4. ANKARA INTERNATIONAL AEROSPACE CONFERENCE AIAC-2007-014
10-12 September, 2007 - METU, Ankara

A NUMERICAL APPROACH FOR ORDER REDUCTION AND BOUNDARY SEPARATION IN NONLINEAR AND TIME DELAY PDE PROCESSES

Mehmet Önder Efe*
TOBB Economics and Technology University
Ankara, Turkey

ABSTRACT

Nonlinearity and time delays are two of the prime problems in dynamic system models. There is a very extensive research volume on both of these subject areas, however, for processes modeled by Partial Differential Equations (PDEs), dealing with these difficulties is further tedious. This paper considers a general PDE model and order reduction based on Proper Orthogonal Decomposition (POD) with boundary separation. The reduced order model has time delays inherited from the PDE process and it is nonlinear. The reduced order model is shown to be capable of capturing the essential behavior accurately through simulations.

INTRODUCTION

Processes described by PDEs are infinite dimensional due to the spatial continuity. Modeling and control of PDE processes is an interesting research area and the outcomes of which address many physical phenomena, e.g. heat and fluid flows. Understanding the behavior of processes governed by PDEs is a tedious task if the process has nonlinearities with time delays. Order reduction is therefore a practical choice giving concessions from accuracy yet the reduced order model is finite dimensional and mathematically tractable. It is a well known fact that for linear PDEs, there are well established alternatives other than POD, [7], but POD is a widely used method in modeling of more complicated systems. The goal of this paper is to present a POD based modeling effort involved with nonlinearities and time delays.

Proper Orthogonal Decomposition was introduced in the pioneering work of Lumley [11] with the goal of explaining the modal nature of turbulent flows. In [10], Sirovich introduced the method of snapshots for reducing the computational intensity of the original POD algorithm. The POD method is widely accepted as a powerful tool for decomposing the dynamical content of a time varying spatially continuous PDE process into some spatial and temporal components meeting orthogonality properties. The spatial part is a set of basis functions while the temporal part is a set of differential equations. The decomposition is accomplished in the order of dominance, which is a significant property enabling the designer to truncate the expression at a particular mode number. Modeling of flow problems governed by PDEs have therefore enjoyed the POD method in obtaining the finite dimensional models at the cost of giving concessions from the model performance, see for example [1-2,4,6] and the references therein. The amount of uncertainty introduced during model reduction depends heavily on the numerical solver, grid density, number of snapshots, coherence of the solution (decomposability) and so on.

Procedurally, the PDE set is solved for the given initial and boundary conditions. Several samples from the solution set are selected and the POD method with Galerkin projection is applied. As a result of this, a set of autonomous Ordinary Differential Equations (ODEs) is obtained. The solution of the obtained ODEs with the given initial conditions synthesize the temporal part of the solution and the spatial basis functions obtained through the POD method yield the approximate solution of the PDE. Unfortunately, the set of ODEs are specific to the initial and boundary conditions used in the model derivation stage. In other words, one needs to change the ODE model for every different instance of boundary excitation regimes and this is a significant problem for boundary control goal. This paper demonstrates how the aforementioned autonomous set of ODEs is converted into non-autonomous ones and how the external excitations are seen explicitly in the model.

* Associate Professor and Chairman of the Department of Electrical and Electronics Engineering, Email: onderefe@etu.edu.tr

This paper is organized as follows. The POD algorithm specific to the chosen PDE process is described in the second section. In the third section, development of the order reduction and boundary separation is presented. The fourth section discusses the modeling results. The concluding remarks are given at the end of the paper.

PROPER ORTHOGONAL DECOMPOSITION (POD)

Consider the ensemble $U_i(x)$, $i = 1,2,...N_s$, where N_s is the number of elements. Every element of this set corresponds to a snapshot observed from a process, say for example the process with initial and boundary conditions given below

$$\begin{aligned} &u_t(x,t) = c^2 u_{xx}(x,t) - \mu u_{xx}(x,t-\tau_1) - \eta u(x,t)u_x(x,t) - \zeta u(x,t-\tau_2)u_x(x,t-\tau_2) + \sigma u(x,t) + f(x)\gamma_1(t) \\ &u(x,0) = 0 \quad \forall x \in \Omega := [0,1], \quad u(0,t) = \gamma_0(t),\ u(1,t) = 0 \end{aligned} \tag{1}$$

where, c, μ, η, ζ, σ are known constants, $f(x)$ is a spatial gain with $f(0) = f(1) = 0$, $\gamma_0(t)$ and $\gamma_1(t)$ are external excitations affecting the process from the 0-boundary and the domain Ω, respectively. The subscripts $_x$, and $_t$ refer to the partial differentiation with respect to space and time, respectively. Note that the process in (1) corresponds to 1D heat equation when $c > 0$ and the parameters μ, η, ζ, σ are all set to zero. Under these settings, the parameter c corresponds to the thermal diffusivity coefficient. Similarly, if $c, \eta > 0$ and the parameters μ, ζ, σ are all set to zero, the resulting process becomes the 1D Burgers equation. Switching on the term μ introduces the effect of τ_1 sec. delayed value of the linear term u_{xx}, and the term ζ activates the contribution of the τ_2 sec. delayed value of the nonlinear terms uu_x. The variables τ_1 and τ_2 are the parameters determining the delay in linear and nonlinear terms and the parameter σ is responsible for the effect of the current solution u. Clearly, the described PDE process is a generalized version of several physical processes that can be characterized by setting the parameters appropriately. The motivating fact for utilizing this process is its time delay terms on both linear and nonlinear components.

The continuous time process takes place over the physical domain $\Omega := \{x \mid x \in [0,1]\}$ and the solution is obtained on a spatial grid denoted by Ω_d, which describes the coordinates of the elements of every snapshot in the ensemble. The entities described over Ω_d, are $N_x \times 1$ dimensional vectors. Note that in (1), $f(0) = f(1) = 0$ so that the problem description is consistent at the boundaries of Ω, and $\gamma_1(t)$ becomes independent from $\gamma_0(t)$, consequently the external excitations can be selected independently and arbitrarily.

With this problem description, the goal of applying POD is to find an orthonormal basis set letting us to write the solution as

$$u(x,t) = \sum_{i=1}^{R_L} \alpha_i(t)\Phi_i(x) \tag{2}$$

where $\alpha_i(t)$ is the i-th temporal mode, $\Phi_i(x)$ is the i-th spatial function (basis function or the eigenfunction), R_L is the number of independent basis functions that can be synthesized from the given ensemble, or equivalently that spans the space described by the ensemble. It will later be clear that if the basis set $\Phi_i(x)$, $i = 1, 2, \ldots, R_L$ is an orthonormal set, Galerkin projection yields the autonomous set of ODEs directly. The POD procedure for obtaining the basis set can be outlined as follows.

First we calculate the $N_s \times N_s$ dimensional correlation matrix L, the ij-th entry of which is $L_{ij} = \langle U_i, U_j \rangle_{\Omega d}$, where $\langle \cdot, \cdot \rangle_{\Omega d}$ is the inner product operator defined over $\Re^{Nx}$.

Second the eigenvectors (denoted by v_i) and the associated eigenvalues (λ_i) of the matrix L are found. These are sorted in a descending order in terms of the magnitudes of λ_i. Note that every v_i is an $N_s \times 1$ dimensional vector satisfying $\|v_i\|^2 = 1/\lambda_i$, here, for simplicity of the exposition, we assume that the eigenvalues are distinct.

Then we construct the basis set by using

$$\Phi_i = \sum_{j=1}^{N_s} v_{ij} U_j \tag{3}$$

where v_{ij} is the j-th entry of the eigenvector $v_i = (v_{i1}\ v_{i2}\ \ldots\ v_{iNs})^T$, and $i = 1,2,...,R_L$, with $R_L = \text{rank}(L)$. It can be shown that $\langle \Phi_i(x), \Phi_j(x) \rangle_\Omega = \delta_{ij}$ with δ_{ij} being the Kronecker delta function. Notice that the basis functions are admixtures of the snapshots, [5-6].

Finally, the temporal coefficients are computed. Taking the inner product of both sides of (2) with $\Phi_i(x)$, the orthonormality property leads to

$$\begin{aligned}\alpha_i(t_0) &= \langle \Phi_i(x), u(x,t_0)\rangle_\Omega \\ &= \langle \phi_i, U_{t_0}\rangle_{\Omega_d} \\ &\coloneqq \frac{1}{N_s}\sum_{j=1}^{N_x}\phi_i(x_j)U_{t_0}(x_j) \\ &\coloneqq \phi_i(x)\oplus U_{t_0}(x)\end{aligned} \tag{4}$$

where $\phi_i \in \Re^{Nx}$ is a sampled form of the basis function Φ_i defined over Ω. The operator denoted by $\oplus$ computes a real number that is the sum of all elements of a matrix obtained through the elementwise multiplication of the two vectors that $\oplus$ lies in between. Without loss of generality, an element of the ensemble $U_i(x)$, i=1,2,...,N_s may be $U(x,t_0)$. Therefore, in order to generate the temporal gain, $\alpha_k(t)$, of the spatial eigenfunction ϕ_k, one would take the inner product of ϕ_k with the elements of the ensemble as given below,

$$\begin{aligned}\langle U_1, \phi_k\rangle_{\Omega_d} &\approx \alpha_k(t_1) \\ \langle U_2, \phi_k\rangle_{\Omega_d} &\approx \alpha_k(t_2) \\ &\vdots \\ \langle U_{N_s}, \phi_k\rangle_{\Omega_d} &\approx \alpha_k(t_{N_s})\end{aligned} \tag{5}$$

The above computation is important for making a comparison between the quantities obtained from the decomposition (See (5)) and the quantities obtained from the model. Note that the temporal coefficients satisfy orthogonality properties over the discrete set $t \in \{t_1, t_2,...,t_{Ns}\}$ (See (6)).

$$\sum_{i=1}^{N_s}\langle U_i, \phi_k\rangle_{\Omega_d}^2 \approx \sum_{i=1}^{N_s}\alpha_k^2(t_i) = \lambda_k \tag{6}$$

For a more detailed discussion on the POD method, the reader is referred to [1-6,11] and the references therein.

First Fundamental Assumption: The majority of works dealing with POD and model reduction applications presume that the flow is dominated by coherent modes, which means that the flow can be decomposed into distinguishable components in the order of dominance. Because of the dominance of coherent modes, the typical spread of the eigenvalues of the correlation matrix turns out to be logarithmic and the terms decay very rapidly in magnitude. This fact enables us to assume that a reduced order representation, say with M modes ($M<R_L$) can also be written as an equality

$$u(x,t) = \sum_{i=1}^{M}\alpha_i(t)\Phi_i(x) \tag{7}$$

and the reduced order model is derived under the assumption that (7) satisfies the governing PDE in (1), ([3-6,8]). Unsurprisingly, such an assumption results in a model having uncertainties, however, one should keep in mind that the goal is to find a model, which matches the infinite dimensional system in some sense of approximation with typically $M<<R_L<N_s$. To represent how good such an expansion is, a percent energy measure is defined as follows

$$E = \frac{\sum_{i=1}^{M}\lambda_i}{\sum_{i=1}^{R_L}\lambda_i}\times 100\% \tag{8}$$

where the tendency of $E\rightarrow 100\%$ means that the model captures the dynamical information contained in the snapshots well. Conversely, a poor model will be obtained as E gets away from 100%. Clearly, POD lets us reduce the dimensionality of the problem from infinity to R_L, and the first fundamental assumption further enables us to reduce the low dimensional (LD) model order to M. In the next section, we demonstrate how the order reduction is performed and how the boundary conditions are transformed into explicit control terms in the corresponding set of ODEs.

ORDER REDUCTION AND BOUNDARY SEPARATION

In the order reduction phase, we need to obtain the autonomous ODE model first. Towards this goal, if (7) is a solution to the PDE in (1), then it has to satisfy the PDE. Substituting (7) into (1) with the fundamental assumption yields

$$\sum_{i=1}^{M}\dot{\alpha}_i(t)\Phi_i(x)=c^2\sum_{i=1}^{M}\alpha_i(t)\Psi_i(x)-\mu\sum_{i=1}^{M}\alpha_i(t-\tau_1)\Psi_i(x)-\eta\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\Phi_i(x)\Lambda_i(x)$$
$$-\zeta\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t-\tau_2)\alpha_j(t-\tau_2)\Phi_i(x)\Lambda_i(x)+\sigma\sum_{i=1}^{M}\alpha_i(t)\Phi_i(x)+f(x)\gamma_1(t) \tag{9}$$

where $\Psi_i(x) = \partial^2\Psi_i(x)/\partial x^2$. Taking the inner product of both sides with $\Phi_k(x)$ and remembering $\langle\Phi_i(x),\Phi_k(x)\rangle_\Omega=\delta_{ik}$ with δ_{ik} being the Kronecker delta results in

$$\dot{\alpha}_k(t)=c^2\sum_{i=1}^{M}\alpha_i(t)\langle\Psi_i(x),\Phi_k(x)\rangle_\Omega-\mu\sum_{i=1}^{M}\alpha_i(t-\tau_1)\langle\Psi_i(x),\Phi_k(x)\rangle_\Omega-\eta\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\langle\Phi_i(x)\Lambda_i(x),\Phi_k(x)\rangle_\Omega$$
$$-\zeta\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t-\tau_2)\alpha_j(t-\tau_2)\langle\Phi_i(x)\Lambda_i(x),\Phi_k(x)\rangle_\Omega+\sigma\sum_{i=1}^{M}\alpha_i(t)\langle\Phi_i(x),\Phi_k(x)\rangle_\Omega+\langle f(x),\Phi_k(x)\rangle_\Omega\gamma_1(t) \tag{10}$$

Defining ϕ_k, ψ_k, β_k and F as the entities in Ω_d corresponding to the entities Φ_k ,Ψ_k , Λ_k and f respectively in Ω, one could rewrite (10) as

$$\dot{\alpha}_k(t)=c^2\sum_{i=1}^{M}\alpha_i(t)\langle\psi_i,\phi_k\rangle_{\Omega_d}-\mu\sum_{i=1}^{M}\alpha_i(t-\tau_1)\langle\psi_i,\phi_k\rangle_{\Omega_d}-\eta\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\langle\phi_i\otimes\beta_i,\phi_k\rangle_{\Omega_d}$$
$$-\zeta\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t-\tau_2)\alpha_j(t-\tau_2)\langle\phi_i\otimes\beta_i,\phi_k\rangle_{\Omega_d}+\sigma\sum_{i=1}^{M}\alpha_i(t)\langle\phi_i,\phi_k\rangle_{\Omega_d}+\langle F,\phi_k\rangle_{\Omega_d}\gamma_1(t) \tag{11}$$

where $\otimes$ stands for the elementwise multiplication of the entities it lies in between. The equation in (11) can be written explicitly by using $\otimes$ and $\oplus$ operators as

$$\dot{\alpha}_k(t)=c^2\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i\oplus\phi_k\right)-\mu\sum_{i=1}^{M}\alpha_i(t-\tau_1)\left(\psi_i\oplus\phi_k\right)-\eta\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\left(\left(\phi_i\otimes\beta_i\right)\oplus\phi_k\right)$$
$$-\zeta\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t-\tau_2)\alpha_j(t-\tau_2)\left(\left(\phi_i\otimes\beta_i\right)\oplus\phi_k\right)+\sigma\sum_{i=1}^{M}\alpha_i(t)\left(\phi_i\oplus\phi_k\right)+\left(F\oplus\phi_k\right)\gamma_1(t) \tag{12}$$

A. Processing of the linear term $\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i\oplus\phi_k\right)$

Notice that $\oplus$ operator can be applied individually over $\Omega_d^1,\Omega_d^2,...,\Omega_d^n$, which are n nonoverlapping subdomains of Ω_d such that $\Omega_d^1\cup\Omega_d^2\cup...\cup\Omega_d^n=\Omega_d$. This lets us separate the entries corresponding to boundaries without modifying the value of $\langle\phi_k,\psi_i\rangle_{\Omega_d}$, i.e. $\phi_k\oplus\psi_i$ as seen in (13),

$$\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i\oplus\phi_k\right)=\frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\psi_i^{\mathrm{T}}\phi_k$$
$$=\frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i(1)\phi_k(1)+\psi_i^{\circ\mathrm{T}}\phi_k^{\circ}\right) \tag{13}$$

In above, ϕ_k° denotes a vector which is obtained when the 0-boundary element of ϕ_k is removed. In the computation of terms like $\phi_k\oplus\psi_i$, the term $\phi_k(1)$ and $\psi_i(1)$ correspond to the first elements of the vectors ϕ_k and ψ_i, respectively.

The k-th component of the first summation in (13), which is obtained when $i = k$, can be separated from the expression and we obtain (14), which lets us embed the boundary conditions into the expression,

$$\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i\oplus\phi_k\right)=\frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i(1)\phi_k(1)+\psi_i^{\circ\mathrm{T}}\phi_k^{\circ}\right)$$
$$=\frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\psi_i(1)\phi_k(1)+\frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\psi_i^{\circ\mathrm{T}}\phi_k^{\circ}$$
$$=\frac{1}{N_s}\alpha_k(t)\psi_k(1)\phi_k(1)+\frac{1}{N_s}\sum_{i=1}^{M}(1-\delta_{ik})\alpha_i(t)\psi_i(1)\phi_k(1)+\frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\psi_i^{\circ\mathrm{T}}\phi_k^{\circ} \tag{14}$$

At this stage of the modeling, we need to paraphrase the 0-boundary condition in such a way that the final expression above can be incorporated with this expression. The underlying idea is straightforward: If (7) is a solution, then is must be satisfied at the boundaries as well, i.e.

$$\sum_{i=1}^{M} \alpha_i(t)\phi_i(1) = \gamma_0(t) \tag{15}$$

which can be paraphrased as

$$\alpha_k(t)\phi_k(1) = \gamma_0(t) - \sum_{i=1}^{M}(1-\delta_{ik})\alpha_i(t)\phi_i(1) \tag{16}$$

or equivalently

$$\frac{1}{N_s}\alpha_k(t)\phi_k(1)\psi_k(1) = \frac{1}{N_s}\gamma_0(t)\psi_k(1) - \frac{1}{N_s}\sum_{i=1}^{M}(1-\delta_{ik})\alpha_i(t)\phi_i(1)\psi_k(1) \tag{17}$$

Substituting (17) into the last line of (14) yields

$$\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i \oplus \phi_k\right) = \frac{1}{N_s}\gamma_0(t)\psi_k(1) + \frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\psi_i^{\circ\mathrm{T}}\phi_k^{\circ} + \frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i(1)\phi_k(1) - \phi_i(1)\psi_k(1)\right) \tag{18}$$

Arranging the terms lets us have the following term

$$\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i \oplus \phi_k\right) = \frac{1}{N_s}\gamma_0(t)\psi_k(1) + \frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\left(\psi_i^{\mathrm{T}}\phi_k - \phi_i(1)\psi_k(1)\right) \tag{19}$$

B. Processing of the linear and delay term $\sum_{i=1}^{M}\alpha_i(t-\tau_1)\left(\psi_i \oplus \phi_k\right)$

The difference in this term is the time delay in temporal part of (19), therefore the corresponding term would be as follows:

$$\sum_{i=1}^{M}\alpha_i(t-\tau_1)\left(\psi_i \oplus \phi_k\right) = \frac{1}{N_s}\gamma_0(t-\tau_1)\psi_k(1) + \frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t-\tau_1)\left(\psi_i^{\mathrm{T}}\phi_k - \phi_i(1)\psi_k(1)\right) \tag{20}$$

C. Processing of the nonlinear term $\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\left(\left(\phi_i \otimes \beta_j\right)\oplus \phi_k\right)$

Referring to (12), the first nonlinear term seen in the expression can be rewritten as follows

$$\begin{aligned}\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\left(\left(\phi_i \otimes \beta_j\right)\oplus \phi_k\right) &= \frac{1}{N_s}\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\mathrm{diag}\left(\phi_i\beta_j^{\mathrm{T}}\right)\phi_k \\ &= \frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\phi_i(1)\sum_{j=1}^{M}\alpha_j(t)\beta_j(1)\phi_k(1) + \frac{1}{N_s}\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\mathrm{diag}\left(\phi_i^{\circ}\beta_j^{\circ\mathrm{T}}\right)\phi_k^{\circ} \\ &= \frac{1}{N_s}\gamma_0(t)\sum_{j=1}^{M}\alpha_j(t)\beta_j(1)\phi_k(1) + \frac{1}{N_s}\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\mathrm{diag}\left(\phi_i^{\circ}\beta_j^{\circ\mathrm{T}}\right)\phi_k^{\circ}\end{aligned} \tag{21}$$

D. Processing of the nonlinear and delay term $\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t-\tau_2)\alpha_j(t-\tau_2)\left(\left(\phi_i \otimes \beta_j\right)\oplus \phi_k\right)$

Similar to the delay in the linear terms, the delay in nonlinear terms can be expressed as follows. This is due to (21).

$$\begin{aligned}\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t-\tau_2)\alpha_j(t-\tau_2)\left(\left(\phi_i \otimes \beta_j\right)\oplus \phi_k\right) &= \frac{1}{N_s}\gamma_0(t-\tau_2)\sum_{j=1}^{M}\alpha_j(t-\tau_2)\beta_j(1)\phi_k(1) + \\ &\frac{1}{N_s}\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t-\tau_2)\alpha_j(t-\tau_2)\mathrm{diag}\left(\phi_i^{\circ}\beta_j^{\circ\mathrm{T}}\right)\phi_k^{\circ}\end{aligned} \tag{22}$$

E. Processing of the term $\sum_{i=1}^{M}\alpha_i(t)(\phi_i \oplus \phi_k)$

Among other terms, the contribution of the term $\sum_{i=1}^{M}\alpha_i(t)(\phi_i \oplus \phi_k)$ is simplest due to the orthonormality of the basis functions and the result is given in (23).

$$\begin{aligned}\sum_{i=1}^{M}\alpha_i(t)(\phi_i \oplus \phi_k) &= \frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\phi_i^{\mathrm{T}}\phi_k \\ &= \frac{1}{N_s}\sum_{i=1}^{M}\alpha_i(t)\delta_{ik}\end{aligned} \tag{23}$$

F. Processing of the domain excitation term $(F \oplus \phi_k)\gamma_1(t)$

The excitation $\gamma_1(t)$ is distributed over Ω_d by the spatial gain F. The contribution of this term is $\frac{1}{N_s}F^{\mathrm{T}}\phi_k\gamma_1(t)$. Concatenating the terms computed so far yields the model given in (24).

$$\begin{aligned}\dot{\alpha}_k(t) = {} & \gamma_0(t)\frac{c^2\psi_k(1)}{N_s} + \sum_{i=1}^{M}\alpha_i(t)\frac{c^2\left(\psi_i^{\mathrm{T}}\phi_k - \phi_i(1)\psi_k(1)\right)}{N_s} - \gamma_0(t-\tau_1)\frac{\mu\psi_k(1)}{N_s} \\ & - \sum_{i=1}^{M}\alpha_i(t-\tau_1)\frac{\mu\left(\psi_i^{\mathrm{T}}\phi_k - \phi_i(1)\psi_k(1)\right)}{N_s} - \gamma_0(t)\sum_{j=1}^{M}\alpha_j(t)\frac{\eta\left(\beta_j(1)\phi_k(1)\right)}{N_s} - \sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t)\alpha_j(t)\frac{\eta\,\mathrm{diag}\left(\phi_i^{\circ}\beta_j^{\circ\mathrm{T}}\right)\phi_k^{\circ}}{N_s} \\ & - \gamma_0(t-\tau_2)\sum_{j=1}^{M}\alpha_j(t-\tau_2)\frac{\zeta\left(\beta_j(1)\phi_k(1)\right)}{N_s} - \sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i(t-\tau_2)\alpha_j(t-\tau_2)\frac{\zeta\,\mathrm{diag}\left(\phi_i^{\circ}\beta_j^{\circ\mathrm{T}}\right)\phi_k^{\circ}}{N_s} + \\ & \sum_{i=1}^{M}\alpha_i(t)\frac{\sigma\delta_{ik}}{N_s} + \frac{\left(F^{\mathrm{T}}\phi_k\right)}{N_s}\gamma_1(t) \qquad k = 1,2,\cdots,M\end{aligned} \tag{24}$$

Defining the state vector $\alpha(t) = \begin{pmatrix}\alpha_1(t) & \alpha_2(t) & \cdots & \alpha_M(t)\end{pmatrix}^{\mathrm{T}}$, one can compactly write the reduced order dynamics as in (25);

$$\begin{aligned}\dot{\alpha}(t) = {} & \left(c^2A + \sigma I_{M\times M}\right)\alpha(t) - \mu A\alpha(t-\tau_1) - \eta B(\alpha(t)) - \zeta B(\alpha(t-\tau_2)) + \left(c^2C - \eta D\alpha(t)\right)\gamma_0(t) - \mu C\gamma_0(t-\tau_1) \\ & - \zeta D\alpha(t-\tau_2)\gamma_1(t-\tau_2) + E\gamma_1(t)\end{aligned} \tag{25}$$

where the computation of the terms are as follows

$$\begin{aligned}A_{ki} &= \frac{1}{N_s}\left(\psi_i^{\mathrm{T}}\phi_k - \phi_i(1)\psi_k(1)\right) \\ B(\alpha(t)) &= \begin{pmatrix}\alpha(t)^{\mathrm{T}}B_1\alpha(t) & \alpha(t)^{\mathrm{T}}B_2\alpha(t) & \cdots & \alpha(t)^{\mathrm{T}}B_M\alpha(t)\end{pmatrix}^{\mathrm{T}} \\ (B_k)_{ij} &= \frac{1}{N_s}\mathrm{diag}\left(\phi_i^{\circ}\beta_j^{\circ\mathrm{T}}\right)\phi_k^{\circ} \\ C &= \frac{1}{N_s}\begin{pmatrix}\psi_1(1) & \psi_2(1) & \cdots & \psi_M(1)\end{pmatrix}^{\mathrm{T}} \\ D_{kj} &= \frac{1}{N_s}\beta_j(1)\phi_k(1) \\ E &= \frac{1}{N_s}\begin{pmatrix}F^{\mathrm{T}}\phi_1 & F^{\mathrm{T}}\phi_2 & \cdots & F^{\mathrm{T}}\phi_M\end{pmatrix}^{\mathrm{T}}\end{aligned} \tag{26}$$

This result practically lets us have a representative nonlinear dynamical model for the infinite dimensional process in (1), which is excited through the 0-boundary and the domain. The next section presents to what extent the modeling strategy discussed here could be successful.

SIMULATIONS FOR JUSTIFICATION OF THE DYNAMIC MODEL

The first issue in the order reduction procedure is to obtain several exemplar snapshots. Due to the numerical advantages, the PDE has been solved by using Crank-Nicholson method, [9]. For this purpose, we discretize the PDE in (1) as follows: The time derivative is approximated as in (27),

$$u_t \approx \frac{1}{\Delta t}\left(u_x^{t+\Delta t} - u_x^t\right) \tag{27}$$

where we set Δt = 1 msec. and the second derivative is approximated as

$$u_{xx} \approx \frac{\lambda}{(\Delta x)^2}\left(u_{x+\Delta x}^{t+\Delta t} - 2u_x^{t+\Delta t} + u_{x-\Delta x}^{t+\Delta t}\right) + \frac{1-\lambda}{(\Delta x)^2}\left(u_{x+\Delta x}^{t} - 2u_x^{t} + u_{x-\Delta x}^{t}\right) \tag{28}$$

where we set $\lambda = 0.5$ and $\Delta x = 1/(N_x - 1)$and the second derivative is approximated as

$$\begin{aligned}\frac{1}{\Delta t}\left(u_x^{t+\Delta t} - u_x^t\right) = {} & \frac{c^2\lambda}{(\Delta x)^2}\left(u_{x+\Delta x}^{t+\Delta t} - 2u_x^{t+\Delta t} + u_{x-\Delta x}^{t+\Delta t}\right) + \frac{c^2(1-\lambda)}{(\Delta x)^2}\left(u_{x+\Delta x}^{t} - 2u_x^{t} + u_{x-\Delta x}^{t}\right) \\ & - \frac{\mu\lambda}{(\Delta x)^2}\left(u_{x+\Delta x}^{t+\Delta t-N\Delta t} - 2u_x^{t+\Delta t-N\Delta t} + u_{x-\Delta x}^{t+\Delta t-N\Delta t}\right) - \frac{\mu(1-\lambda)}{(\Delta x)^2}\left(u_{x+\Delta x}^{t-N\Delta t} - 2u_x^{t-N\Delta t} + u_{x-\Delta x}^{t-N\Delta t}\right) \\ & + \sigma u_x^t - \eta u_x^t \frac{u_{x+\Delta x}^t - u_x^t}{\Delta x} - \zeta u_x^{t-P\Delta t} \frac{u_{x+\Delta x}^{t-P\Delta t} - u_x^{t-P\Delta t}}{\Delta x} + f_x\gamma^t\end{aligned} \tag{29}$$

$$x = i\Delta x, \quad i = 2,3,\cdots,N_x - 1, \qquad t = k\Delta t, \quad k = 0,1,\cdots,T/\Delta t$$

Second Fundamental Assumption: A snapshot is obtained from the spatial and temporal running of the expression in (29). If the grid is dense enough then the uncertainty on a snapshot gets tolerable. The assumption here is the use of equality in (29), which states that the performed discretization is accurate enough to collect snapshots.

Corollary: A natural conclusion of the first and the second fundamental assumptions is the order reducibility and boundary separability for a PDE process.

In the modeling stage, the initial values are taken zero everywhere and we have $c = 2$, $\mu = -1$, $\eta = 0.2$, $\zeta = -0.2$, $\sigma = 1$. In order to form the solution, a linear grid having N_x=100 points is chosen. According to the above parameter values, a set of 501 snapshots embodies the entire numerical solution, among which a linearly sampled $N = 251$ snapshots have been used for the POD scheme. Although one may use the entire set of snapshots, it has been shown by Sirovich that a reasonably descriptive subset of them can be used for the same purpose, [10]. In the literature, this approach is called *method of snapshots*, which significantly reduces the computational intensity of the overall scheme, [6,8]. Once the modes have been obtained, we truncate the solution at $M = 6$, which represents 99.9999957739675% of the total energy described in the denominator of the expression in (8). It is visible that if the snapshots are descriptive enough, a LD model with the first six modes captures the information contained implicitly within the snapshots almost perfectly. In order to demonstrate the performance of the dynamic model, we choose the function which is effective over Ω as $f(x)=\sin(4\pi x)$. As the temporal excitations for modeling we chose the following excitation signals,

$$\gamma_0(t) = \sin\left(2\pi 50t(T-t)\right) \tag{30}$$

$$\gamma_1(t) = 50\sin\left(10\pi t\right) \tag{31}$$

where $T = 0.5$ seconds. The choice of the above excitations signals is deliberate as they are spectrally rich enough. As can be seen from Figure 1, $\alpha_k(t)$'s will undergo regimes that change sometimes slowly and sometimes fast depending on the spectral composition of the excitations. The POD algorithm computes the eigenvalues of the correlation matrix L as depicted in Figure 2. The sufficiency of choosing the first six modes is apparent from the figure too. The Matlab/Simulink realization of the reduced order model is shown in Figure 3. The response of the LD model is illustrated in Figure 4, in which every subplot on the left contains two curves. Obviously, the temporal variables obtained form the POD algorithm are very close to those obtained from the LD model and this observation indicates that the LD model is a good representative for the chosen modeling conditions. This can also be seen from Figure 5, where the approximate solution obtained from (7) and (25) is shown with the numerical solution obtained from (29). Unsurprisingly, the two responses are very similar. In (32)-(41), the numerical details of the model in (25) are given.

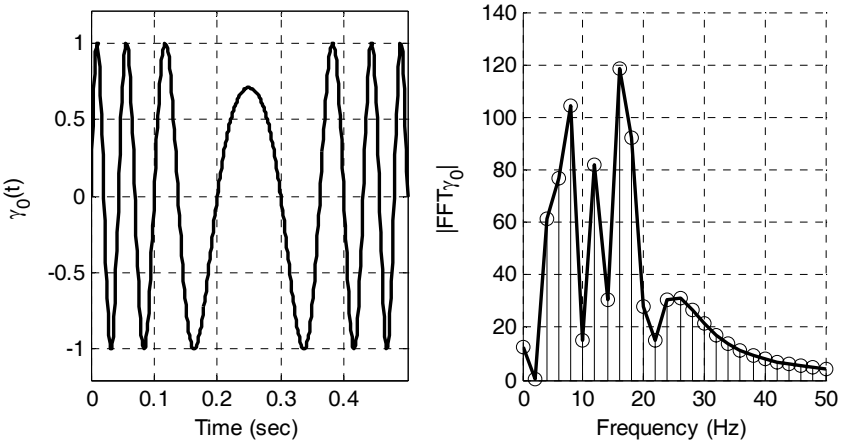

Figure 1. (Left) The excitation $\gamma_0(t)$ and (Right) its spectral view

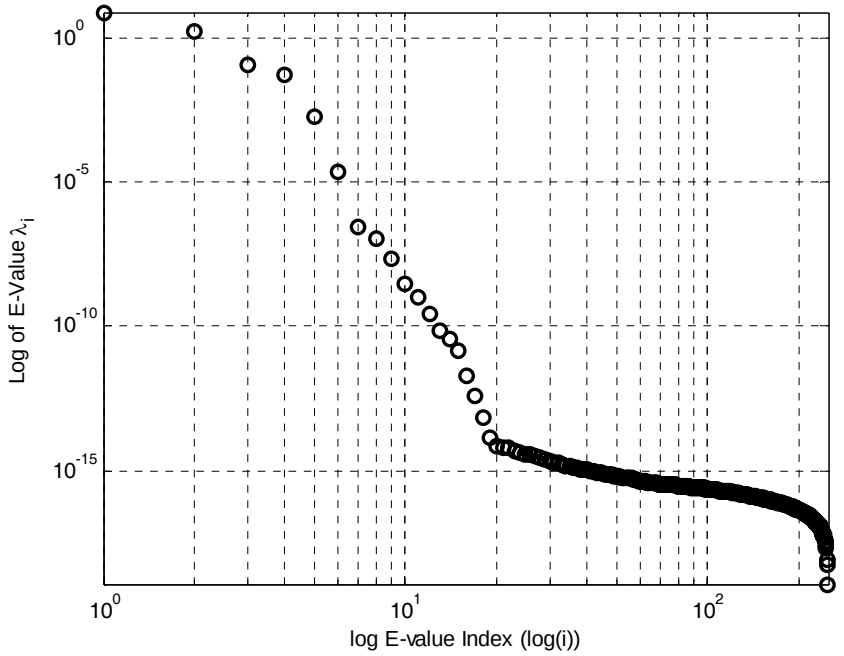

Figure 2. Spread of the eigenvalues of the correlation matrix L

$$A = \begin{pmatrix} -3.8937 & 32.8752 & 35.9159 & 109.7163 & -186.1378 & -328.3902 \\ -10.0541 & -3.8114 & 16.5850 & 57.6965 & -157.1901 & -312.7914 \\ -9.8407 & -5.9212 & -73.5306 & 77.8695 & -61.2901 & -152.5457 \\ -0.5581 & -8.2657 & 63.3593 & -95.9258 & -121.2345 & -320.8153 \\ 25.0177 & 27.5362 & 17.9476 & 37.3025 & -47.1323 & 130.3228 \\ 55.0243 & 60.6417 & 35.9768 & 76.6397 & -84.6801 & -212.6550 \end{pmatrix} \tag{32}$$

$$B_1 = \begin{pmatrix} 5.3355 & 17.6827 & 11.6382 & 20.8407 & -17.7730 & -15.0712 \\ -0.6176 & 6.9689 & 7.4282 & 19.9304 & -20.9198 & -21.6081 \\ -1.1027 & 1.4929 & 1.2345 & 12.7549 & -14.3040 & -9.1131 \\ -1.2567 & 0.2640 & -1.2871 & 10.8927 & -19.1720 & -37.9435 \\ 0.9975 & 2.5790 & 3.4666 & -1.6219 & 9.2748 & 28.1855 \\ -0.2023 & 3.1254 & -2.2817 & 14.4102 & -4.8491 & 12.6354 \end{pmatrix} \tag{33}$$

$$B_2 = \begin{pmatrix} -0.6176 & 6.9689 & 7.4282 & 19.9304 & -20.9198 & -21.6081 \\ 3.4470 & 6.0575 & -1.5387 & 11.6700 & -17.9604 & -24.9351 \\ 1.1654 & 5.9399 & 2.7703 & 3.7539 & -0.4243 & -23.7748 \\ -0.0691 & 4.8431 & 5.1633 & 9.6452 & -14.3005 & -18.4730 \\ 0.7227 & -0.5180 & -9.5642 & -0.5426 & 9.2660 & 16.0355 \\ 1.1304 & 2.7321 & 9.3642 & -5.3636 & 2.6319 & 8.5675 \end{pmatrix} \quad (34)$$

$$B_3 = \begin{pmatrix} -1.1027 & 1.4929 & 1.2345 & 12.7549 & -14.3040 & -9.1131 \\ 1.1654 & 5.9399 & 2.7703 & 3.7539 & -0.4243 & -23.7748 \\ 3.3713 & 0.5968 & 0.9293 & -3.6308 & -10.6210 & -5.4872 \\ 0.2249 & 3.3281 & 5.4885 & 3.2106 & -7.1517 & -10.1129 \\ 0.5174 & -1.2366 & 1.9605 & -8.2096 & 2.7647 & 2.7833 \\ 1.0711 & 2.6398 & -0.6542 & -0.3378 & 6.6114 & 5.2517 \end{pmatrix} \quad (35)$$

$$B_4 = \begin{pmatrix} -1.2567 & 0.2640 & -1.2871 & 10.8927 & -19.1720 & -37.9435 \\ -0.0691 & 4.8431 & 5.1633 & 9.6452 & -14.3005 & -18.4730 \\ 0.2249 & 3.3281 & 5.4885 & 3.2106 & -7.1517 & -10.1129 \\ 2.9425 & 5.8443 & 8.2394 & 9.1895 & -9.3010 & -18.6174 \\ -1.3660 & -7.7399 & 2.4911 & -8.9989 & 8.6451 & 8.0415 \\ 0.9504 & -0.5583 & -3.9711 & -5.0815 & 9.1487 & 9.2594 \end{pmatrix} \quad (36)$$

$$B_5 = \begin{pmatrix} 0.9975 & 2.5790 & 3.4666 & -1.6219 & 9.2748 & 28.1855 \\ 0.7227 & -0.5180 & -9.5642 & -0.5426 & 9.2660 & 16.0355 \\ 0.5174 & -1.2366 & 1.9605 & -8.2096 & 2.7647 & 2.7833 \\ -1.3660 & -7.7399 & 2.4911 & -8.9989 & 8.6451 & 8.0415 \\ 2.4591 & 2.9250 & 7.0272 & 5.9043 & -7.6344 & -9.6191 \\ -0.0403 & 4.9787 & 4.3408 & 9.3279 & -6.9734 & -7.3946 \end{pmatrix} \quad (37)$$

$$B_6 = \begin{pmatrix} -0.2023 & 3.1254 & -2.2817 & 14.4102 & -4.8491 & 12.6354 \\ 1.1304 & 2.7321 & 9.3642 & -5.3636 & 2.6319 & 8.5675 \\ 1.0711 & 2.6398 & -0.6542 & -0.3378 & 6.6114 & 5.2517 \\ 0.9504 & -0.5583 & -3.9711 & -5.0815 & 9.1487 & 9.2594 \\ -0.0403 & 4.9787 & 4.3408 & 9.3279 & -6.9734 & -7.3946 \\ 4.3487 & 10.1925 & 4.5050 & 10.1452 & -9.6047 & -7.9217 \end{pmatrix} \quad (38)$$

$$C = \begin{pmatrix} 0.1689 & -0.3328 & -1.2986 & -1.3813 & 7.2871 & 16.0609 \end{pmatrix}^T \quad (39)$$

$$D = \begin{pmatrix} -0.0298 & -0.4318 & -0.4150 & -1.0536 & 1.8493 & 3.2569 \\ -0.0328 & -0.4748 & -0.4564 & -1.1585 & 2.0334 & 3.5811 \\ -0.0198 & -0.2867 & -0.2756 & -0.6996 & 1.2279 & 2.1624 \\ -0.0405 & -0.5865 & -0.5637 & -1.4309 & 2.5116 & 4.4232 \\ 0.0410 & 0.5945 & 0.5714 & 1.4504 & -2.5458 & -4.4835 \\ 0.0497 & 0.7203 & 0.6923 & 1.7575 & -3.0848 & -5.4327 \end{pmatrix} \quad (40)$$

$$E = \begin{pmatrix} -0.1344 & -0.0882 & -0.2820 & 0.3030 & -0.0038 & -0.0009 \end{pmatrix}^T \quad (41)$$

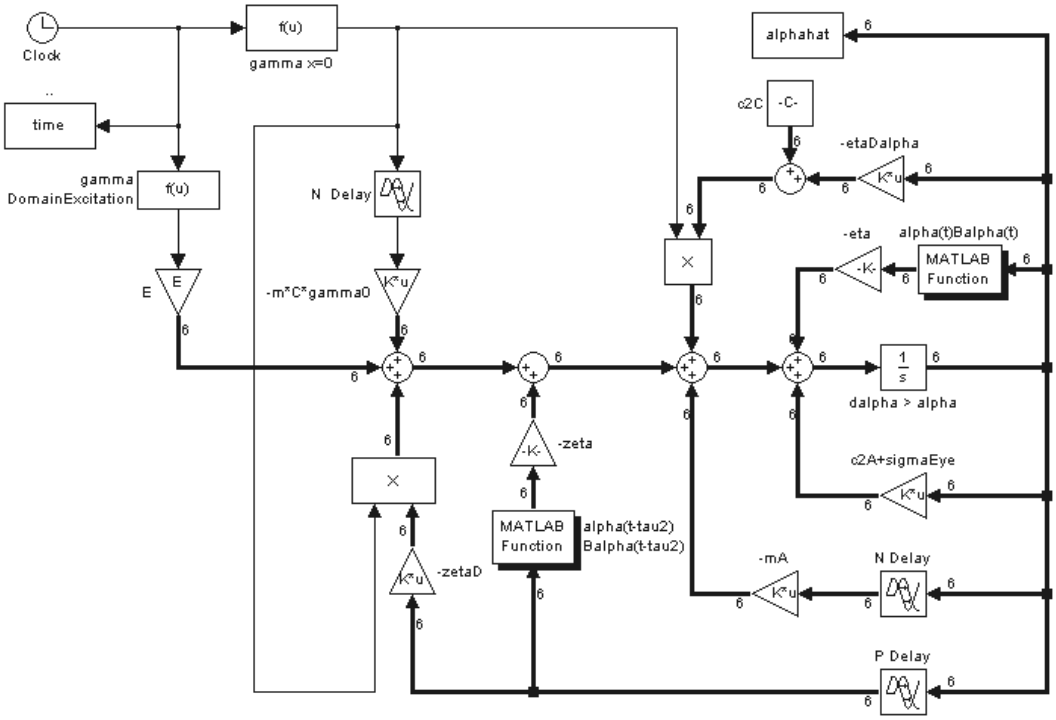

Figure 3. Block diagram of the Matlab realization of the reduced order model in (25)

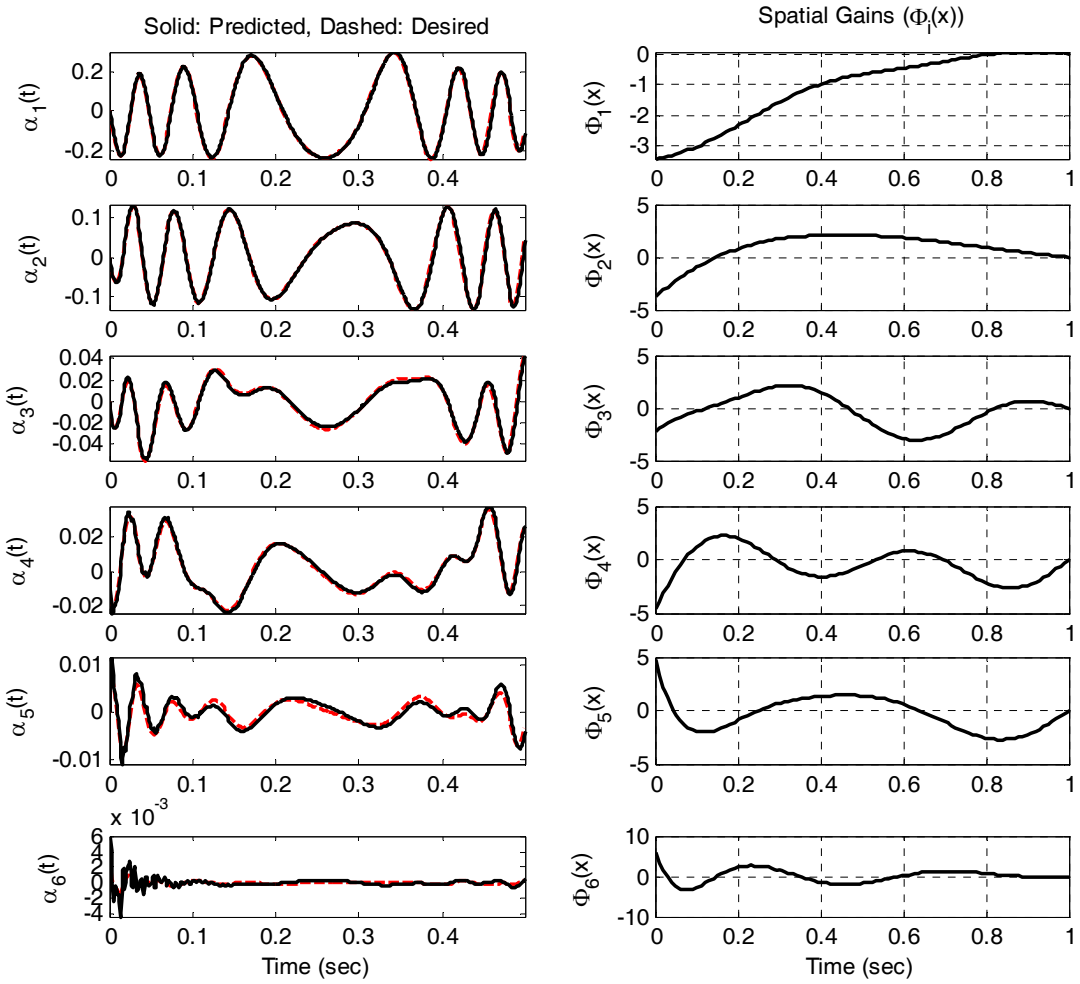

Figure 4. Left: Temporal responses of the reduced order model and the desired values obtained from POD. Right: The orthonormal set of basis functions

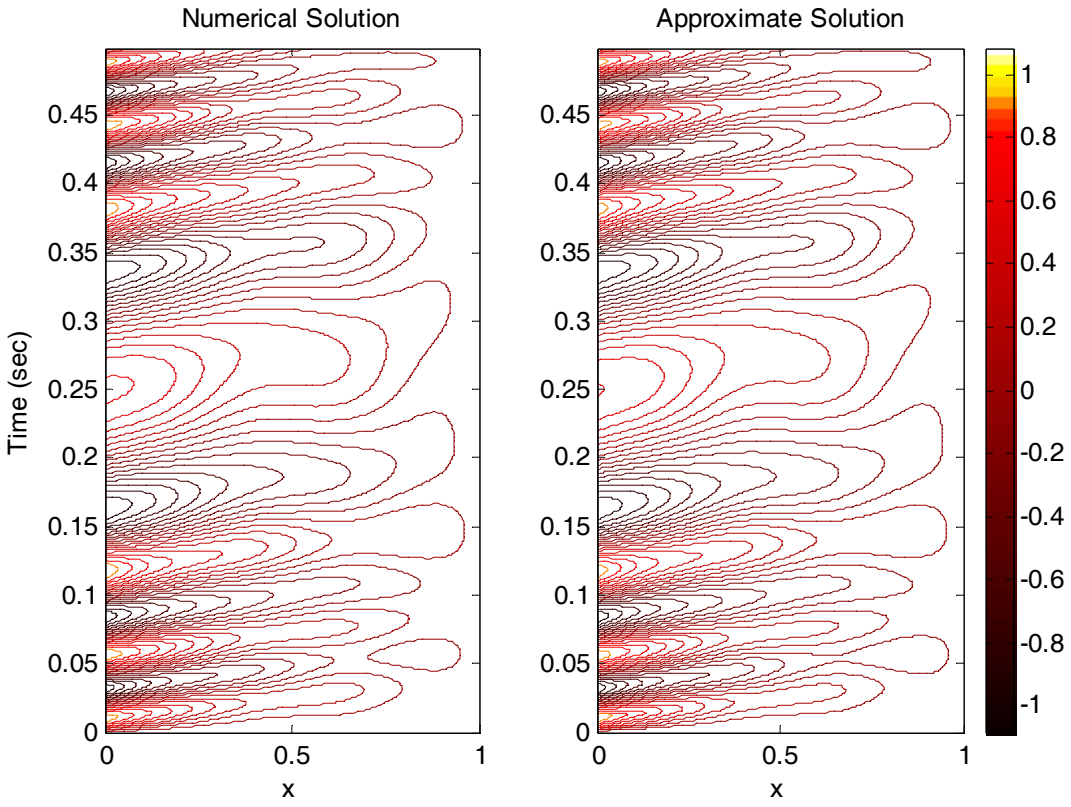

Figure 5. The responses for the modeling conditions in (31) and (42). Left: Numerical solution obtained from (29) and Right: The approximate solution obtained from the LD model

Undoubtedly, one would expect a good match between the state variables obtained from the POD algorithm and the state variables obtained through the numerical solution of the ODE set in (25). One might question whether the model is specific to the boundary conditions above. A practical remedy to this can be accomplished by choosing another set of external inputs and obtaining the response of the model without modifying the model parameters. For this purpose, we change the boundary excitation $\gamma_0(t)$ as follows and leave the domain excitation as it is.

$$\gamma_0(t) = \sin(2\pi 40t^2(1-2t-t^2)) \tag{42}$$

With this change we obtain the results illustrated in Figure 6 and Figure 7. It is seen that the state variables are obtained precisely and a very good match between the spatiotemporal views are observed.

CONCLUSIONS

This paper considers POD based order reduction and boundary separation of a flow described by a PDE. Proper orthogonal decomposition is a widely used technique to express the approximate solution in an ordered fashion. The ordering provided by the algorithm is based on the energy carried by a mode. Therefore, the designer is able to truncate the expansion at an affordable mode number. The higher the number of modes contained, the more complicated the resulting LD model will be. The paper considers a general PDE model that can be used to simulate the Burgers equation, heat equation and similar flows. Furthermore, the paper focuses on the effect of time delays on the reduced order models. Two fundamental assumptions are given. First states that the flow is dominated by coherent modes and a finite term approximation can be written as equality. The second fundamental assumption states that the numerical solution is accurate enough to utilize in the modeling stage. The paper validates the delay based nonlinear dynamic model and emphasizes that the resulting model is useful on a range of operating conditions that display some similarity to the model derivation conditions. The simulation results have shown that the finite dimensional model reconstructs the behavior very precisely.

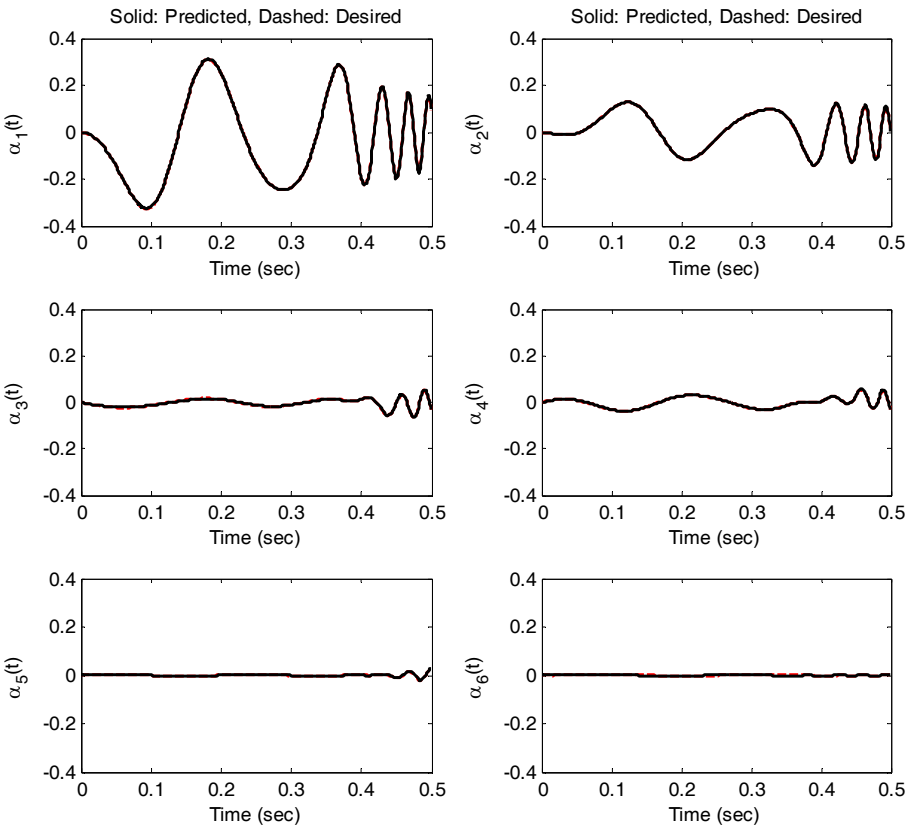

Figure 6. Temporal responses of the reduced order model and the desired values obtained from POD.

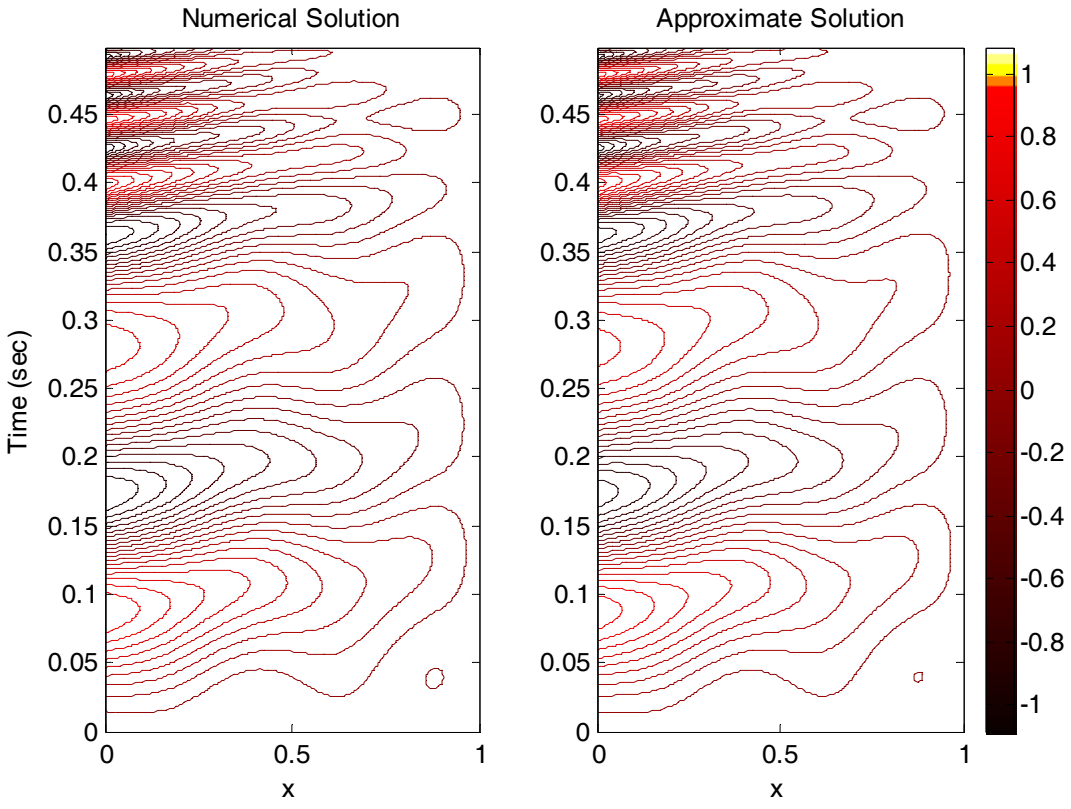

Figure 7. The responses for the test conditions in (31) and (42). Left: Numerical solution obtained from (29) and Right: The approximate solution obtained from the LD model

REFERENCES

[1] Rowley, C.W., 2005: Model reduction for fluids, using balanced proper orthogonal decomposition. International Journal of Bifurcation and Chaos, 15 (3), pp.997-1013.

[2] Rowley, C.W., Colonius, T. and Murray, R.M., 2004: Model reduction for compressible flows using POD and Galerkin projection. Physica D-Nonlinear Phenomena, 189 (1-2), pp.115-129.

[3] Efe, M.Ö. and Özbay, H., 2004: Low dimensional modeling and Dirichlet boundary controller design for Burgers equation. International Journal of Control, 77, 10, pp.895-906.

[4] Caraballo, E., Malone, J., Samimy, M., and DeBonis, J.,2004: A study of subsonic cavity flows: low dimensional modeling. AIAA Paper, AIAA 2004-2124.

[5] Efe, M.Ö. and Özbay, H., 2003: Proper orthogonal decomposition for reduced order modeling: 2D heat flow. Proc. of the IEEE Int. Conf. on Control Applications (CCA'2003), Istanbul, Turkey, June 23-25, pp.1273-1278.

[6] Ly, H.V. and Tran, H.T., 2001: Modeling and control of physical processes using proper orthogonal decomposition. Mathematical and Computer Modelling of Dynamical Systems, 33, pp.223-236.

[7] Gügercin, S. and Antoulas, A.C., 2000: A comparative study of 7 model reduction algorithms. Proc. of the 39th IEEE Conference on Decision and Control, Sydney, Australia.

[8] Ravindran, S.S., 2000: A reduced order approach for optimal control of fluids using proper orthogonal decomposition. Int. Journal for Numerical Methods in Fluids, 34, pp.425-488.

[9] Farlow, S.J., 1993: Partial Differential Equations for Scientists and Engineers. Dover Publications Inc., New York, pp.317-322.

[10] Sirovich, L., 1987: Turbulence and the dynamics of coherent structures. Quarterly of Applied Mathematics, XLV, no. 3, pp. 561-590.

[11] Lumley, J., 1967: The structure of inhomogeneous turbulent flows. Atmospheric Turbulence and Wave Propagation, Nauca, Moscow, pp.166-176.