD parametric model

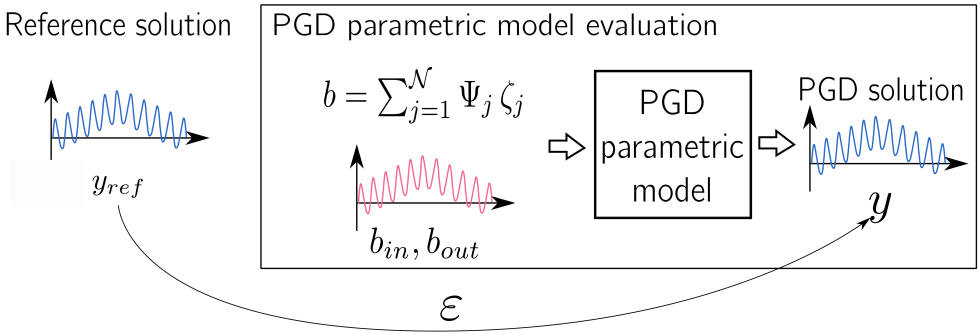



Figure 4: Methods assessment's procedure sum-up

Finally, the approximation basis are introduced into a PGD parametric model to get a combined parametric model. The error of the combined model noted $\varepsilon$ is computed between the calculated temperature profile and the reference solution. Figure 4 summarizes the methodology.

$$\varepsilon : \mathcal{N},\, \zeta,\, \Psi,\, \Delta\zeta,\, \mathcal{M} \mapsto \max_{t} \left( \sqrt{\frac{1}{\mathtt{N_x}} \sum_{0}^{\mathtt{N_x}} \left[ y_{ref} - \sum_{m\,=\,1}^{\mathcal{M}} X_m\left(x\right) E_m\left(b_{in}\right) F_m\left(b_{out}\right) \prod_{j\,=\,1}^{\mathcal{N}} G_m^j\left(\zeta^j\right) \right]^2} \right) \tag{17}$$

## 3.3 Indicator for the CPU time

A fair comparison of the computational time for various methods is not easy to undertake as it will depends on the way we code and the tools we use. To get a fair comparison, calculation times were measured on the same computer and on the same environment. Except for the reference solution calculation (where the `Matlab` toolbox `Chebfun` has been used), we developed all the other computational codes by ourselves. We paid attention to code each model in the same way (for example the same algorithm is always used to solve a system of equation).

For each step, the CPU calculation time is evaluated on a Lenovo, windows 10 with 8Go RAM IntelCore i5, 2.60 GHz. The CPU calculation time is normalized by the time constant $t_0$. It corresponds to the maximum CPU time observed. This information will be given in the titles of the figures. The CPU time ratio is noted $\rho_{\text{CPU}}$ and defined as follows:

$$\rho_{\text{CPU}} = \frac{t_{CPU}}{t_0} \tag{18}$$

# 4 Theoretical case study

## 4.1 Description of the case study

### 4.1.1 Physical constants used

The case study consists of a wall of one-layer of thickness $\mathrm{L} = 0.20\,\mathsf{m}$, made of concrete, with a thermal conductivity $\mathrm{k} = 1.75\,\mathsf{W.m^{-1}.K^{-1}}$ and a specific heat capacity $\mathrm{c} = 2.2\,10^{6}\,\mathsf{J.m^{-3}.K^{-1}}$.

On the outdoor side of the wall, a sinusoidal variation of the air temperature and the net radiative heat flux are considered. Their variations are defined as:

$$\mathrm{u_{out}} = \mathrm{u_{o,m}} + \delta_{\mathrm{o},1}\,\sin(2\,\pi\,\omega_{\mathrm{o},1}\,\mathrm{t}) + \delta_{\mathrm{o},2}\,\sin(2\,\pi\,\omega_{\mathrm{o},2}\,\mathrm{t}) \tag{19}$$

$$\mathrm{q} = \mathrm{q_m}\,\sin(2\,\pi\,\omega_{\mathrm{q},}\,\mathrm{t})^{20} \tag{20}$$

On the indoor side of the wall, a sinusoidal variation of the air temperature is considered, as described below:

$$\mathrm{u_{in}} = \mathrm{u_{i,m}} + \delta_{\mathrm{i}}\,\sin(2\,\pi\,\omega_{\mathrm{i},1}\,\mathrm{t}) \tag{21}$$

As presented before, the net radiative heat flux is neglected on that side of the wall. The error due to this simplification of the mathematical model is studied for the specific case study in the appendix A.

The following numerical values are considered for the outdoor and indoor boundary conditions:

$$\mathrm{u_{o,m}} = 20\,[^{\circ}\mathsf{C}],\ \delta_{\mathrm{o},1} = -4.4\,[\mathsf{K}],\ \omega_{\mathrm{o},1} = \frac{1}{72}\,[\mathsf{h}^{-1}],\ \delta_{o,2} = -11.7\,[\mathsf{K}],\ \omega_{o,2} = \frac{1}{24}\,[\mathsf{h}^{-1}],$$

$$\mathrm{q_m} = 500\,[\mathsf{W.m^{-2}}],\ \omega_{q,} = \frac{1}{48}\,[\mathsf{h}^{-1}],\ \mathrm{u_{i,m}} = 20\,[^{\circ}\mathsf{C}],\ \delta_{i} = -2.0\,[\mathsf{K}],\ \omega_{i,1} = \frac{1}{48}\,[\mathsf{h}^{-1}].$$

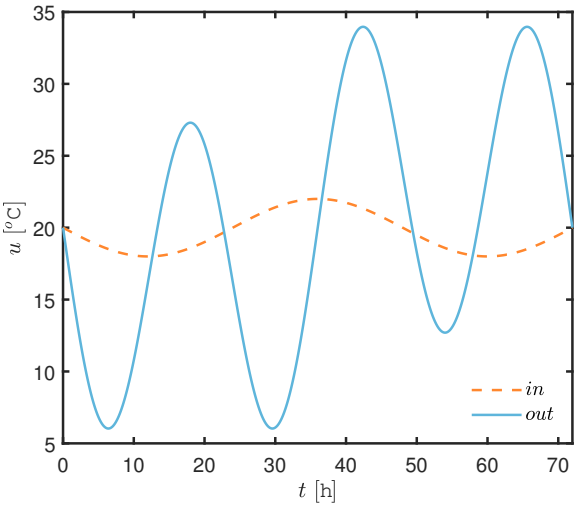
Some of the numerical values are inspired from 1D numerical application [2]. The boundary conditions used are presented in the Figure 5. The convective heat transfer coefficients are set to : $\mathrm{h_{in}} = 8.7\,\mathsf{W.m^{-2}}.K^{-1}$ and $\mathrm{h_{out}} = 23.2\,\mathsf{W.m^{-2}.K^{-1}}$.

The numerical values of the dimensionless quantities are the following ones:

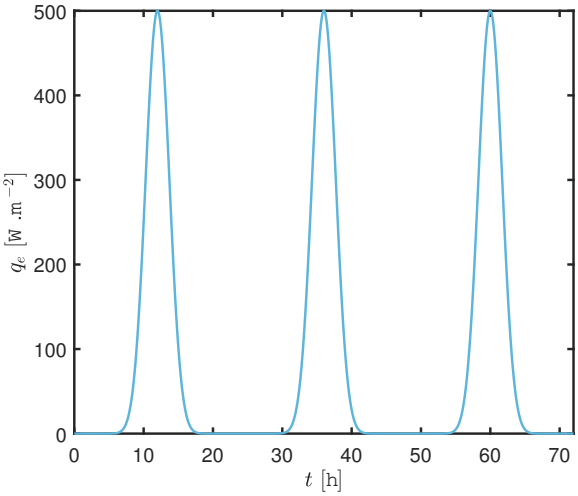
$$Bi_{in} := 0.4971; \quad Bi_{out} := 1.3314; \quad Fo := 1; \quad \mathrm{t_{ref}} := 1.2571 \times 10^{4}$$

### 4.1.2 Reference solution

The reference solution $y_{ref}(x,t)$ is computed using the `Matlab` toolbox `Chebfun` [24] for a time horizon of 3 days, with a dimensionless time step of $\Delta t = 10^{-3}$ and a space mesh made of 200 nodes. The evolution of the temperature for the reference solution is presented in Figure 6. Figure 6(a) describes the temporal evolution of the surface temperature on each side of the wall and figure 6(b) gives an overview of the temperature profiles within the wall. It represents the source term that needs to be parameterized with the several studied approximation basis.

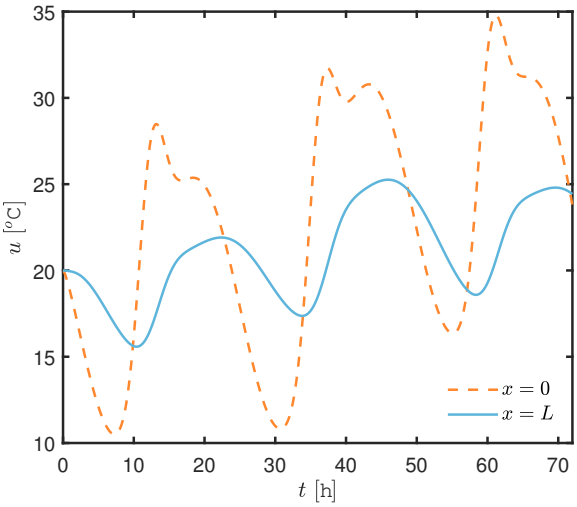



(a) Inside and outside air temperature signal

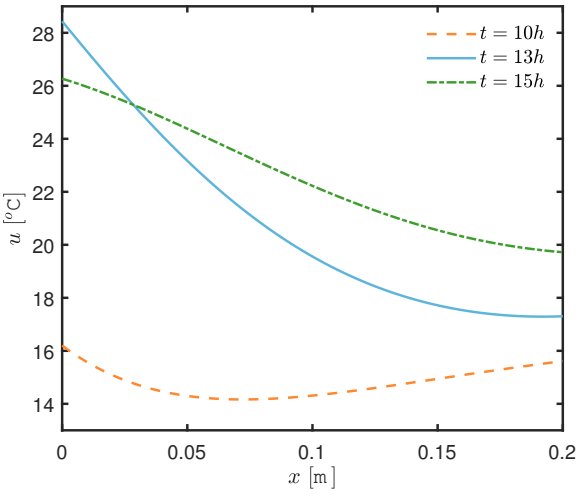



(b) Net radiative heat flux

Figure 5: Boundary conditions of the theoretical case study




(a) Indoor and outdoor surface temperature time series




(b) Temperature profiles inside the wall

Figure 6: Temperature field for the reference solution

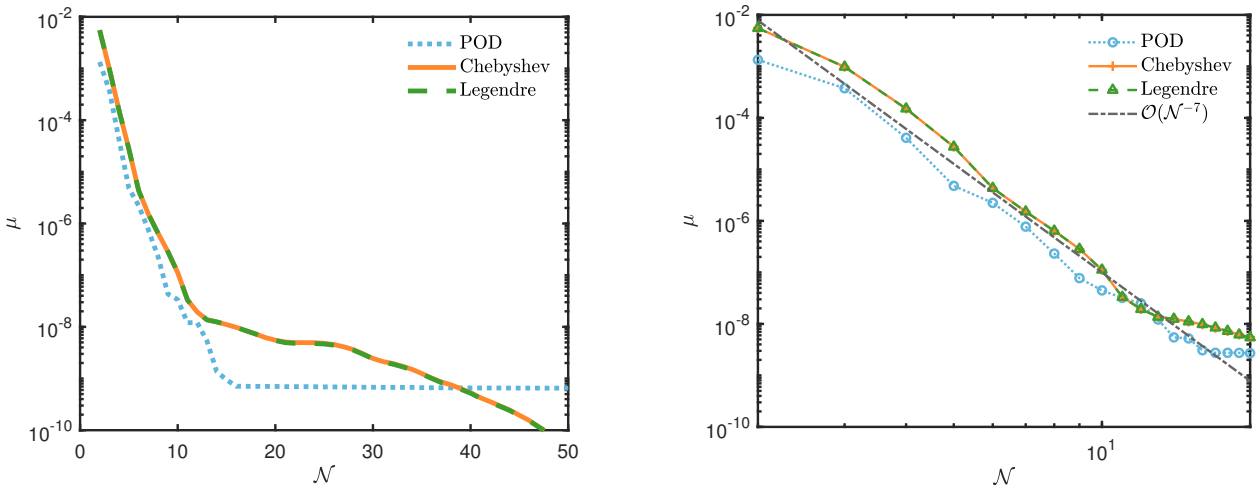
### 4.1.3 Learning process

As presented in the Section 2.6, the POD basis is built on an available data-set. The choice was made to use the complete reference solution data-set to built the POD basis. The POD basis is then used in identical conditions than the one used for the learning process. Thus, the condition of the learning process will not influence the accuracy of the basis.

## 4.2 Evaluation of the approximation of the source term

The ability of each basis to approximate the source term depends on two parameters : the number of modes in the approximation basis $\mathcal{N}$ and the discretization of the parameters. The influence of those two criteria is studied hereafter.

### 4.2.1 Influence of the number of modes in the approximation basis




(a) Semi-logarithmic scale

(b) Logarithmic scale

Figure 7: Evolution of the error $\mu$ as a function of the number of modes in the approximation basis

Figure 7 presents the evolution of the approximation error as a function of the number of modes $\mathcal{N}$ in the three approximation basis. In Figure 7(a), we can observe that the error decreases as the number of modes in the approximation basis increases. In the case of the POD basis, the error decreases until it gets constant around $\mathcal{N} = 18$. The results of the Chebyshev and the Legendre polynomial basis are very close. They both decrease with a large slope for the first ten modes and continue to decrease slowly. The polynomial basis cross the POD basis around $\mathcal{N} = 38$ modes. The polynomial basis are then more accurate than the POD one.

The smoothness of the function can be linked to the number of times the function is differentiable. As explained by Trefethen (2013 [5]), the smoother a function, the faster its approximates converge. Figure 7(b) gives information on the smoothness of the function. In this specific case, the three approximation basis have similar trends. They converge at a rate of $\mathcal{O}(\mathcal{N}^{-7})$.

The CPU calculation time is another criterion to compare the performance of the three basis. CPU time ratios are presented on Figure 8. The results are normalized by the maximum CPU time observed (for

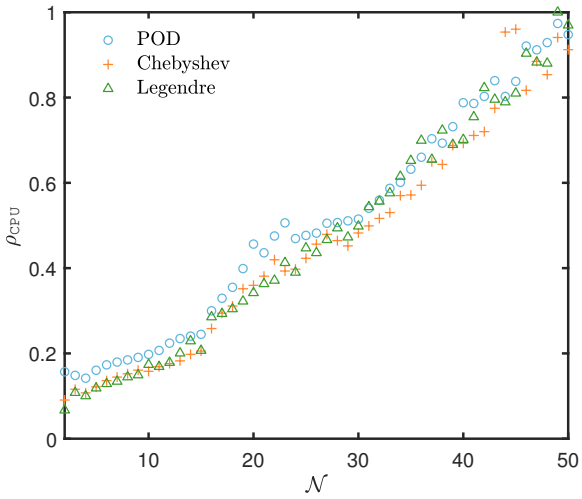



Figure 8: Evolution of the CPU calculation time as a function of the number of modes in the three approximation basis with $t_0 = 0.2341\,\text{sec}$

LEGENDRE basis with $\mathcal{N} = 50$). The calculation time presented for the POD basis includes the learning process. We can observe that the CPU time increases linearly and that it is slightly higher for the POD basis than for the two polynomial basis. However, the results are of the same order of magnitude.

#### 4.2.2 Influence of the discretization

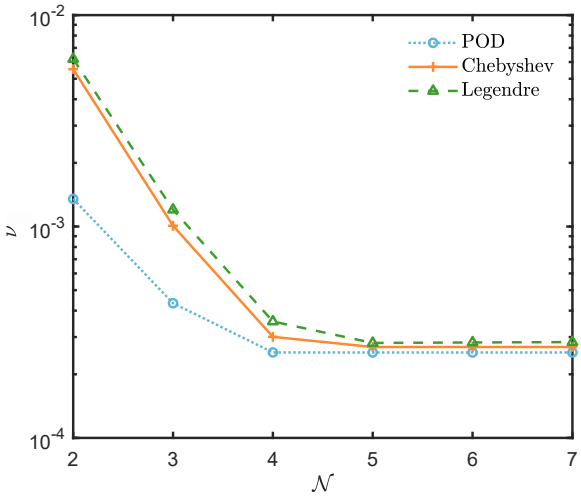
As reported in the Section 2.5, each parameter of the model (the mesh, the boundary conditions, and the coefficients of the approximation basis) needs to be converted into vectors of parameters. For that purpose, their domain needs to be discretized, by converting the continuous functions into discrete values. The mesh of discretization of the parameter $\zeta$ has a direct impact on the accuracy of the approximation of the source term. The influence of the mesh of discretization $\Delta\overline{\zeta}$ is studied hereafter.

Three dimensionless discretizations have been selected $\Delta\overline{\zeta} = 10^{-2}, 10^{-4}, 10^{-6}$. For the three criteria, the error between the reference solution and the solution projected on the truncated basis is plotted as a function of the number of modes in the truncated basis. Figure 9 presents the results. For each curve, the same tendencies can be observed, the error drops and then stabilizes. Indeed as we increase the number of modes, the error of the approximation decreases. However, as the coefficients are rounded, part of the information is lost. As the error stabilizes, the addition of a supplementary mode does not improve the accuracy of the approximation. For a discretization $\Delta\overline{\zeta} = 10^{-2}$, the threshold is reached for $\mathcal{N} = 5$ and 7 and 12 modes for respectively: $\Delta\overline{\zeta} = 10^{-4}$ and $\Delta\overline{\zeta} = 10^{-6}$.

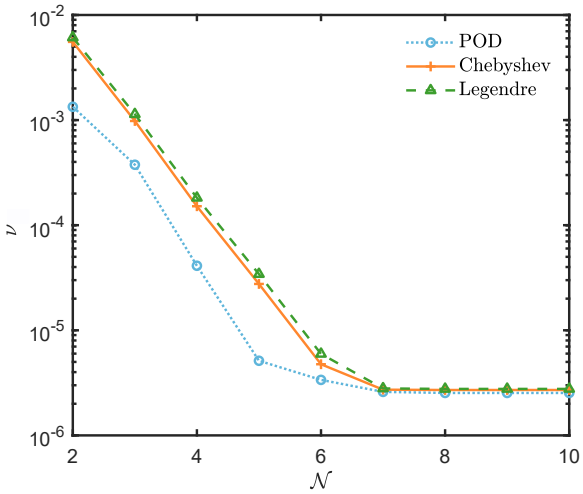
#### 4.2.3 Discussion

From those two first influence analyses, the approximation basis can not be ranked, as their performances are close. For both the number of modes $\mathcal{N}$ and the discretization, the same tendencies can be observed for the three approximation basis.

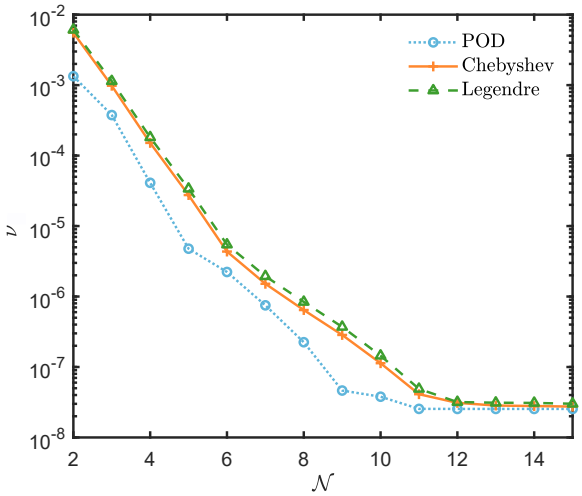
Moreover, the accuracy of the POD basis depends on the quality of the learning process (it should be representative of the conditions of future study cases). In the theoretical case study, the learning process has




(a) Evolution of the error for $\Delta\overline{\zeta} = 10^{-2}$




(b) Evolution of the error for $\Delta\overline{\zeta} = 10^{-4}$




(c) Evolution of the error for $\Delta\overline{\zeta} = 10^{-6}$

Figure 9: Influence of the truncation of the approximation basis on the accuracy of the approximation for various discretizations

been made on the complete reference solution data-set. We are then in ideal conditions for the use of the POD basis. In the practical application (Section 5), the influence of the learning process will be investigated.

This first step enables the comparison of the behavior of the three studied basis outside of the PGD framework. However, once implemented in the PGD framework, the tendencies observed before could be different. to verify the consistency, the influence of the parameters studied should be studied in the PGD framework.

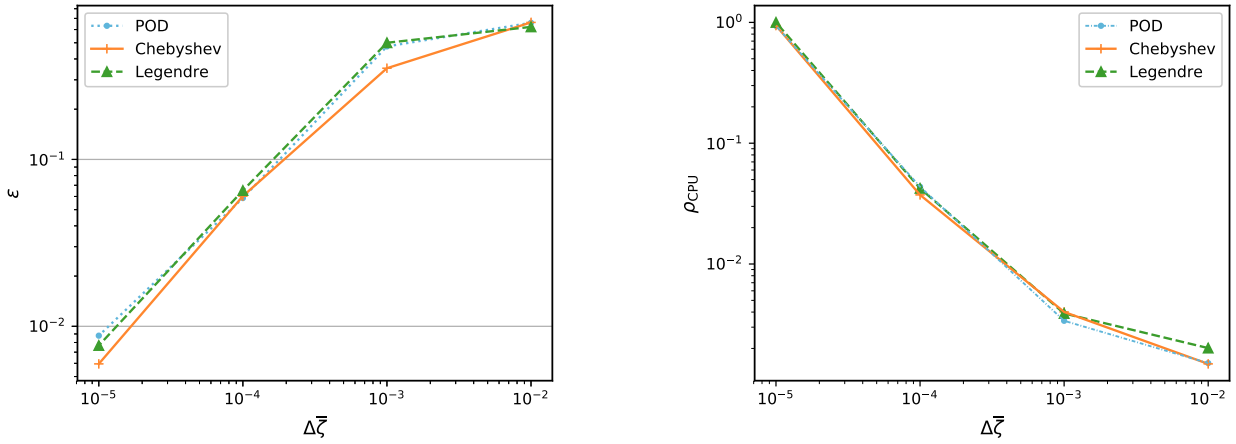
## 4.3 Evaluation of the PGD parametric model

A PGD parametric model is built to solve the problem studied here. The boundary conditions and the source term are defined as parameters of the parametric model. The approximation basis are used to describe the initial condition in a few parameters (modes). The PGD model is then combined to an approximation basis. The accuracy of the combined model depends on three parameters:

- the accuracy of the approximation basis for a given number of modes $\mathcal{N}$,
- the discretization of each of the parameters vectors,
- the number of PGD modes $\mathcal{M}$.

To study the influence of those three parameters on the accuracy of the model, several PGD basis have been generated, one for each: combination of the three approximation basis, number of modes in the basis $\mathcal{N} \in [2, 5]$ and discretization $\Delta\overline{\zeta} \in [10^{-5}, 10^{-2}]$. In total 48 PGD combined models have been compared. For each model, both parameters of the alternating direction process and the enrichment process are fixed to $\tilde{\epsilon} = 10^{-6}$ and $\epsilon = 10^{-8}$. The influence of each parameter $\Delta\overline{\zeta}$, $\mathcal{N}$, and $\mathcal{M}$ is studied hereafter, based on the results of those basis.

### 4.3.1 Influence of the discretization of the approximation coefficient



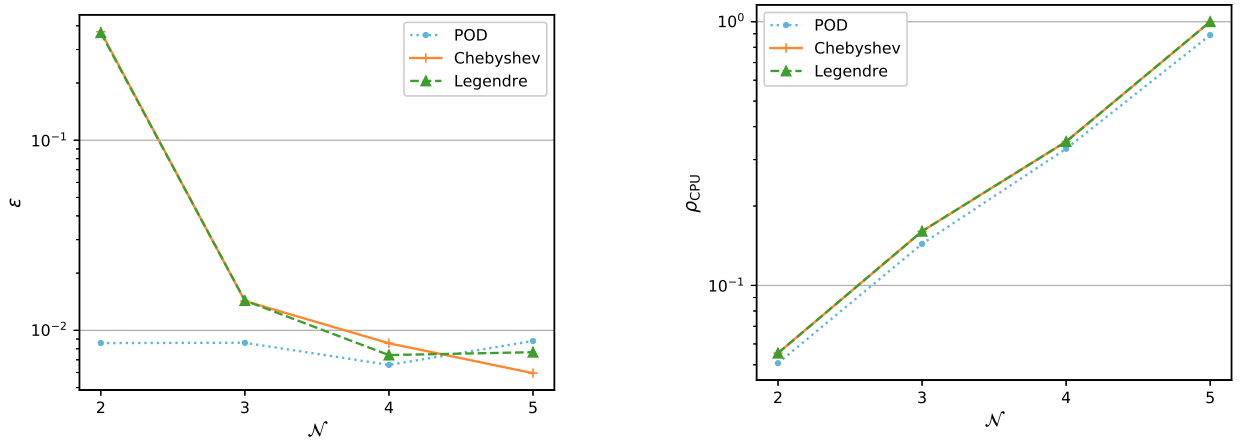

(a) $\varepsilon_\infty$ error for $\mathcal{N} = 5$

(b) Online CPU calculation time ratio for $\mathcal{N} = 5$

Figure 10: Influence of the discretization of the approximation coefficients on the error and on the CPU time ratio with $t_0 = 23879\,\text{sec.}$

The influence of the approximation coefficients is studied here. Figure 10 presents the results of the error and about the calculation time. The results are displayed for the most accurate basis used made of $\mathcal{N} = 5$ approximation modes. The accuracy of the model and the CPU time of each model increase, as the discretization gets finer. Those two results are in accordance with the previous ones. The discretization induces a loss of information. The continuous function is converted into discrete values as it is done for a spatial mesh for any numerical method. The finer the mesh, the closer the discrete representation to the continuous function. However, as we increase the discretization, we increase the number of elements in the vector. The online CPU time then increases. The same tendencies are observed for basis made of 3 and 4 modes. For basis made of 2 modes, the same tendencies are observed for the POD. However for the Chebyshev and Legendre basis, the error remains high and constant as we decrease the discretization. For both polynomial basis, using 2 modes is not enough to approximate the source term accurately.

### 4.3.2 Influence of the number of modes in the approximation basis




(a) For $\Delta\overline{\zeta} = 10^{-5}$, evolution of the $\varepsilon$ error

(b) For $\Delta\overline{\zeta} = 10^{-5}$, evolution of the CPU calculation time ratio

Figure 11: Influence of the number of modes in the approximation basis on the $\varepsilon$ error and CPU time ratio with $t_0 = 23879\,\text{sec}$

The influence of the number of modes in the approximation basis is now studied. Figure 11(a) presents the evolution of the $\varepsilon$ error as a function of this parameter. Results are presented for a fixed discretization of $\Delta\overline{\zeta} = 10^{-5}$ for each approximation basis. In the case of the Chebyshev and Legendre combined parametric models, the error decreases with the number of modes. This phenomenon can be observed for fine discretizations ($\Delta\overline{\zeta} = 10^{-4}$ or $\Delta\overline{\zeta} = 10^{-5}$). For coarser discretizations, the error remains constant as we increase the number of modes. Adding a supplementary mode is not useful if the discretization remains constant.

In the case of the POD combined parametric model, for a fixed discretization, adding a supplementary mode will not decrease the error of the model. The discretization will only have an impact on the error of the model. Here, the model is trained and used on the same data-set. The results may have been different if only part of the data-set has been used to train the basis. This point will be illustrated in the practical application (Section 5).

For every model, a threshold around $\mathcal{O}(10^{-3})$ is reached after a few modes. The error of the final PGD model is then not mainly due to the approximation of the source term but also to other parameters: the discretization of the boundary condition on x = L fixed at $10^{-3}$, the discretization of the boundary condition on x = 0 fixed at $10^{-4}$, the spatial grid fixed at $10^{-2}$.

### 4.3.3 Influence of the number of modes in the PGD basis

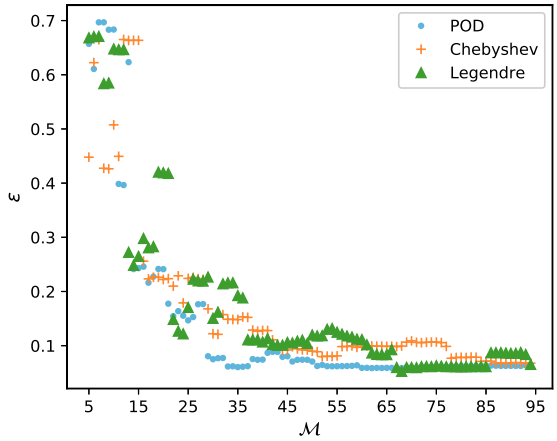



Figure 12: Influence of the number of PGD modes $\mathcal{M}$ on the $\varepsilon$ error for $\delta\overline{\zeta} = 10^{-4}$ and $\mathcal{N} = 4$.

The last parameter studied is the influence of the number of PGD modes $\mathcal{M}$. Figure 12 presents the evolution of the error as a function of the number of PGD modes $\mathcal{M}$ for $\mathcal{N} = 4$. The parametric models are built for $\Delta\overline{\zeta} = 10^{-4}$ for each approximation basis. This Figure gives information on how fast the PGD strategy converges. There are not many differences between the three methods. Applied to non-symmetric differential operators, the PGD algorithm converges slowly as its optimality is not guaranteed [11]. The PGD could contain more terms than strictly needed.

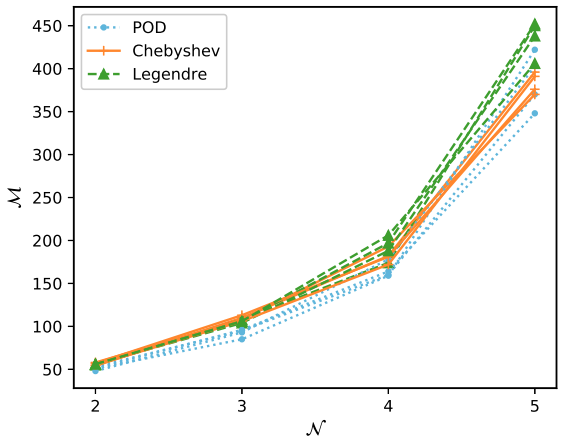
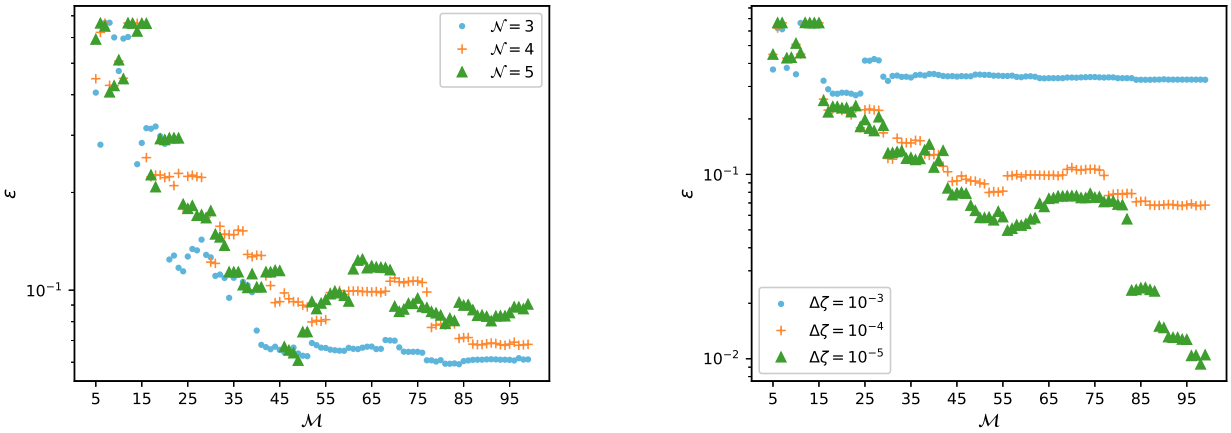



Figure 13: Evolution of the total number of PGD modes as a function of the number of modes of the approximation basis. Each curve of each basis corresponds to a different discretization $\Delta\overline{\zeta} \in [10^{-5}, 10^{-2}]$

Each time a mode is added to the parametric model, a new variable is added to the problem. The computational domain becomes of higher dimension, it must cover not only the physical and boundary conditions coordinates but also the parametric domain [25]. Adding a parameter increases the complexity of the tensor subspace. In the case of the PGD, this complexity grows linearly with the number of dimensions [17, 22]. Figure 13 illustrates the impact of adding a new parameter to the PGD parametric model on the total number of PGD modes. As we increase the number of modes in the approximation basis, we increase the number of parameters in the parametric model. As soon in Figure 14 the convergence rate of the algorithm decreases. Thus, the number of necessary PGD modes increases to achieve the desired accuracy ($\tilde{\epsilon} = 10^{-6}$ and $\epsilon = 10^{-8}$) as we increase the number of parameters involved.




(a) For $\delta = 10^{-4}$, evolution of the $\varepsilon$ error

(b) For $\mathcal{N} = 3$, evolution of the $\varepsilon$ error

Figure 14: Evolution of the $\varepsilon$ error as a function of the number of modes for the Chebyshev basis

### 4.3.4 Discussion

For the approximation coefficients discretization and the number of modes in the approximation basis, the same tendencies are observed than the one observed for the approximation of the source term. The comparison of the approximation basis on the approximation of the source term gives a good first overview of the behavior of the basis.

However, two modes are not sufficient to approximate the source term with Chebyshev and Legendre combined parametric models. For the POD basis, the final accuracy of the PGD model is reached with two modes for a fixed discretization.

Finally, Leon *et al.* (2018, [23]) have shown on the Poisson equation that the final accuracy of a PGD model depends on the discretization of the parameters and the number of terms $\mathcal{M}$ in the final sum. Indeed the finer the discretization of each parameter, the closer will be the discrete values to the continuous one. However, as they decrease the mesh, they increase the convergence rate of the PGD algorithm and the necessary number of PGD modes in the model. The same tendencies can be observed here in Figure 14(b). It presents the evolution of the $\varepsilon$ error as a function of the number of modes for the Chebyshev basis with $\mathcal{N} = 3$. The error decreases and then reaches a threshold. Then adding a supplementary PGD mode to the parametric model is not sufficient to decrease the error of the model. The discretization should be decreased.

# 5 Practical application

In the previous parts, the POD basis, as most of the time, has shown its optimality. However, as mentioned above, the performance of the POD basis depends on the quality of the learning process. It should be representative of the boundary conditions applied to the case study.

In the theoretical case study, the learning process has been made on the complete reference solution data-set. The POD basis is then used in identical conditions than the one used for the learning process. The influence of the learning process has not been studied yet.

To obtain a POD basis, a training data-set is necessary. It can be obtained from measurements or from another numerical model. Both methods are expensive since a large range of data is needed. To give an example, if we want to use the parametric model to predict the temperature distribution in a wall during a year, the training data-set should be representative of all the boundary conditions encountered in practice.

To illustrate this limit, the accuracy of various POD basis are compared to the polynomial basis. The same methodology as the one used for the theoretical case study is applied. The influence of the learning period is first studied on the approximation itself and then on the combination of the approximation basis with the PGD parametric model.

Another major objective of this part is to evaluate the reliability of the model in a realistic case study. For that purpose, the results of the model are compared to laboratory measurements.

## 5.1 Description of the case study

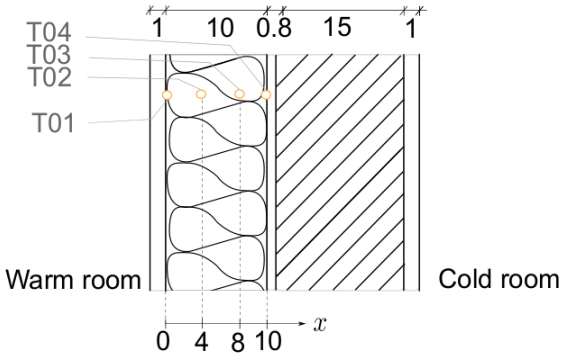



Figure 15: Sensors position illustration and nomenclature

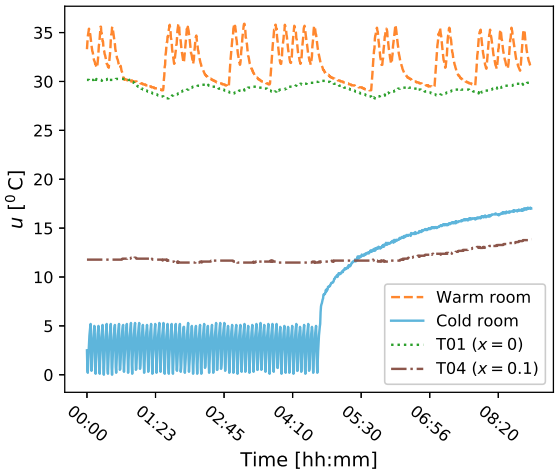



Figure 16: Evolution of the air temperature and inside wall temperatures measurements

### 5.1.1 Experimental set-up

The experimental set-up described hereafter was think up with the objective to obtain realistic boundary conditions and measurements on a common building wall. It consists of a multi-layer building wall, made

of traditional building materials: 1 cm of plasterboard, 10 cm of insulation (expanded polystyrene), 15 cm of structural material and approximately 1 cm of mineral coating. The wall is built between two rooms. One can be heated by an electric heater and the second one can be cooled by the evaporator of a heat pump.

For this study, we will only focus on the insulation layer of the wall. Indeed, the insulation material experiences greater temperature gradients which makes it more interesting to observe. Moreover, the insulation material is a homogeneous material and the temperature is easier to measure in such a material, contrary to the structure material made of concrete cellular blocks for which the measured temperature is strongly dependent on the position of the sensor. Indeed, for such cellular materials, the measured temperature can be very different whether it is measured on a cavity or near the wall of this cavity that creates a thermal bridge. By the more, the thermal conductivity of the insulation material is well known, whereas only the macroscopic thermal resistance is known for the concrete cellular block, which makes it difficult to obtain a calculated temperature directly comparable to the measured temperature, although the heat flux is correct. This insulation layer is thus equipped with four type K thermocouples located at the surface and in the insulation layer.

The global experimental uncertainty has been calculated with equation 22 [26].

$$\sigma = \sqrt{\sigma_m^2 + \left(\frac{\partial u}{\partial x}\,\delta x\right)^2} \tag{22}$$

The thermocouples have been calibrated by measuring the temperature of melting ice and boiling water before the measurement. The sensor measurement uncertainty is then $\sigma_m = \pm 0.1^\circ\,\mathsf{C}$. The sensor position uncertainty has been evaluated as the product of the temperature derivative (with second-order centered approximation) at the sensor position and $\delta x = \pm 0.1\,\mathsf{cm}$. The temporal mean global experimental uncertainty is noted hereafter $\overline{\sigma}$.

### 5.1.2 Experimental observations

Data were recorded for 5 days, with a 30 sec time step. Several cycles were tested during this period, turning on and off the heater and/or the heat pump. The cycles are described in Table 1. A pattern made of three cycles with three different time periods (25 min, 40 min, and 60 min) is repeated twice. The first three cycles are run with a temperature set-point of $5^\circ\mathsf{C}$ in the cold room. For the last three cycles, the heat pump was turned off to modify the boundary conditions of the cold room. The boundary conditions are described through the evolution of the air temperature in the warm and cold room in Figure 16.

### 5.1.3 Reference solution

The reference solution $y_{ref}(x,t)$ of this problem is computed using a Euler implicit finite difference model for a time horizon of 96 hours with a time step of 30sec (dimensionless time step of $\Delta = 10^{-2}$) and a spacial mesh made of 99 nodes. On each side, two Dirichlet boundary conditions are set using the temperature signal $T01$ and $T04$. Figure 16 gives an overview of the temperature evolution at the boundary conditions.

The first temperature profile is initialized using the temperature profile measured in the wall at the beginning of the experiment. Linear interpolation is done between the measured points to obtain the temperature distribution at each point of the spatial mesh (Figure 17). The simulation is then run for the all period (5

| Cycle number | Heater | Heat Pump | Duration [min] | Time |
|---|---|---|---|---|
| Initialization | on | on (5°C) | 5220 (87h) | - |
| 0 | off | on (5°C) | 40 | 00:00 to 00:40 |
| 1 | on | on (5°C) | 40 | 00:40 to 1:20 |
| 1 | off | on (5°C) | 40 | 1:20 to 2:00 |
| 2 | on | on (5°C) | 25 | 2:00 to 2:25 |
| 2 | off | on (5°C) | 25 | 2:25 to 2:50 |
| 3 | on | on (5°C) | 60 | 2:50 to 3:50 |
| 3 | off | on (5°C) | 60 | 3:50 to 4:50 |
| 4 | on | off | 40 | 4:50 to 5:30 |
| 4 | off | off | 40 | 5:30 to 6:10 |
| 5 | on | off | 25 | 6:10 to 6:35 |
| 5 | off | off | 25 | 6:35 to 7:00 |
| 6 | on | off | 60 | 7:00 to 8:00 |
| 6 | off | off | 60 | 8:00 to 9:00 |

Table 1: Description of the cycles

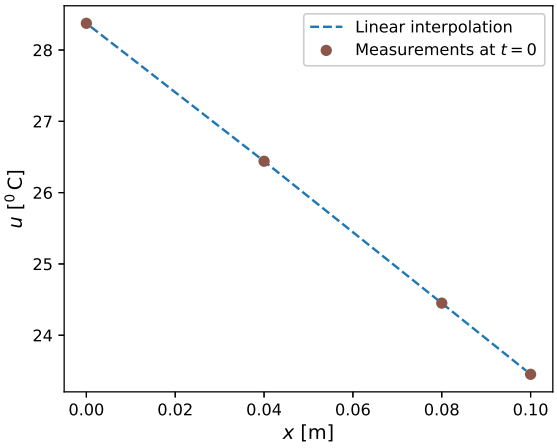



Figure 17: Initialization temperature profile

days). The first 87h of the simulation are not used. They are left as initialization period of the model. It consists of turning on the heater and the heat pump until an equilibrium between the two rooms is reached. The boundary conditions of this initialization cycle are described in Table 1. The rest of the data-set is used to evaluate the model in different conditions. As the first 87h are not used to evaluate the model, they are not presented in the following figures. Thermal properties from the French regulations database [27] are used with: $k = 0.04\,\mathsf{W.m^{-1}.K^{-1}}$ and $c = 30.10^3\,\mathsf{J.m^{-3}.K^{-1}}$ .

#### 5.1.4 Learning process

Three training data-sets for the POD basis are compared:

1. the full evaluation data-set (noted $t \in \Omega_\tau = [0\,,\,\tau]$ with $\tau = 9\mathsf{h}$),
2. half of the evaluation data-set, made of the cycles 0 to 3 (noted $t \in \Omega_{\frac{\tau}{2}} = [0\,,\,4\mathsf{h}50]$),
3. the cycle 1 (noted $t \in \Omega_{t_1} = [0\,,\,0\mathsf{h}40]$).

The three basis are compared to the Chebyshev and Legendre polynomial basis. For those two last methods, no learning period is required to build the basis.

### 5.2 Influence of the learning period

As for the previous case study, the influence of the learning period is first evaluated on the approximation of the source term, then it is evaluated on the PGD parametric model. The same parameters and computation code than the one used for the previous sections are applied.

#### 5.2.1 Evaluation of the approximation of the source term

The accuracy of the approximation of the source term is studied for various training data-sets. The results are presented on figure 18. The error is plotted for the various number of modes in the approximation basis: $\mathcal{N} \in [2, 8]$.

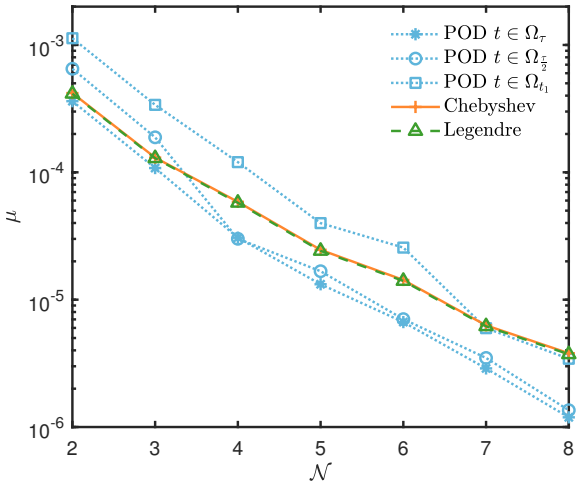



Figure 18: Approximation basis error as a function of the number of modes for several training periods

As in the theoretical example, the POD basis is the most accurate one for $\mathcal{N} \in [2, 8]$, if the full data-set is used for the training period. However if only a part of the data is available, the Chebyshev and Legendre approximation basis are more efficient for $\mathcal{N} \in [2, 3]$. The POD basis trained with half of the cycles seems to be as efficient as the one built with the full training data-set for $\mathcal{N} \in [4, 8]$. Indeed the same pattern is repeated from cycle 1-2-3 to cycle 4-5-6. Building the POD basis with one pattern could be enough for $\mathcal{N} > 3$.

This learning process has a numerical cost as it requires running a large original model and building the POD basis as described in Section 2.6. Table 2 compares the computation time needed to build the basis from the results of the finite difference model for the various learning periods. As large is the training data-set as large is the time needed. Building the basis with one cycle results in a saving of 35% of the offline computation cost.

| Learning period | $\rho_{CPU}$ |
|---|---|
| $t \in \Omega_\tau$ | 1 |
| $t \in \Omega_{\frac{\tau}{2}}$ | 0.89 |
| $t \in \Omega_{t_1}$ | 0.65 |

Table 2: Offline calculation time $t_0 = 0.003987\,\mathsf{sec}$

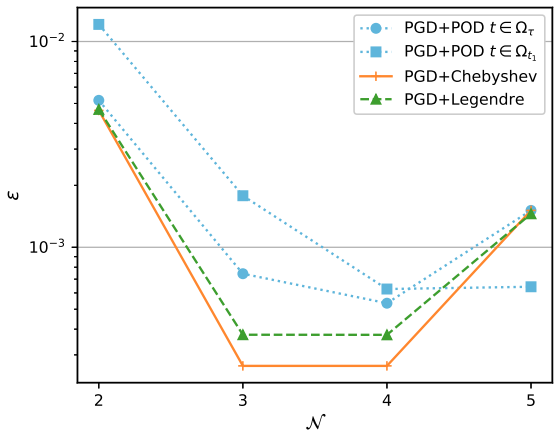
Finally, a compromise should be found to minimize the training period and the computational cost needed to build the basis while keeping an accurate approximation basis. For that purpose, a methodology to select an efficient training period should be developed.

### 5.2.2 Evaluation of the PGD parametric model

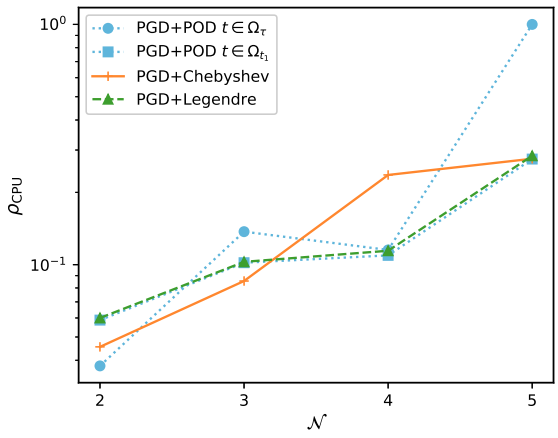
The influence of the learning period is now studied for the combination of the PGD parametric model with the various approximation basis. Results for the most favourable ($t \in \Omega_\tau$) and unfavourable ($t \in \Omega_{t_1}$) POD basis are compared to the Chebyshev and Legendre polynomial basis. Several PGD basis have been generated one for each: combination of the four approximation basis (the favourable POD, the unfavourable POD, the Chebyshev and Legendre polynomial basis), number of modes $\mathcal{N} \in [2, 5]$ and discretization $\Delta\overline{\zeta} \in [10^{-5}, 10^{-4}]$. In total 32 PGD basis have been compared for this application. As done before, both parameters of the alternating direction process and the enrichment process are fixed to $\tilde{\epsilon} = 10^{-6}$ and $\epsilon = 10^{-8}$. The accuracy of the PGD parametric model is compared for various number of modes $\mathcal{N} \in [2, 5]$ and for a fixed discretization $\Delta\overline{\zeta} = 10^{-5}$. We use the same parameters than in section 4.3.2 to compare the results.

Figure 19 presents the evolution of the error and CPU time for various numbers of modes $\mathcal{N}$. For each combined parametric model, for $\mathcal{N} \in [2, 4]$ the error decreases with the number of modes. The error for $\mathcal{N} = 5$ increases. This can also be observed in Figure 11. As previously explained, a threshold (around $\mathcal{O}(10^{-3})$) is reached after a few modes. This phenomenon can be observed for both discretizations. One this threshold has been reached, the error of the final PGD model is then not mainly due to the approximation of the source term. This could explain the fact that the error slightly increases.

In the theoretical case study, the error of the POD basis remained constant with the number of modes. Adding supplementary modes did not improve the total accuracy of the model. It is not the case here. For more complex boundary conditions (realistic signal), supplementary modes are necessary to accurately parametrize the previous temperature profile.




(a) For $\Delta\bar{\zeta} = 10^{-5}$, evolution of the $\varepsilon$ error




(b) For $\Delta\bar{\zeta} = 10^{-5}$, evolution of the CPU time ratio

Figure 19: PGD parametric model $\varepsilon$ error and CPU time ratio as a function of the number of modes for several training periods ( $t_0 = 51.31\,\mathsf{sec}$)

For this practical example, the CHEBYSHEV and LEGENDRE polynomial basis are more accurate once combined with the PGD basis for a similar computational time. This could be due, once more, to the complexity of the boundary condition signal. It could be also due to the discretization $\Delta\bar{\zeta}$. To encounter the same method ranking as the one presented in Figure 18, the POD coefficients may need to be discretized more finely.

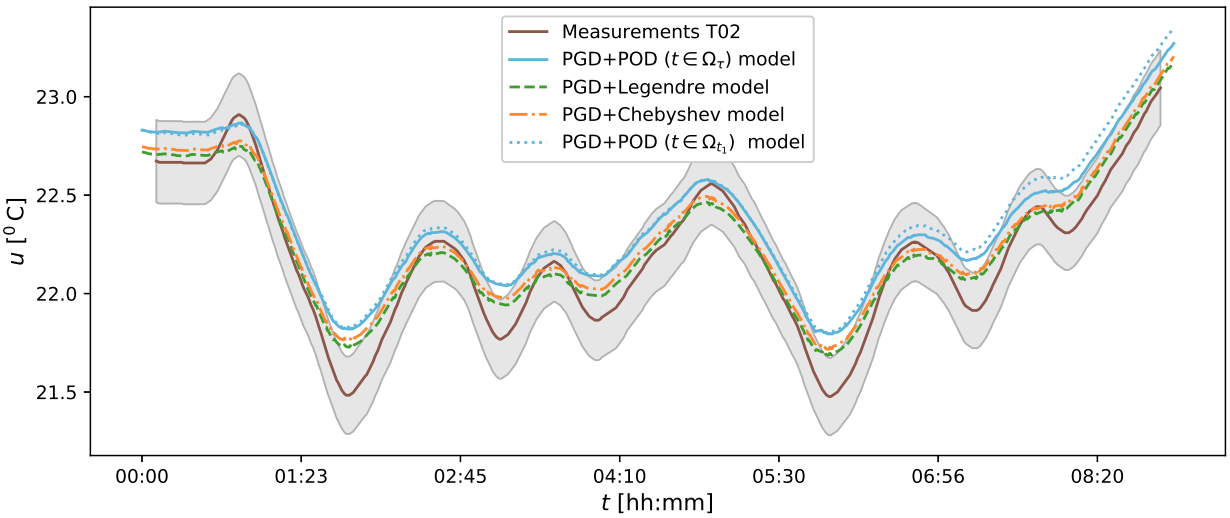
## 5.3 Comparison with experimental data

Finally, the ability of the PGD parametric model to reproduce the dynamics on a realistic example is here studied. The results of the four models for $\mathcal{N} = 3$ and $\Delta\bar{\zeta} = 10^{-5}$ are compared to the measurements.

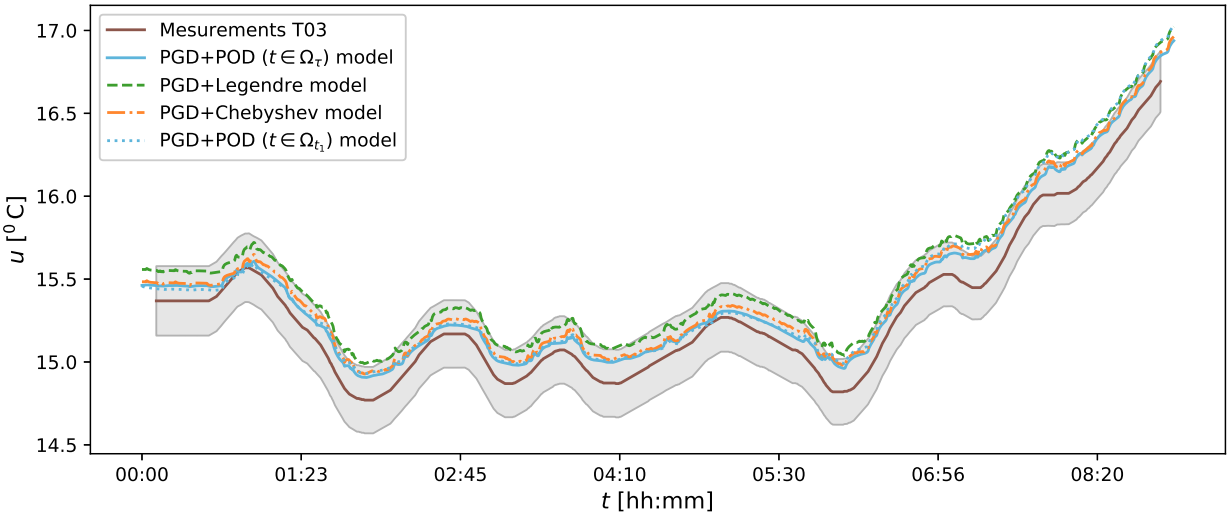
Figure 20 presents the time evolution at the position of sensors T02 and T03, respectively at 4 cm and 8 cm from the inner surface. All four models follow the dynamics of the measured curve. In figure 20(a), we can observe that the unfavorable POD basis matches the favorable POD basis for the first cycles, then the two curves depart from each other. It denotes the fact that the POD basis will be accurate as it encounters its training boundary conditions but will deviate as it encounters different boundary conditions.

Figure 21 presents the error to the measurement data at both depth 4 cm and 8 cm from the inner boundary condition for the various numbers of modes $\mathcal{N}$. The same tendencies are observed as the ones described for figure 19. The error decreases and stabilizes after a few modes for each model. Depending on the reference data, 4 cm and 8 cm, the method ranking is not the same. Results for 4 cm are similar to the one observed comparing the PGD solution to the reference solution (finite difference model). In the results for 8 cm, we can see that the training period of the POD basis has less influence. Indeed at this location, the signal amplitude is eased. It fluctuates less. It could be easier to parameterize this part of the temperature profile.

This last study confirms the ability of the PGD parametric model with the approximation basis to reproduce the dynamics of the signal. At 4 cm, for $\mathcal{N} > 2$, all four models reach the mean experimental

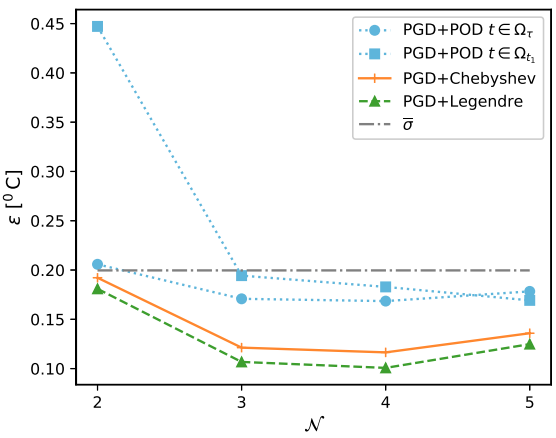



(a) Temperature at 4cm from the inner boundary condition

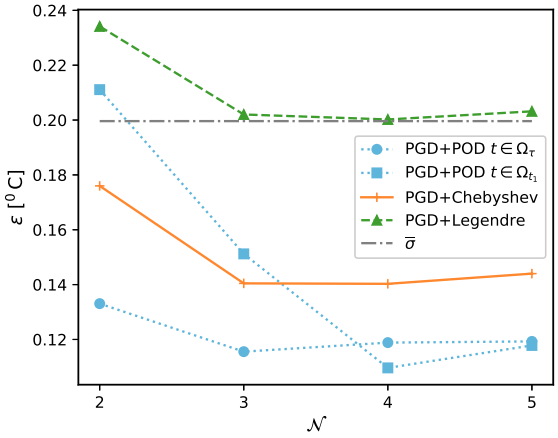



(b) Temperature at 8cm from the inner boundary condition

Figure 20: Time evolution of the temperature measured and calculated by the models at various depths. The grey zone corresponds to $\pm\sigma$, the global experimental uncertainty.




(a) For $\Delta\overline{\zeta} = 10^{-5}$, evolution of the $\varepsilon$ error at 4cm from the inner boundary condition




(b) For $\Delta\overline{\zeta} = 10^{-5}$, evolution of the $\varepsilon$ error at 8cm from the inner boundary condition

Figure 21: PGD parametric model $\varepsilon$ error to the measurement data ($\overline{\sigma}$ correspond to the mean experimental uncertainty)

uncertainty. At 8 cm, for $\mathcal{N} > 2$, the LEGENDRE combined models is getting closer to $\overline{\sigma}$, while the other models errors are under the threshold of the mean experimental uncertainty.

Finally, the accuracy of the models is of the same order of magnitude than the reference solution. Indeed, if we quantify the error between the reference solution and the measurements, we obtain an error of $0.18°C$ at 4 cm from the left boundary condition and $0.13°C$ at 8 cm from the inner boundary condition. Those values are close to the one presented in Figure 21.

# 6 Conclusions

The POD, the CHEBYSHEV and LEGENDRE polynomial approximation basis have been compared first on a theoretical example. This case study was an opportunity to quantify the influence of three main parameters:

1. the number of modes $\mathcal{N}$ in the approximation basis,
2. the discretization coefficient,
3. the number of modes $\mathcal{M}$ in the PGD basis.

The different basis were then compared on a practical example based on measurements. This second case study intended to highlight the influence of the learning process on the accuracy of the POD basis. It also enables the comparison of the three combined PGD parametric models with measurements.

The approximation basis have been first applied to the approximation of the source term. This first step has shown that the discretization should be selected in accordance with the number of modes $\mathcal{N}$. Indeed, increasing the number of modes with an insufficient discretization will not increase the accuracy of the approximation.

The different approximation basis were then integrated into the PGD parametric model. The first study on the influence of the discretization of the approximation coefficient revealed that the accuracy and the computation time are proportional to the discretization. The finer the mesh, the closer the discrete representation to the continuous function. However, as we increase the discretization, we increase the online calculation time.

The study on the influence of the number of modes $\mathcal{N}$ has shown that the error decreases as we increase the number of modes in the approximation basis. This is not the case when the final accuracy is reached with a few modes as it was the case for the POD basis in the theoretical part. Finally, as the number of modes is increased, the computational time increases.

A relation has also been highlighted between the number of approximation modes $\mathcal{N}$ and the convergence rate of the fixed-point algorithm. As the number of modes increases, the number of parameters in the PGD model increases, decreasing the convergence rate of the algorithm. More modes $\mathcal{M}$ are then necessary for the PGD basis to achieve the same accuracy.

The efficiency of the PGD parametric model depends on the three basis on the three previous parameters studied. A compromise should be found between the number of modes $\mathcal{N}$ and $\mathcal{M}$, the discretization and the computation time needed to compute and use the PGD combined model.

The POD approximation basis has the main drawback to require a learning process. The benefit from a PGD parametric model as an *a priori* method is then canceled out by the use of an *a posteriori* method. The combined POD and PGD parametric model becomes then an *a posteriori* model. Its performance depends on the training data-set used.

The influence of this last parameter has been studied in the practical study case. Depending on the data-set used to train the POD basis, it could be the most or the less accurate method to parameterize the source term.

Finally, a compromise should be found to minimize the training period and the computational cost needed to build the basis while keeping an accurate approximation basis. For that purpose, a methodology to select an efficient training period should be developed. This is a point of current work. Some leads have been explored on how to improve the necessary training period by Berger *et al.* (2018, [2]). A methodology has been proposed in [28] to select a short and representative training period for a building wall.

As a final conclusion, we should keep in memory that the POD basis provides an optimum basis if the learning process is complete (the full data-set is used to build the basis). An efficient training data-set is then needed. However, when those data are not available, polynomial basis are a good alternative. They have the main benefit to provide an *a priori* combined PGD parametric model.

However for both methods, the POD or the polynomial approximation, this work should be continued. For the POD method, the learning process remains the main barrier. For polynomial approximation, the parameterization of multi-material wall brings to light new questions. With a multi-layer wall, the source term may not be a smooth function. The efficiency of the polynomial basis to parameterize the temperature profile should then be tested.

Finally, the PGD model combined with each basis as shown its abilities to represent a realistic case study. Those models are ready to be aggregated with other sub-models through a co-simulation process to replace a large original model.

# A Details on the model error due to the inside radiative heat flux

As mentioned in Section 2.1, the net radiative heat flux have been neglected on the inside part of the wall. This heat flux is composed of the short and long-wave radiative heat flux. The short-wave radiative heat flux transmitted through the building windows is generally taken into account and distributed to the building interior surfaces (by solar tracking or with a weighted method) [29]. For the long-wave radiative heat flux, it calculation requires the introduction of non-linear terms, most building simulation tools proposed then simplifications. This heat flux is either neglected, either linearised, and integrated into the convective heat transfer coefficient.

## A.1 Model error for the hypothesis neglecting the inside radiation effects

To evaluate the impact of neglecting the inside net radiative heat flux, it is possible to propose a model error for this hypothesis. To obtain this model, the solution of the heat transfer equation considering inside radiation effects is denoted by $\widetilde{u}$. Then, the boundary condition on the inside part of the wall is:

$$k\,\frac{\partial \widetilde{u}}{\partial x} = -h_{\,\mathrm{in}}\left(\,\widetilde{u} - u_{\,\mathrm{in}}\,\right) + q_{\,\mathrm{in}}\,, \qquad x = L\,,$$

where $q_{\,\mathrm{in}}$ is the incident radiation flux arising from the boundary surfaces facing the studied wall. The error between the solutions is defined by:

$$e \overset{\mathrm{def}}{:=} u - \widetilde{u}\,. \tag{23}$$

Recalling that $u$ is the solution of equation 1, which neglect the inside net radiative heat flux. Since the problem is linear, the model error verifies the following governing equation:

$$c\,\frac{\partial e}{\partial t} = \frac{\partial}{\partial x}\left(\,k\,\frac{\partial e}{\partial x}\,\right), \tag{24}$$

with the following boundary conditions:

$$-k\,\frac{\partial e}{\partial x} = -h_{\,\mathrm{out}}\,e\,, \qquad x = 0\,, \tag{25a}$$

$$k\,\frac{\partial e}{\partial x} = -h_{\,\mathrm{in}}\,e - q_{\,\mathrm{in}}\,, \qquad x = L\,, \tag{25b}$$

and the initial condition:

$$e = 0\,, \quad t = 0\,. \tag{26}$$

The model error equations (24)–(26) can be computed using any of the numerical method presented in Section 2.4 and Section 2.5. This is facilitated by working with dimensionless equations enabling to reuse the same numerical model for different problems.

## A.2 Results for the theoretical case study

The use of the model error is illustrated for the case study defined in Section 4. The inside radiative heat flux is defined through long-wave radiation exchanges with surrounding surfaces:

$$q_{\mathrm{in}} = f_{\mathrm{w}}\,\epsilon_w\,\sigma\,4\left(T^4 - u_{\mathrm{w}}^4\right) + f_{\mathrm{g}}\,\epsilon_g\,\sigma\left(T^4 - u_{\mathrm{g}}^4\right),$$

where $\sigma$ is the BOLTZMANN constant and $\epsilon_{w/g}$ the emissivity of the material. $u_{\mathrm{w}}$ and $u_{\mathrm{g}}$ are the surrounding walls and ground surface temperatures, respectively. The corresponding shape factor are $f_{\mathrm{w}}$ and $f_{\mathrm{g}}$. The first part of the formula corresponds to the radiative balance with the three walls and the ceiling, while the second part corresponds to the balance with the floor.

For the numerical applications, the following values are considered:

$$f_{\mathrm{w}} = f_{\mathrm{g}} = 0.2\,, \epsilon_w = \epsilon_g = 0.9\,, \qquad \sigma = 5.67\cdot 10^{-8}\mathsf{W}\,.\,\mathsf{m}^{-2}\,.\,\mathsf{K}^{-4}\,, u_{\mathrm{w}} = u_{\mathrm{in}}\,, \qquad u_{\mathrm{g}} = 23\ {}^{\circ}\mathsf{C}\,.$$

To obtain the previous numerical values, the followings hypothesis have been made: - the room studied has no windows, - the room is perfectly cubic (all the shape factors are equal to 0.2), - the surface temperatures of the walls and ceiling are equal to the air temperature (an equilibrium has been reached with neighboring rooms), - the floor surface temperature equal to $23{\circ}C$ (underfloor heating).

The flux $q_{\mathrm{in}}$ is computed using *a posteriori* results of the wall. The time variation of the flux is shown in Figure 22(a). It can be remarked that the radiation flux scales between $-25$ and 30 $\mathsf{W}\,.\,\mathsf{m}^{-2}$. It has a very low magnitude compared to the outside flux, illustrated in Figure 5(b). Using the time variation of $q_{\mathrm{in}}$, the model error is computed based on a finite-difference model. The time variation of the model error is given in Figure 22(b). The error reaches a maximum of 1.0 $^{\circ}\mathsf{C}$ located, as expected, on the inside boundary ($x = L$). The impact of the hypothesis neglecting the inside flux can be evaluated on the temperature flux. For this, the solution $\widetilde{u}$ is reconstructed using Eq. (23). The temperature variation are illustrated in Figures 23(a) and 23(b). On the outside surface, the two solutions are almost overlapped. Thus, the influence of the inside radiation is negligible on this part. Indeed, as remarked in Figure 22(b), the model error scales with 0.2 $^{\circ}\mathsf{C}$. On the inside surface, the discrepancy between the solution is higher, around 0.5 $^{\circ}\mathsf{C}$.

As a synthesis, a model error is proposed to evaluate the influence of the hypothesis neglecting the inside net radiative heat flux. It can be computed using any of the numerical models proposed in the manuscript, due to the benefits of working with dimensionless equations. In terms of physical results, the inside radiation effects induce discrepancies on the inside surface of the wall. However, the overall dynamics of heat transfer is not altered. Note that the numerical investigations carried in Sections 4 can be straightforwardly extended to a model considering inside radiation flux.

# B Details on the approximation basis construction

## B.1 Chebyshev polynomials

The CHEBYSHEV polynomials are part of the family of orthogonal polynomials. The first kind CHEBYSHEV polynomial denoted $T_n$ are the following ones:

$$T_0(x) = 1,\ T_1(x) = x,\ T_2 = \left(2x^2 - 1\right),\ T_3 = \left(4x^3 - 3x\right). \tag{27}$$

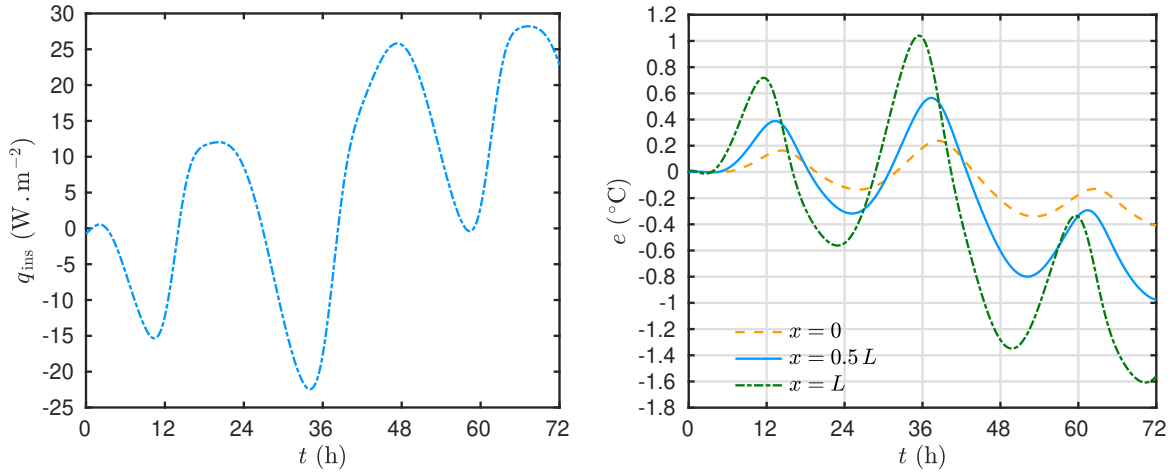



Figure 22: Time evolution of the inside boundary flux due to long-wave radiation *(a)* and of the model error *(b)*

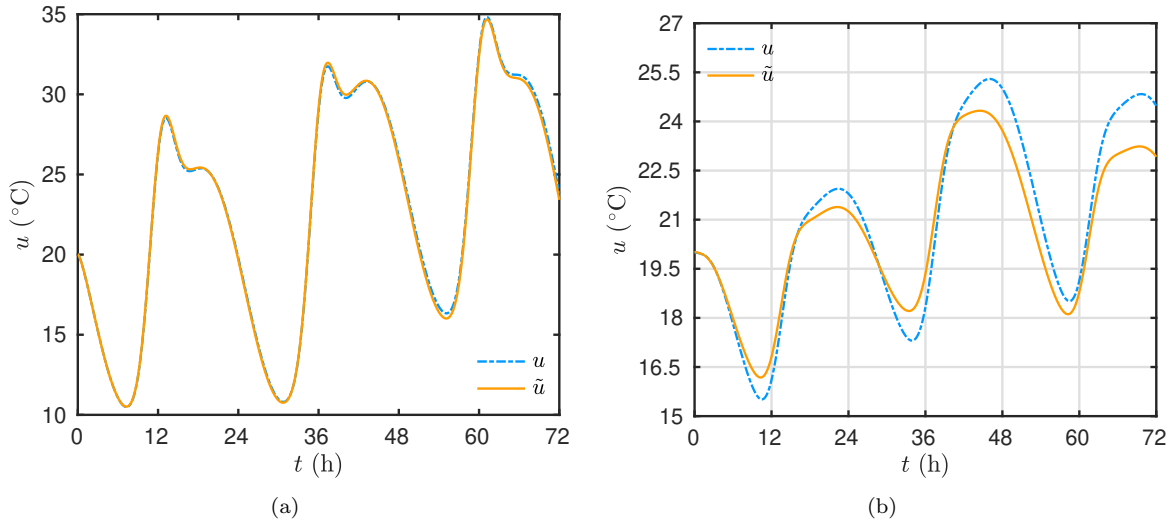



Figure 23: Time evolution of the temperature with or without neglecting the inside radiation flux on the outside *(a)* and inside *(b)* surfaces.

668 They are constructed according to the following relation of recurrence [5, 19]:

$$T_{j+1} = 2\,x\,T_j - T_{j-1} \text{ for } j > 1 \text{ with } T_0 = 1, \text{ and, } T_1 = x \tag{28}$$

669 The Chebyshev approximation basis is made of the Chebyshev polynomials.

$$\Psi_j \equiv T_j \tag{29}$$

670 The Chebyshev polynomials are calculated at the Chebyshev points defined by the equation (30),
671 where $n$ is a positive integer. In the literature several names can be found to describe this set of points as

Chebyshev–Lobatto points, Chebyshev extreme points, or Chebyshev points of the second kind. All those expressions refer to the same set of points according to Trefethen (2013, [5]).

$$x_j \;=\; \cos\left(\frac{j\,\pi}{n}\right)\,,\;\; 0 \;<\; j \;<\; n, \tag{30}$$

Special attention must be given to the spatial domain of the problem. The Chebyshev points define a non-uniform mesh for a space interval $[-1,1]$. Thus, a change of variable must be performed to transform the dimensionless spatial domain $[0,1]$ to $x \in [-1,1]$.

## B.2 Legendre polynomials

The Legendre polynomials are also part of the family of orthogonal polynomials. The first Legendre polynomials are the following ones:

$$P_0\big(x\big) \;=\; 1,\; P_1\big(x\big) \;=\; x,\; P_2 \;=\; \left(\frac{3}{2}\,x^2 \;-\; \frac{1}{2}\right). \tag{31}$$

The next polynomials are constructed according to the following relation of recurrence [5]:

$$(\,j \;+\; 1\,)\,P_{j\,+\,1} \;=\; (\,2\,j \;+\; 1\,)\,x\,P_j \;-\; j\,P_{j\,-\,1} \text{ for } j \;\geqslant\; 1, \text{ and } P_0\big(x\big) \;=\; 1\;,\; P_1\big(x\big) \;=\; x \tag{32}$$

The Legendre approximation basis is made of the Legendre polynomials calculated at the Legendre points.

$$\Psi_j \;\equiv\; P_j \tag{33}$$

As for Chebyshev, special attention must be given to the spatial domain. The spatial mesh will not be uniform and a change of variable must be performed to transform the dimensionless spatial domain from $[0,1]$ to $x \in [-1,1]$.

## B.3 POD reduced basis

The POD method extracts the relevant information from a set of *snapshots* by means of its projection onto a smaller subspace. As a result, from a data-set, the POD builds a deterministic representation, from the basis $\Phi$. The ultimate goal is to retain a detailed representation of the data-set with a minimum or optimal number of modes in $\Phi$. For these properties, the POD method could be used to parameterize the temperature profile (source term in our problem).

$$\Psi_j \;\equiv\; \Phi_j \tag{34}$$

To build the POD basis, a learning process is needed. It has an impact on the accuracy of the reduced-order basis. For this reason, the data-set used must be representative of the problem (boundary values, initial conditions, materials used). More details on the POD methods can be found in [30, 31].

Contrary to the two previous basis, no special attention needs to be paid to the definition of the spatial domain. To standardize the spatial domain used, the same change of variable is performed ($x \in [-1,1]$) and the spatial mesh is set uniform.

# References


[1] Laurent Malys, Marjorie Musy, and Christian Inard. Microclimate and building energy consumption: Study of different coupling methods. *Advances in Building Energy Research*, 9(2):151–174, 2015.

[2] Julien Berger, Walter Mazuroski, Ricardo CLF Oliveira, and Nathan Mendes. Intelligent co-simulation: neural network vs. proper orthogonal decomposition applied to a 2d diffusive problem. *Journal of Building Performance Simulation*, pages 1–20, 2018.

[3] Marie-Hélène Azam, Sihem Guernouti, Marjorie Musy, Julien Berger, Philippe Poullain, and Auline Rodler. A mixed pod–pgd approach to parametric thermal impervious soil modeling: Application to canyon streets. *Sustainable Cities and Society*, 42:444–461, 2018.

[4] Julien Berger and Nathan Mendes. An innovative method for the design of high energy performance building envelopes. *Applied Energy*, 190:266–277, 2017.

[5] Lloyd N Trefethen. *Approximation theory and approximation practice*, volume 128. Siam, 2013.

[6] Suelen Gasparin, Julien Berger, Denys Dutykh, and Nathan Mendes. Solving nonlinear diffusive problems in buildings by means of a spectral reduced-order model. *Journal of Building Performance Simulation*, pages 1–20, 2018.

[7] Suelen Gasparin, Denys Dutykh, and Nathan Mendes. A spectral method for solving heat and moisture transfer through consolidated porous media. *International Journal for Numerical Methods in Engineering*.

[8] Drury B Crawley, Linda K Lawrie, Frederick C Winkelmann, Walter F Buhl, Y Joe Huang, Curtis O Pedersen, Richard K Strand, Richard J Liesen, Daniel E Fisher, Michael J Witte, et al. Energyplus: creating a new-generation building energy simulation program. *Energy and buildings*, 33(4):319–331, 2001.

[9] Suelen Gasparin, Julien Berger, Denys Dutykh, and Nathan Mendes. An adaptive simulation of nonlinear heat and moisture transfer as a boundary value problem. *International Journal of Thermal Sciences*, 133:120–139, 2018.

[10] Lawrence F Shampine, Jacek Kierzenka, and Mark W Reichelt. Solving boundary value problems for ordinary differential equations in matlab with bvp4c. *Tutorial notes*, 2000:1–27, 2000.

[11] Francisco Chinesta, Roland Keunings, and Adrien Leygue. *The proper generalized decomposition for advanced numerical simulations: a primer*. Springer Science & Business Media, 2013.

[12] David González, F Masson, F Poulhaon, Adrien Leygue, Elías Cueto, and Francisco Chinesta. Proper generalized decomposition based dynamic data driven inverse identification. *Mathematics and Computers in Simulation*, 82(9):1677–1695, 2012.

[13] David González, Elías Cueto, and Francisco Chinesta. Real-time direct integration of reduced solid dynamics equations. *International Journal for Numerical Methods in Engineering*, 99(9):633–653, 2014.

[14] Victor Zucatti, Hugo FS Lui, Diogo B Pitz, and William R Wolf. Assessment of reduced-order modeling strategies for convective heat transfer. *Numerical Heat Transfer, Part A: Applications*, 77(7):702–729, 2020.

[15] Elías Cueto, David González, and Icíar Alfaro. *Proper Generalized Decompositions*. SpringerBriefs in Applied Sciences and Technology. Springer International Publishing, Cham, 2016. DOI: 10.1007/978-3-319-29994-5.

[16] Fabien Poulhaon, Francisco Chinesta, and Adrien Leygue. A first step toward a pgd-based time parallelisation strategy. *European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique*, 21(3-6):300–311, 2012.

[17] Domenico Borzacchiello, José V Aguado, and Francisco Chinesta. Non-intrusive sparse subspace learning for parametrized problems. *Archives of Computational Methods in Engineering*, pages 1–24, 2017.

[18] John P Boyd. *Chebyshev and Fourier spectral methods*. Courier Corporation, 2001.

[19] Roger Peyret. *Spectral methods for incompressible viscous flow*, volume 148. Springer Science & Business Media, 2013.

[20] Suelen Gasparin, Julien Berger, Denys Dutykh, and Nathan Mendes. Advanced reduced-order models for moisture diffusion in porous media. *Transport in Porous Media*, 124(3):965–994, 2018.

[21] Francisco Chinesta, Pierre Ladeveze, and Elías Cueto. A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18(4):395, 2011.

[22] Amine Ammar, Francisco Chinesta, Pedro Diez, and Antonio Huerta. An error estimator for separated representations of highly multidimensional models. *Computer Methods in Applied Mechanics and Engineering*, 199(25-28):1872–1880, 2010.

[23] Angel Leon, Anais Barasinski, Emmanuelle Abisset-Chavanne, Elias Cueto, and Francisco Chinesta. Wavelet-based multiscale proper generalized decomposition. *Comptes Rendus Mécanique*, 346(7):485–500, 2018.

[24] Tobin A Driscoll, Nicholas Hale, and Lloyd N Trefethen. Chebfun guide, 2014.

[25] Etienne Pruliere, Francisco Chinesta, and Amine Ammar. On the deterministic solution of multidimensional parametric models using the proper generalized decomposition. *Mathematics and Computers in Simulation*, 81(4):791–810, 2010.

[26] John R Taylor. Error analysis. *Univ. Science Books, Sausalito, California*, 1997.

[27] Journal officiel de la République Fançaise. Th-b-ce, 2012, fascicule th-u matériaux, 2017. `https://www.rt-batiment.fr/batiments-neufs/reglementation-thermique-2012/textes-de-references.html`.

[28] Marie-Hélène Azam, Sihem Guernouti, Marjorie Musy, and Philippe Poullain. How to perform an efficient learning process for a combined pod and pgd soil urban thermal model? application to canyon streets. In *Proceedings of the 16th IBPSA Conference Rome, Italy*, volume 29, pages 3155–3162, Sept. 2-4, 2019.

[29] Nicolas Lauzet, Auline Rodler, Marjorie Musy, Marie-Hélène Azam, Sihem Guernouti, Dasaraden Mauree, and Thibaut Colinart. How building energy models take the local climate into account in an urban context–a review. *Renewable and Sustainable Energy Reviews*, 116:109390, 2019.

[30] YC Liang, HP Lee, SP Lim, WZ Lin, KH Lee, and CG Wu. Proper orthogonal decomposition and its applications—part i: Theory. *Journal of Sound and vibration*, 252(3):527–544, 2002.

[31] Elías Cueto, Francisco Chinesta, and Antonio Huerta. Model order reduction based on proper orthogonal decomposition. In *Separated Representations and PGD-Based Model Reduction*, pages 1–26. Springer, 2014.