Identification of Low Order Models for Large Scale Systems ⋆

Satyajit Wattamwar * **Siep Weiland** * **Ton Backx** *

* *All authors are with Dept. Of Electrical Engg. Technical University of Eindhoven, P.O. Box 513, 5600 MB Eindhoven, The Netherlands*
s.wattamwar@tue.nl

Abstract: In this paper we propose a novel procedure for obtaining a low order non-linear model of a large scale multi-phase, non-linear, reactive fluid flow systems. Our approach is based on the combinations of the methods of Proper Orthogonal Decomposition (*POD*), and non-linear System Identification (*SID*) techniques. The problem of non-linear model reduction is formulated as parameter estimation problem. In the first step POD is used to separate the spatial and temporal patterns and in the second step a model structure and it's parameters of linear and of non-linear polynomial type are identified to approximate the temporal patterns obtained by the POD in the first step. The proposed model structure treats POD modal coefficients as states rather than outputs of the identified model. The state space matrices which happens to be the parameters of a black-box to be identified, comes linearly in parameter estimation process. For the same reason, Ordinary Least Square (*OLS*) method is used to estimate the model parameters. The simplicity and reliability of the proposed method gives computationally very efficient linear and non-linear low order models for extremely large scale processes. The method is of generic nature. The efficiency of proposed approach is illustrated on a very large scale benchmark problem depicting Industrial Glass Manufacturing Process (*IGMP*). The results show good performance of the proposed method.

1. INTRODUCTION

Industrial processes involving fluid flows are usually modeled by Navier-Stoke's equations which are solved by some kind of spatial discretization. Due to this modeling approach they are referred to as Distributed Parameter Systems (DPS). Spatial discretization of DPS is done by means of Finite Volume or Finite Element methods and Galerkin or Petro-Galerkin projection techniques and they are simulated in a Computational Fluid Dynamic (CFD) software environment. Although such discretizations approximate the dynamic process behavior reasonably well, it leads to very large order process model. It takes huge computational efforts (time, CPU requirement) to simulate such models and therefore such process models can not be used for online plant optimization and control purposes. Model Order Reduction (MOR) is therefore an important step before proceeding to control design, see e.g. Shvartsman and Kevrekidis [1998]. The method of Proper Orthogonal Decomposition (POD) or Principle Component Analysis (PCA) is widely used for deriving lower dimensional models from the First Principle Model (FPM). The POD method searches for dominant patterns in the given process and defines an optimal, data-dependent basis, that is subsequently used as a projection space to infer reduced order models through Galerkin type of projections, see Astrid [2004] and the references therein. POD methods are empirical (data dependent) in nature and therefore these methods are susceptible to changes in process inputs and process parameters. The reduced model obtained by POD techniques with Galerkin projections are usually very dense and one loses the original sparse model structure. Such a dense model does not always give computational advantage over original full scale CFD model. This motivates one to look for other possible approaches which can give computationally efficient, reliable models which can be used for the online control and optimization purpose. Other motivations for the method proposed in this paper are that in many commercial CFD packages sometimes it becomes impossible to get access to the Partial Differential Equations (PDE) used to implement full scale model and even with access to the PDEs the reduced order modeling efforts can also be very expensive and laborious. In such situations one needs to explore the other possible ways to get a low order model by some identification method. One of such methods is explained in Wattamwar et al. [2008], which uses POD and system identification tools like N4SID algorithms, as explained in e.g. Overschee and Moor [1996], Favoreel et al. [2000]. But the method proposed there results in linear models which are not sufficient for approximating the non-linearities of large scale applications like IGMP. Moreover in the method proposed there, the states of the linear reduced order model have no physical meaning. These problems have motivated us to investigate another model reduction strategy which can approximate process non-linearity. The identification based approach proposed in this paper can be very useful, because it allows to use the available large-scale first principle based detailed non-linear process model in the form of commercial package, not just for the purpose of computationally extremely efficient dynamic

⋆ This work was supported by the European Union within the Marie-Curie Training Network PROMATCH under the grant number MRTN-CT-2004-512441.

process analysis but also for the purpose of design of the process controller and optimization. Therefore the method proposed here helps in minimizing the dependence on the expensive testing of the plant required for the controller design.

This paper is organized as follows. The overall methodology involving necessary tools from system theory like POD, a black-box type of system identification for linear and non-linear polynomial system is explained in section 2. The application/motivation is IGMP and is explained in the section 3. Some results of the proposed method on the motivation problem are presented in the section 4 which is followed by future work and references.

2. THEORY BACKGROUND AND METHODOLOGY

One of the most promising and successful techniques for an efficient reduction of large-scale nonlinear systems in fluid dynamics is the method of Proper Orthogonal Decompositions (POD) also known as the Karhunen-Loève method Holmes et al. [1996]. The method is based on the observation that flow characteristics reveal coherent structures or *patterns* in many processes in fluid dynamics. This has led to the idea that the solutions of model equations may be approximated by considering a small number of dominant coherent structures (called *modes* or *basis*) that are inferred in an *empirical* manner from measurements or simulated data. Given an ensemble of K measurements $\mathbf{T}^k(\cdot)$, $k = 1, \ldots, K$ with each measurement defined on some spatial domain Ω, the POD method amounts to assuming that each observation $\mathbf{T}^k$ belongs to a Hilbert space $\mathcal{H}$ of functions defined on Ω. With the inner product defined on $\mathcal{H}$, it then makes sense to call a collection $\{\varphi_j\}_{j=1}^{\infty}$ an *orthonormal basis* of $\mathcal{H}$ if any element, say $\mathbf{T} \in \mathcal{H}$, admits a representation

$$\mathbf{T}(z) = \sum_{j=1}^{\infty} a_j \varphi_j(z), \quad z \in \Omega \tag{1}$$

Here, the a_j's are referred to as the *modal coefficients*(MC) and the φ_j's are the *modes* or *basis* of the expansion. The truncated expansion

$$\mathbf{T}_n(z) = \sum_{j=1}^{n} a_j \varphi_j(z), \quad z \in \Omega \tag{2}$$

causes an approximation error $\|\mathbf{T} - \mathbf{T}_n\|$ in the norm of the Hilbert space. We will call $\{\varphi_j\}_{j=1}^{\infty}$ a *POD basis* of $\mathcal{H}$ whenever it is an orthonormal basis of $\mathcal{H}$ for which the *total approximation error* in some norm over the complete ensemble is

$$\sum_{k=1}^{K} \|\mathbf{T}^k - \mathbf{T}_n^k\| \tag{3}$$

is minimal for all truncation levels n. This is an *empirical basis* in the sense that every POD basis depends on the data ensemble. Using variational calculus, the solution to this optimization problem amounts to finding the normalized eigenfunctions $\varphi_j \in \mathcal{H}$ of a positive semi-definite operator $R : \mathcal{H} \to \mathcal{H}$ that is defined as

$$\langle \psi_1, R\psi_2 \rangle := \frac{1}{K} \sum_{k=1}^{K} \langle \psi_1, \mathbf{T}^k \rangle \cdot \langle \psi_2, \mathbf{T}^k \rangle \tag{4}$$

with $\psi_1, \psi_2 \in \mathcal{H}$. R is well defined in this manner and corresponds to a positive semi-definite matrix whenever $\mathcal{H}$ is finite dimensional. In that case, a POD basis is obtained from the normalized eigenvectors of R, see e.g. Astrid [2004].

The POD modal coefficients 'a_j' are then obtained by the projection of the ensemble on the span of dominant POD modal coefficients as given by:

$$a_j(k) = \langle \varphi_j(z), \mathbf{T}_n(k, z) \rangle \tag{5}$$

Subsequently, a Galerkin projection is used to obtain the reduced order model as follows. Suppose that the system is governed by a PDE of the form

$$\frac{\partial T_n}{\partial t} = \mathcal{A}(T_n) + \mathcal{B}(u) + \mathcal{F}(T_n, u, d) \tag{6}$$

and let $\mathcal{H}_n$ denote an n dimensional subspace of $\mathcal{H}$ and let $P_n : \mathcal{H} \to \mathcal{H}_n$ and $I_n : \mathcal{H}_n \to \mathcal{H}$ denote the canonical projection and canonical injection maps or operators respectively. The injection map reconstruct the full scale model from reduced space. The reduced model is then given by

$$P_n \frac{\partial T_n}{\partial t} = P_n \mathcal{A}(T_n) + P_n \mathcal{B}(u) + P_n \mathcal{F}(T_n, u, d) \tag{7}$$

where observation $T_n(\cdot, k) = \mathbf{T}_n(k) \in \mathcal{H}_n = P_n \mathcal{H} \quad \forall k$, $\mathcal{A}$ is the spatial operator for convection and diffusion, and is of linear nature $\mathcal{B}$ defines input matrix and $\mathcal{F}$ is nonlinear source term. In the specific case of a POD basis, the finite dimensional subspace $P_n = \text{span}\{\varphi_j\}$, $j = 1, \ldots, n$ where the φ_j's denote POD basis functions. In that case eq. (6) becomes an *ordinary differential equation* in the coefficients $a_j(k)$ in the expansion of T_n as eq. (8) and eq. (9)

$$\frac{\partial \langle P_n, \mathbf{T}_n \rangle}{\partial t} = \mathcal{A}\langle P_n, \mathbf{T}_n \rangle + \mathcal{B}\langle P_n, u \rangle + P_n \mathcal{F}(T_n, u, d) \tag{8}$$

or equivalently,

$$\frac{da_n}{dt} = \mathcal{A}_n a_n + \mathcal{B}_n u + P_n \mathcal{F}(P_n^{-1} a_n, u, d) \tag{9}$$

Eq. (9) is reduced order model (ROM) and the POD modal coefficients 'a_j' are the states of the ROM. Therefore the POD MC can also be viewed as *dominant temporal patterns/dynamics* along which system evolves. The optimization problem to obtain POD basis as mentioned above in eq. (4) equivalently can also be solved for the ensemble $\mathbf{T}_n$ as a 'Singular Value Decomposition' *SVD* which then gives POD basis function (*spatial patterns*) in the form of left singular vectors and POD modal coefficients (*temporal patterns* as singular values multiplied by the right singular vectors. From the property of SVD these patterns are arranged as per their importance, i.e. the first POD basis corresponds to the direction of maximum energy. Usually a tolerance criterion based on amount of energy captured in the reduced model is used to decide the order or the reduced model, i.e. the span of POD basis as defined above in $\mathcal{H}_n$. The criterion is usually called projection energy and is given as below:

$$P_{tol} = \frac{\sum_{k=1}^{r} \lambda_k}{\sum_{k=1}^{n} \lambda_k} \tag{10}$$

where λ_k is the 'k^{th}' eigenvalue of the correlation operator as defined in eq. (4), 'r' is order of ROM and 'n' is order of finite dimensional full scale model. The first two terms of

eq. (9) on RHS are linear and the third non-linear term do not appear for the systems defined by linear PDEs. For the system governed by linear PDEs the differential equation eq. (9) can be transformed in equivalent discrete time form as:

$$a_n(k+1) = A_d a_n(k) + B_d u(k) \tag{11}$$

At this point one can observe that given the ensemble $\mathbf{T}_n$ one can obtain POD basis and corresponding MC, and from this knowledge of MC and system inputs 'u' the system parameters 'A_d' and 'B_d' can be easily estimated by ordinary least square (OLS) estimation techniques. If one now think of the possible approach to identify the system parameters when the governing equations are non-linear like the one in eq. (9), one needs then some approximation for non-linear terms. There are many possible ways to approximate the non-linearities like black-box, neural net, fuzzy logic, grey box, e.g. see Romijn et al. [2008] and many other input-output based fit of Weiner-Hammerstein type. It is also well known that Taylor series expansion of a nonlinear function can be a good approximation of a non-linear function. The use of Taylor Series is not considered in usual input-output identification methods due to the lack of state information. But as explained earlier in the case of model reduction, the states of ROM are accessible and therefore one can make use of Taylor series to approximate the non-linear terms. If one is interested in approximating the original full scale non-linear model then one need to include the Jacobian terms of the Taylor series in ROM. But if the approximation by linear system is not sufficient enough then one must consider the Hessian and other higher terms from the Taylor series. Note that the inclusion of the Hessian terms results into polynomial form of the identified ROM. Replacing the non-linear part by a polynomial system for multi-variable system is cumbersome due to the involvement of the tensor algebra (*Hessian computation*). For this reason we will briefly explain what does a Taylor series expansion for a scalar valued function means and then we will explain it for the vector valued function, and its implementation for the computation purpose. Another interesting feature of he polynomial systems is that they are promising candidates and have structure better suited for analytical analysis and for extension of the notions from linear system theory, e.g. see Ebenbauer et al. [2005]. For a scalar valued function,

$$\dot{x} = f(x)\,, \text{ where } f : \mathcal{R} \to \mathcal{R} \;\; \& f(x^*) = 0 \tag{12}$$

Taylor series expansion in x as a nominal variable and $\tilde{x}$ as a deviation variable, $\tilde{x} = x - x^*$

$$\dot{\tilde{x}} = f(x^*) + f^{'}(x^*)\,\tilde{x} + (1/2!)\, f^{''}(x^*)\,\tilde{x}^2 + ... \tag{13}$$

where, $f^{'}(x) = \mathcal{J}(x) : \mathcal{R} \to \mathcal{R}$, system jacobian operator
$f^{''}(x) = H(x) \;: \mathcal{R} \to \mathcal{R}$, system Hessian operator.
For a vector valued function $f : \mathcal{R}^n \to \mathcal{R}^n$,

the first derivative is defined as a map: $f^{'} : \mathcal{R}^n \to \mathcal{L}(\mathcal{R}^n, \mathcal{R}^n)$, and when the first derivative is evaluated at $x^* \in \mathcal{R}^n$ then $f^{'}(x^*) \in \mathcal{L}(\mathcal{R}^n, \mathcal{R}^n)$, i.e. $f^{'}(x^*)$ is a linear operator, and when it acts on the 'n' dimensional vector 'x' then its image is $\in \mathcal{R}^n$, i.e. $f'(x^*)(x) \in \mathcal{R}^n$. This lets us to understand first derivative as a map, $f' : \mathcal{R}^n * \mathcal{R}^n \to \mathcal{R}^n$. As $f'(x^*)$ is constant term (fixed operator), we better write it as $[f'(x^*)](x) \in \mathcal{R}^n$.

We usually refer the above operator as system Jacobian matrix as, $[f'(x^*)] := \mathcal{J}(x^*)$.

The operator defined in the last expression can be written in terms of partial derivatives as,

$$[f'(x^*)]\,(x) = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1}(x^*) \ldots \dfrac{\partial f_1}{\partial x_n}(x^*) \\ . \\ \dfrac{\partial f_n}{\partial x_1}(x^*) \ldots \dfrac{\partial f_n}{\partial x_n}(x^*) \end{bmatrix} \begin{bmatrix} x_1 \\ . \\ . \\ . \\ x_n \end{bmatrix} \tag{14}$$

equivalently,

$$[f^{'}(x^*)](x) = \begin{bmatrix} \displaystyle\sum_{k=1}^{n} \frac{\partial f_1(x^*)}{\partial x_k} x_k \\ . \\ \displaystyle\sum_{k=1}^{n} \frac{\partial f_n(x^*)}{\partial x_k} x_k \end{bmatrix} \tag{15}$$

The same procedure is repeated for computing the second derivative of the function,

$f'' : \mathcal{R}^n * \mathcal{R}^n * \mathcal{R}^n \to \mathcal{R}^n$, i.e.
$f'' : \mathcal{R}^n \to \mathcal{L}(\mathcal{R}^n, \mathcal{L}(\mathcal{R}^n, \mathcal{R}^n))$, i.e.
$f''(x^*) \in \mathcal{L}(\mathcal{R}^n, \mathcal{L}(\mathcal{R}^n, \mathcal{R}^n))$, i.e.
$f''(x^*)(x) \in \mathcal{L}(\mathcal{R}^n, \mathcal{R}^n)$, i.e.
$f''(x^*)(x)(x) \in \mathcal{R}^n$, i.e. $[f''(x^*)](x, x) \in \mathcal{R}^n$
$[f''(x^*)] := H(x^*)$, system Hessian operator.

It is clear from the above discussions that the Hessian operator is a tensor with argument from two domains while its codomain remains the same that of the Jacobian operator. The linearity of Hessian operator allows us to compute it like the Jacobian operator as in (15), but now with one more argument as:

$$[f^{''}(x^*)](x, x) = \begin{bmatrix} \displaystyle\sum_{k=1}^{n}\sum_{j=1}^{n} \frac{\partial^2 f_1(x^*)}{\partial x_k \partial x_j} x_k x_j \\ . \\ \displaystyle\sum_{k=1}^{n}\sum_{j=1}^{n} \frac{\partial^2 f_n(x^*)}{\partial x_k \partial x_j} x_k x_j \end{bmatrix} \tag{16}$$

the above expression can be written as:

$$[f^{''}(x^*)](x, x) = A_1(x \otimes x) \tag{17}$$

where, $(x \otimes x)$ is the Kroneckar product.

The complete simplification procedure mentioned above is aimed to express, $f'' : \mathcal{R}^n \to \mathcal{L}(\mathcal{R}^n, \mathcal{L}(\mathcal{R}^n, \mathcal{R}^n))$ as, $f'' : \mathcal{R}^n \to \mathcal{L}(\mathcal{R}^{n^2}, \mathcal{R}^n)$. This is possible due to the notion of the linearity of the tensor operator.

From the discussion above, a nonlinear equation of the form $\dot{x} = f(x, u)$ can be expanded in Taylor series as in (13) which can be approximated by a polynomial of the form,

$$\begin{aligned} \dot{x} =& Ax(t) + Bu(t) + A_1(x(t) \otimes x(t)) \\ &+ B_1(u(t) \otimes u(t)) + Q(x(t) \otimes u(t)) \end{aligned} \tag{18}$$

Where, A_1, B_1, Q are equivalent Hessian operators and $x \in \mathcal{R}^n$, $u \in \mathcal{R}^l$, $A \in \mathcal{R}^{n*n}$, $B \in \mathcal{R}^{n*l}$, $A_1 \in \mathcal{R}^{(n*n)*n}$, $B_1 \in \mathcal{R}^{(l*l)*n}$, $Q \in \mathcal{R}^{(l*n)*n}$ and $\otimes$ is the Kronecker products.

These methodological developments are based on CFD software as plant model, so for the moment we are not considering the output equations here.

Equivalent discrete form of Eq. (18) can be written as:

$$x(k+1) = A_d x(k) + B_d u(k) + A_{1d}(x(k) \otimes x(k)) + B_{1d}(u(k) \otimes u(t)) + Q_d(x(k) \otimes u(k)) \quad (19)$$

As we are considering the discrete identification problem here, for the convenience in remaining part of the paper we have dropped the superscript 'd' from eq. (19).

Please note that the polynomial equation (19) is non-linear in states and inputs but it is linear in all the system parameters (equivalent Jacobian and Hessian terms). This is a big advantage. Because if the states and inputs are known then by fixing the above polynomial model structure we can estimate the system parameters by Least Square parameter Estimation (LSE) techniques.

Coming back to the problem of the reduced model identification, the states in the (18) can be seen as POD modal coefficients (MC) and then linear and non-linear part in (9) can be written as (18).

Another interesting feature of the proposed framework is that for a large scale parameter varying systems, given the knowledge of the variation of the time varying parameter, similar approach as proposed above can be used. But the uncertain parameter should then be treated like process inputs and therefore the corresponding process snapshots due to the parameter excitation need to be included while computing the POD basis functions and MCs. As per the knowledge of the author, this approach of model reduction for very large scale process under process parameter uncertainty is never studied in past.

Once the MC and POD basis are obtained from the full scale CFD model as mentioned earlier, then by using the tensors decomposition as in eq.(16) for eq.(18), the problem of polynomial model parameter estimation is an ordinary least square estimation(*OLS*) problem and if we define,

$$\xi_k := col\,(x(k), u(k), (x(k) \otimes x(k)), (x(k) \otimes u(k)), (u(k) \otimes u(k))) \quad (20)$$

then from (19), $x_{k+1} \simeq \Theta\,\xi_k$ Where, $\Theta = [A\,B\,A_1\,B_1\,Q]$ and define the parameter estimation error at each time instance as

$$e_{k+1} = x_{k+1} - \Theta\,\xi_k \quad (21)$$

similarly the estimation error that is minimized by LSE method over the complete simulation horizon 'N' is

$$E := [x_1 \dots x_N] - \Theta[\xi_0 \dots \xi_{N-1}] \quad (22)$$

equivalently, $E := X - \Theta\Xi$
where, N is the number of samples and ,
$X \in \mathcal{R}^{n*(N-1)}$, $\Xi \in \mathcal{R}^{(n+l+n*n+l*l+n*l)*(N-1)}$ and $\Theta \in \mathcal{R}^{n*(n+l+n*n+l*l+n*l)}$

The least square solution will be

$$\Theta = X\,\Xi^T\,(\Xi\,\Xi^T)^{-1} \quad (23)$$

Please make a note here that the system parameter vector Θ is rank deficient due to the involved Kronecker product. Nevertheless, there are some simple ways to estimate the parameters for rank deficient problem as well. We simply used *Matlab* routines for our case.

The complete CFD spatio-temporal information can be reconstructed by projecting back the solution of reduced model (19) on the span of dominant POD basis P_n. The reconstructed CFD state space will be:

$$\tilde{\mathcal{T}}_n(k) = I_n\,a_n(k) = P_n^{-1}\,a_n(k), \text{or equivalently} \quad (24)$$

$$\tilde{\mathcal{T}}_n(k) = \sum_{j=1}^{r} \phi_j^{-1}\,a_j(k) \quad (25)$$

As this study is based on software simulations, the outputs can be chosen as per the user choice. In our study we have decided them close to the real life situation. The constructed output equations can be approximated as:

$$\tilde{y}(k) = C\,\tilde{\mathcal{T}}_n(k) \quad (26)$$

Note that the original Navier-Stokes equation (*non-linear PDEs*) modeled in CFD software are continuous in time and in the approach presented above we have proposed to approximate them by using discrete time linear or polynomial type non-linear equations.
The error involved here will be the sum of projection error and the statistical fit in the identification step to the few selected POD modal coefficients corresponding to the maximum energy content as per eq. (10).

One of the serious drawbacks of this approach is that the OLS estimation method for as described earlier can easily lead to an unstable system, although the original system could be a stable one. Notion of stability is discussed here as divergence of simulation results. We think one of the possible explanation could be the small data set, another could be that the POD MC obtained from SVD are right singular vectors and they are orthonormal vectors. These Orthonormal basis functions (MC) are considered as signals while they are being fitted by using a polynomial model. The orthogonality of vectors is equivalent to the property of uncorrelatedness of signals. Or, orthogonality of MC in terms of the inner product is

$$< a_i,\, a_j > \begin{cases} = 1,\ i = j \\ = 0,\ i \neq j \end{cases} \quad (27)$$

To overcome this drawback of spurious instability one might like to try some other parameter estimation method or to impose the stability in the proposed polynomial model by using some regularization trick. But usually regularization leads to bad performance of the identified model. Moreover regularization if not carried out smartly can lead to completely different dynamics of the identified model. Typically in subspace state space linear model identification techniques, regularization is imposed in the form of forcing the eigenvalues of the identified model to lie in the unit circle, e.g. see Gestel et al. [2000].
In this paper we have not solved the stability issue as the research in polynomial systems is still relatively new and imposing the stability in identification procedure will need considerable amount of further efforts.

3. MOTIVATION: GLASS MANUFACTURING

This section describes the motivating example of Industrial Glass Manufacturing Process, *IGMP*. IGMP is usually carried out in large furnaces which are very well designed in order to have a desired laminar flow pattern of the glass. A 2D view of a typical furnace is shown in figure 1. The flow pattern of glass determines the residence time of the glass in the melting furnace which in turn determines the quality of the glass produced. The process is an example of very large scale integrated systems. Most of the process variables like temperature, velocity, pressure, viscosity are

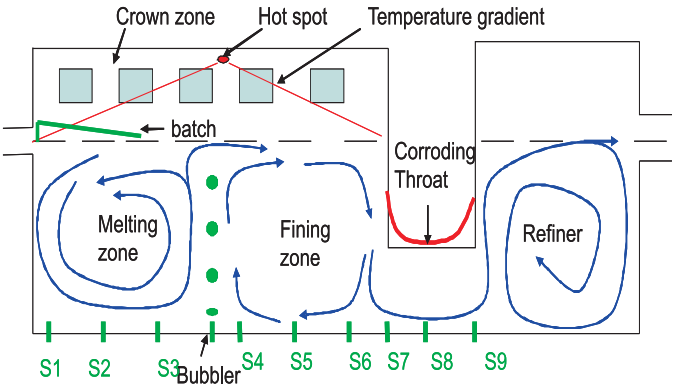

Fig. 1. Glass Manufacturing Furnace

interacting with each other. Due to this interacting nature the control of the furnace has to be done carefully. Usually pull rate (production rate), heat input and pressure valve positions are some of the control variables. Whereas variables of interest are temperature distribution and velocity profiles in the furnace. The product quality is determined by these two factors. The temperature maintained inside the furnace varies between $1400 - 1650$ ^{0}C. The glass raw material enters from the left side (inlet) in the form of a batch blanket, it is melted by applying heat from the top. After circulating through the glass furnace for many hours glass passes through the throat and then leaves via the outlet. Based on the process operation there are roughly three regimes - glass melting, fining to remove high concentration of dissolved gases from the molten glass and refining to remove all remaining undissolved gases from the glass. The IGMP shows large variation in the time constants, from minutes to days. The transport of physical quantities in IGMP can be approximated with reasonable accuracy by modeling it by a set of nonlinear Navier-Stokes equations, see Huisman [2005]. There are many different types of glass furnaces and many different ways to manufacture glass depending on the type of glass required. Most of the glass manufacturing process dynamics are series combination of Continuous Stirrer Tank Reactor (CSTR) and Plug Flow Reactor (PFR). Some more details about mathematical modeling of glass can be found in Huisman [2005], Patankar [1980], Post [1988].

Due to very high process temperature and due to the viscous nature of glass, the glass furnace is a hostile environment for sensor systems. Sensors are largely limited to temperature measurements in the bottom refractory of the melting furnace. As 3 dimensional glass furnace model easily consist of $10^4 - 10^6$ finite elements, simulating its steady and/or dynamic behavior takes days for a normal configured PC and therefore it becomes very difficult to generate and process sufficient data that can be used to develop a model reduction method. For this reason we are using an approximate $2D$ glass furnace which mimics the vertical cross section along the length of $3D$ glass furnace and has only 2 grids cells in width direction.

Currently, apart from modeling the process non-linearity in the reduced order model, we are also trying to model the very slow geometric changes that take place in real $3D$ furnace in the form of throat or dam wall corrosion, see figure 1. This corrosion results into back-flow of molten glass from the refining zone to the fining zone. Such back-flow behavior causes undesired changes in the temperature distribution in the furnace which ultimately leads to economic losses. In this paper we are not addressing the corrosion problem but interested readers can refer Wattamwar et al. [2008] and a Linear Parameter Varying (LPV) system approximation in Wattamwar et al. [2009].

4. RESULTS AND DISCUSSION

In this paper a $2D$ benchmark CFD model of the original process is considered. The full scale CFD model has 3000 cells. It has many variables like temperature, velocity, concentration, pressure, etc. in each grid cell. Although most of the variables are interconnected, for the study here we have considered only temperature as variable of the interest. Therefore the order of the full scale model is 3000. From the method explained in the section 2 we have obtained a fourth order linear and non-linear polynomial model. The choice of fourth order approximation is decided based on the stability issue of polynomial model. Approximation order larger than four leads to an unstable ROM. For the linear model as well, there is not much improvement in the parameter fit above fourth order. This means that for the linear reduced model larger than fourth order there is no way to improve its performance merely by increasing its order, and there is need for non-linear reduced order model. The four POD modal coefficients corresponding to the order approximately capture 80% of the total projection energy. Usually it is desired to capture approximately 99% of the energy of the full model. But due to the stability limitation we can not satisfy this requirement.

The input considered for the identification purpose is pull-rate(feed) in terms of tons/day, which varies 5% around the nominal value in the form of +/- steps superimposed by PRBS signal. This is done to excite the slow and fast dynamics. The simulation horizon is 120 hrs and sampling time is 16 mins, therefore we have 450 snapshots. Like most of the POD related methods, this identification process is very sensitive to the type of input excitation signal. For such complex process it is also very important to know what non-linearity the identification input signal excites. If one excites soft non-linearities for such a complex process then one can expect to get a better and stable polynomial model which would fit more POD MC.

Figure 2 shows the identification result for both linear and polynomial models as proposed in this paper. Figure 3 shows zoomed version of the faster dynamics from the figure 2. Plot shows the result for four outputs which are temperature at the bottom of the four main zones of the glass, viz. Melting, Fining, Throat and Refining section. The sensors are assumed to be placed at the bottom of the tank. This is close to the real life situation. The readers can refer to the figure 1 for sensor locations. $S1$ to $S9$ are the sensors in the figure. Plot shows that the both the models approximates the overall trend very well, but the linear model fails to capture the PRBS signal dynamics precisely compared to the polynomial model.

Figure 4 shows the performance of the two models for the validation signal, which is a step input on the raw-material feed rate. Plot shows that both models follows the trend very well, but both models do not match the time constant and the final gain exactly. This is due to the two reasons. First, this is a distributed system and the

excitation signal used for the identification was designed based on the average time constant of the whole glass tank and it was not designed based on only the four location shown in the figure. Reason for the mismatch of the final gain is that these ROM could not capture 99% of the projection energy of the full scale model. One can expect smaller offset if the approximation order of the reduced model is higher. Unfortunately, as explained earlier in 2, approximation order can not be increased more than 4^{th} for the polynomial form of ROM. Nevertheless, for the size and involved complexity in GMP, even the current results seems to be very interesting.

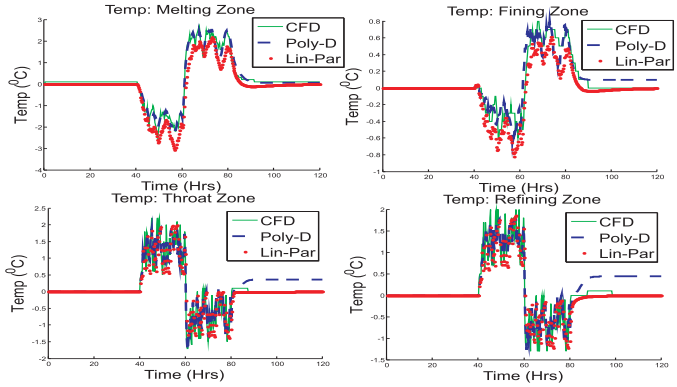

Fig. 2. Model Identification

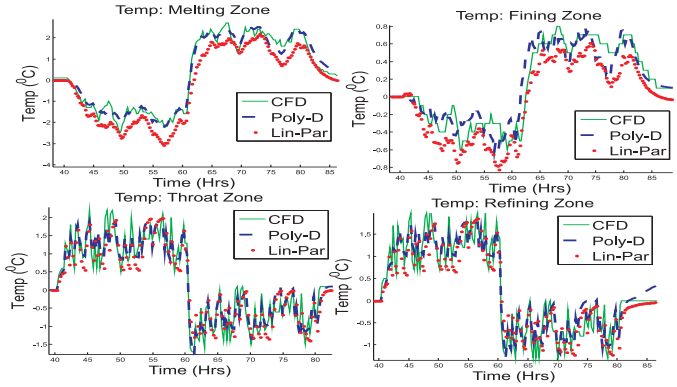

Fig. 3. Model Id: Zoom

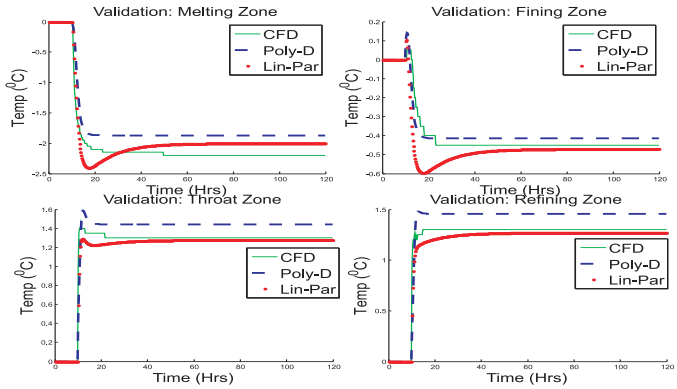

Fig. 4. Model Validation

5. CONCLUSION AND FUTURE RESEARCH

In this paper we have proposed a new model reduction method and its application on large scale industrial application. The proposed method is promising and suited especially for the very large scale processes where complexity reduction by using merely physical insight is not possible. The proposed method is also well formulated in technical aspects and with further improvements in imposing the stability in the identification of polynomial system could make this method of great potential.

We want to explore following topics in near future which has never/rarely been explored in literature like:
1. To investigate the possibility of imposing the stability in the identification process for the polynomial systems.
2. It is also possible to identify multiple linear/polynomial ROM at different working points by the method explained in this paper and construct a non-linear LPV ROM like the one described in Wattamwar et al. [2009].
3. Observer and controller design for polynomial ROM.

REFERENCES

P. Astrid. *Reduction of Process Simulation Models- A proper Orthogonal Decomposition Approach.* PhD report, Technical University of Eindhoven, 2004.

C. Ebenbauer, R. Renz, and F. Allgower. Polynomial feedback and observer design using nonquadratic lyapunov function. *Conference on Decision and Control*, 2005.

W. Favoreel, B. D. Moor, and P. Van Overschee. Subspace state space system identication for industrial processes. *Journal of Process Control*, 10:149–155, 2000.

T. V. Gestel, J. A. K. Suykens, P. V. Dooren, and B. De Moor. Imposing stability in subspace identification by regularization. *Proceedings of IEEE-CDC*, 2000.

P. Holmes, J. L. Lumley, and G. Berkooz. *Turbulence, cohorent structures, dynamical systems and symmetry.* Cambridge University Press: In Cambridge monograph on mechanics, 1996.

L. Huisman. *Control of Glass Melting Processes based on Reduced CFD Models.* PhD report, Technical University of Eindhoven, 2005.

P. Van Overschee and B. De Moor. *Identication for Linear Systems: Theory, Implementation, Applications.* Kluwer Academic, Dordrecht, The Netherlands, 1996.

S.V. Patankar. *Numerical Heat Transfer and Fluid Flow.* Hemisphere, 1980.

L. Post. *Modeling of Flow and Combustion in a Glass Melting Furnace.* PhD report, Technical University of Delft, 1988.

R. Romijn, L. Ozkan, S. Weiland, W. Marquardt, and Ludlage J. A grey-box modeling approach for the reduction of nonlinear systems. *Journal of Process Control*, 18(9):906–914, 2008.

S. Y. Shvartsman and I. G. Kevrekidis. Nonlinear model reduction for control of distributed parameter systems: A computer assisted study. *AIChE Journal*, 44(7):1579–1595, 1998.

S. Wattamwar, S. Weiland, and T. Backx. Identification of low dimensional models for slow geometric parameter variation in an industril glass manufacturing process. *Proceedings of IEEE MSC (CCA)*, 2008.

S. Wattamwar, S. Weiland, and T. Backx. Identification of low dimensional parameter varying models for large scale systems. *sent to the Proceedings of IFAC SYSID*, 2009.