Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Accelerating simulations of first principle models of complex industrial systems using quasi linear parameter varying models

Robert Bos, Xavier Bombois and Paul M. J. Van den Hof

Abstract— **First principles models of complex industrial processes are often derived using finite element or finite difference methods. One of the advantages of these models is that the states in the model have a clear physical interpretation. Using such models we can attempt to monitor or control selected physical quantities, even though they may not be directly measurable. Unfortunately the CPU time associated with each model evaluation of these complex models is often far too large for use in modern online monitoring or control algorithms. This paper introduces a general purpose method to approximate the computationally expensive first principles models with a quasi-Linear Parameter Varying (qLPV) model structure. The CPU time associated with the resulting qLPV models is generally considerably less than the original first principles model. The identification algorithm is such that the physical interpretation of the state vector is retained in the identified model. In contrary to other qLPV identification algorithms, the proposed algorithm extensively utilizes the availability of the original first principles model.**

I. Introduction

Large scale first principles models of complex industrial processes are generally obtained using finite element or finite difference methods with a very fine spatial grid. Consequently, the obtained state-space models are characterized by a very large state vector $x(k)$ (typically $dim(x(k)) \sim 10^3 - 10^9$) and very complex nonlinear functions $f(\cdot)$ and $h(\cdot)$:

$$x(k+1) = f(x(k), u(k)) \quad (1)$$
$$y(k) = h(x(k), u(k)). \quad (2)$$

Consequently, the amount of CPU time required to compute the state and measurement updates using that type of models is generally at least of the order of the sampling interval. The advantages of these detailed first principles models are that the states $x(k)$ have a clear physical interpretation. As a result we can use such a model to monitor or control physical properties of the system even though these properties might not be directly measurable.

Unfortunately in practice the large dimension of $x(k)$ and the large computation time reduces the use of those large scale first principles model to offline simulation studies.

The problem of an overly large state dimension can be solved by using projection based model reduction technique such as Proper Orthogonal Decomposition (POD)[1] or (Empirical) nonlinear Balancing [2][3]. Using a projection based model reduction techniques it is possible to derive reduced order models of the form

$$x_{red}(k+1) = f_{red}(x_{red}(k), u(k)) \quad (3)$$
$$y(k) = h_{red}(x_{red}(k), u(k)). \quad (4)$$

with $\dim(x_{red}(k)) << \dim(x(k))$. After reduction the reduced order state $x_{red}(k)$ still has a physical interpretation, because there exists a known simple mapping that can accurately reconstruct the full order state $x(k)$ from a reduced order state $x_{red}(k)$.

Unfortunately model reduction does not necessarily reduce the computational effort involved to evaluate (3)-(4) [4]. To enable state estimation and other related online tasks for complex industrial processes, we thus require in addition to state reduction techniques, techniques that can find new models that approximate (3)-(4), but require considerably less CPU time to perform simulations. The goal in this paper is to find generically applicable algorithms that accomplish this task. Since by far the most CPU time is generally spent evaluating (3), we will only consider methods that attempt to find faster models for (3).

Methods available in the literature that attempt to reduce the computational effort for simulations with large scale first principles can roughly be divided into four main categories.

The first category of methods consists of either optimizing the spatial grid of the finite element method generating the large-scale model (1)-(2) or using simpler approximate physical relations per grid cell. While good results can be obtained in this manner, this approach is highly problem specific, and requires expensive specialists to perform the model simplification.

A very popular second approach to deriving a faster simulation model is to linearize the (1)-(2) in a chosen working point before performing projection based model reduction. For linear models, projection based model reduction instantly results in a model that requires considerably less computational effort for performing model evaluations. A drawback of this method is that linearization only results in reasonably accurate models if the original system was already close to linear.

A third approach consists of partitioning methods [5][6]. Partitioning methods split the original state $x(k)$ into two parts: $x^{[1]}(k)$ and $x^{[2]}(k)$. The model (1) is only used to compute $x^{[1]}$, the remaining states are reconstructed using linear methods.

The fourth and final approach encountered in the literature to find faster approximative models is an identification based

The work of Robert Bos is supported by TNO Science and Industry

All authors are with the Delft Center for Systems and Control (DCSC), Faculty of Mechanical Engineering, Delft University of Technology, 2628 CD Delft, The Netherlands r.bos@dcsc.tudelft.nl

0-7803-9568-9/05/$20.00 ©2005 IEEE

approach. In [7] equation (3) is approximated using a linear state space model that has the state of the reduced order model as outputs:

$$\zeta(k+1) = \hat{A}\zeta(k) + \hat{B}u(k) \quad (5)$$
$$x_{red}(k) = \hat{C}\zeta(k) + \hat{D}u(k). \quad (6)$$

The state space matrices $\hat{A}, \hat{B}, \hat{C}, \hat{D}$ are identified using a subspace estimator on simulation data generated by (3). The resulting model is a linear approximation to the nonlinear model. The main drawback of the method is that if the original model shows significant nonlinearities, the usefulness of the identified model is limited.

In this paper we will introduce a new identification based algorithm to approximate (3) with a model that requires considerably less CPU time. Whereas [7] used a single linear model to approximate (3) we shall approximate (3) with a model $f_{id}(x_{red}(k), u(k), \theta)$ that has an quasi-Linear Parameter Varying model structure[1]:

$$f_{id}(x_{red}(k), u(k)) = A_0 x_{red}(k) + B_0 u(k) + x_{off,0} + \sum_{m=1}^{M} \phi_m(x_{red}, u(k), \theta_m) \left[A_m x_{red}(k) + B_m u(k) + x_{off,m}\right]. \quad (7)$$

In model structure (7) $(A_0, B_0, x_{off,0})$ are state space matrices that form a global linear approximation of (3) and $(A_m, B_m, x_{off,m})$ are component linear models that are summed after being weighted according to scheduling functions $\phi_m(\cdot)$ to allow the behavior of the identified model to differ from the basic linear model $(A_0, B_0, x_{off,0})$. The scheduling functions $\phi_m(\cdot)$ determine how the behavior of the identified model should change, depending on the current operating point. It can thus be interpreted as an extension of the method described in [7]. Like in [7] we will need to identify matrices $(A_0, B_0, x_{off,0})$ and $(A_m, B_m, x_{off,m})$ and functions $\phi_m(\cdot)$, for $m = 1, \ldots, M$ such that the $f_{id}(\cdot)$ is as close as possible to the reduced order model $f_{red}(\cdot)$.

Even using a relatively low number of component models and relatively simple scheduling functions $\phi_m(\cdot)$ a wide variety of nonlinear models can be approximated. As a result, the CPU time required to evaluate (7) is generally much lower than the time required to evaluate (3). Drawback of using the qLPV model structure are that even for a relatively low number of components models M the identification problem is already much more difficult than for linear models. Not only is it necessary to determine a single linear model given by $A_0, B_0, x_{off,0}$, we also need to find optimal component models $A_m, B_m, x_{off,m}$ and scheduling functions $\phi_m(\cdot)$.

The problem of identifying qLPV models has been extensively studied in nonlinear identification literature. Current methods to identify model structure of the form (7) can be divided into the following categories:

1) identify scheduling functions $\phi_m(\cdot)$ and models $A_m, B_m, x_{off,m}$ for $m = 0, \ldots, M$ simultaneously, see for instance [8][9][10],
2) functions $\phi_m(\cdot)$ are assumed known, identify only $A_0, B_0, x_{off,0} \ldots A_M, B_M, x_{off,M}$,
3) two stage methods: first determine scheduling functions $\phi_m(\cdot)$ then identify $A_m, B_m, x_{off,m}$ for $m = 0, 1, \ldots, M$. [11][12].

All the identification approaches in the literature use experimental data consisting of measured inputs $u(k)$ and measured outputs $y(k)$ to perform the identification. A plant model of the form (3) is never assumed available. As a result the state vector $x_{red}(k)$ of the identified qLPV model will not have the same physical interpretation as in (3).

In this paper we will introduce a new method to identify qLPV models. There are three main differences between our approach and the approaches in the literature. The first difference is that our procedure will use simulation data $Z^N = \{u(1), x_{red}(1), \ldots, u(N), x_{red}(N)\}$ generated with (3) instead of measured input and output signals. The advantages of using simulation data instead of measured data are threefold:

- in simulation data we know the state $x_{red}(k)$ which will allow us to retain the physical interpretation of the state vector $x_{red}(k)$ in the identified model,
- to obtain suitable practical data (expensive) experiments may be required, while simulation data only requires computer time,
- there is no measurement noise in simulation data.

The second difference between our method and qLPV identification methods in literature is that we will not only use the available simulation data for the identification, but also we will make extensive use of the knowledge of the model (3).

The third difference is that using our method the both the required number of component models M and the component models $A_m, B_m, x_{off,m}$ themselves for $m = 1, \ldots, M$ can be easily determined using a singular value decomposition.

The remainder of this paper is organized as follows; first we will provide an outline of our proposed identification method in section II. Then in sections III to VI we will discuss the proposed algorithm in more detail. In section VII the effectiveness of the proposed algorithm is illustrated in simulation example. Finally the paper ends with a summary of the main results.

II. Overview identification procedure

This section outlines our proposed identification procedure. The goal of the procedure is to find a quasi-LPV model (7) that forms a good approximation to the orginal reduced order model $f_{red}(\cdot)$ defined in (3). As mentioned in the introduction, we will use simulation data $Z^N = \{x_{red}(1), u(1), \ldots, x_{red}(N), u(N)\}$ generated with the known system (3). It is assumed this data is generated such that the following assumption holds:

[1]We will refer to model structures specified by (7) as quasi-Linear Parameter Varying (qLPV) models, but in the literature these model structures are also known under the names local linear models or fuzzy models.

Assumption 1: We have generated data $Z^N = \{u(1), x_{red}(1), \ldots, u(N), x(N)\}$ using the reduced order model (3). It will be assumed that the available data Z^N is representative for the whole working area of the reduced order model (3).

The outline of the identification procedure to identify the quasi-LPV model (7) is given below:

1) Determine $A_0, B_0, x_{off,0}$ such that the linear model given by $A_0, B_0, x_{off,0}$ is minimizes the residual signal $\psi(k)$

$$[A_0, B_0, x_{off,0}] = \arg\min \sum_{k=1}^{N} \|\psi(k)\|^2, \tag{8}$$

with $\psi(k)$ defined as:

$$\psi(k) = f_{red}(x_{red}(k), u(k)) - [A_0 x_{red}(k) + B_0 u(k) + x_{off,0}]. \tag{9}$$

2) Compute time-dependent matrices $A_k^\psi, B_k^\psi, x_{off,\psi}(k)$ such that:

$$\psi(k) = A_k^\psi x_{red}(k) + B_k^\psi u(k) + x_{off,k}^\psi. \tag{10}$$

The computed time dependent matrices A_k^ψ, B_k^ψ and $x_{off,k}^\psi$ will be used the step to determine the fixed matrices $A_m, B_m, x_{off,m}$ of our final qLPV expansion.

3) In the previous step we created an expansion of the residual $\psi(k)$ in time-varying matrices A_k^ψ, B_k^ψ and $x_{off,k}^\psi$. Since we want to identify a quasi-LPV structure of the form (7) we would like an expansion of $\psi(k)$ in $M << N$ fixed matrixes $A_m, B_m, x_{off,m}$ and scheduling coefficients $\beta_m(k)$ for $m = 1, \ldots, M$ such that for the simulation data in Z^N:

$$\frac{1}{N}\sum_{k=1}^{N} \left\| \psi(k) - \sum_{m=1}^{M} \beta_m(k) \left[A_m x_{red}(k) + B_m u(k) + x_{off,m}\right] \right\|^2 < \alpha. \tag{11}$$

with α a small user defined constant.
As will be explained in section V we will determine the required number of component models M, the component models $A_m, B_m, x_{off,m}$ and the time dependent scheduling coefficients $\beta_m(k)$ using a Singular Value Decomposition (SVD) in the 'parameter-domain' of the time-dependent matrices $A_k^\psi, B_k^\psi, x_{off,k}^\psi$.

4) The scheduling coefficients $\beta_m(k)$ computed in the previous step are unstructured coefficients while in fact they depend on current reduced order state $x_{red}(k)$ and input $u(k)$. In the final step of our identification procedure, as elaborated in section VI, we will determine structured functions $\phi_m(x_{red}(k), u(k))$ to replace coefficients $\beta_m(k)$ in (11) such that

$$\frac{1}{N}\sum_{k=1}^{N} \left\| \psi(k) - \sum_{m=1}^{M} \phi_m(x_{red}(k), u(k)) \left[A_m x_{red}(k) + B_m u(k) + x_{off,m}\right] \right\|^2 < \alpha. \tag{12}$$

III. Step 1: Determination of A_0, B_0 and $x_{off,0}$

In the first step of the proposed quasi-LPV identification procedure we determine matrices A_0, B_0 and $x_{off,0}$. In the proposed identification algorithm we choose to determine these matrices that the obtained matrices minimize the residual signal $\psi(k)$ as defined in (9):

$$[A_0, B_0, x_{off,0}] = \arg \min_{A,B,x_{off}} \sum_{k=1}^{N} \|f(x_{red}(k), u(k)) - Ax_{red}(k) - Bu(k) - x_{off}\|^2. \tag{13}$$

Since we chose to use simulation data for the identification procedure, the states $x_{red}(k)$ and inputs $u(k)$ are known. As a result the criterion (13) is linear in A, B, x_{off} and thus A_0, B_0 and $x_{off,0}$ can be easily determined by solving a linear least squares minimization problem.

IV. Step 2: Determination of A_k^ψ, B_k^ψ, and $x_{off,k}^\psi$

In this second step of the algorithm we shall construct time-dependent matrices A_k^ψ, B_k^ψ and $x_{off,k}^\psi$ such that (10) holds. Note that such a expansion is not unique. Since we will later use the expansion (10) to compute the quasi-LPV model we will choose a particular expansion. In this expansion, the matrices A_k^ψ, B_k^ψ are chosen such that:

$$A_0 + A_k^\psi = \left.\frac{\partial f_{red}(x,u)}{\partial x}\right|_{x=x_{red}(k), u=u(k)} \tag{14}$$

$$B_0 + B_k^\psi = \left.\frac{\partial f_{red}(x,u)}{\partial u}\right|_{x=x_{red}(k), u=u(k)}. \tag{15}$$

$$x_{off,0} + x_{off,k}^\psi) = f_{red}(x_{red}(k), u(k)) - [A_0 + A_k^\psi]x_{red}(k) - [B_0 + B_k^\psi]u(k) \tag{16}$$

In other words, the matrices $A_\psi(k), B_\psi(k), x_{off,\psi}(k)$ are chosen such that $A_0 + A_\psi(k), B_0 + B_\psi(k)$ and $x_{off,0} + x_{off,\psi}(k)$ form the linearization of $f_{red}(\cdot)$ at time index k. As will be explained in the next section this particular choice for the expansion (11) will allow us to easily compute the fixed matrices $A_m, B_m, x_{off,m}$ for $m = 1, \ldots, M$ using a simple SVD operation.

V. Step 3: Determination of fixed matrices $A_m, B_m, x_{off,m}$ and time dependent coefficients $\beta_m(k)$ for $m = 1 \ldots, M$

By construction (see (9)) it holds that for each data pair $\{x_{red}(k), u(k)\} \in Z^N$:

$$f_{red}(x_{red}(k), u(k)) = A_0 x_{red}(k) + B_0 u(k) + x_{off,0} + \psi(k) \tag{17}$$

with $\psi(k)$ the discrepancy between the linear model $A_0, B_0, x_{off,0}$ and the reduced order model $f_{red}(\cdot)$. In the previous section, we provided an expansion of the prediction error $\psi(k)$ in a set of time dependent state space matrices:

$$\psi(k) = A_k^\psi x_{red}(k) + B_k^\psi u(k) + x_{off,k}^\psi. \tag{18}$$

In this third step of our methodology, we will determine $M << N$ models $(A_m, B_m, x_{off,m})$ $(m = 1, \ldots, M)$ and M scheduling coefficients $\beta_m(k)$ $(m = 1, \ldots, M)$, $(k = 1, \ldots, N)$ such that (11) holds.

To abbreviate the notation in this section we introduce the following notation. Define parameter vectors $\mathcal{V}_m^s$ for $m = 1, \ldots, M$ as:

$$\mathcal{V}_m^s = [\text{vec}(A_m)^T \ \text{vec}(B_m)^T \ x_{off,m}^T]^T. \tag{19}$$

with vec$(\cdot)$ the operator that transforms an arbitrary matrix into a vector, by stacking its columns. Similarly we define parameter vectors $\mathcal{V}_{\psi,k}$ for $k = 1, \ldots, N$ as

$$\mathcal{V}_{\psi,k} = [\text{vec}(A_k^{\psi})^T \ \text{vec}(B_k^{\psi})^T \ x_{off,k}^{\psi \ T}]^T. \tag{20}$$

Using this notation the problem of finding appropriate matrices $A_m, B_m, x_{off,m}$ such that (11) holds is equivalent the problem of finding parameter vectors V_m^s for $m = 1, \ldots, M$.

The problem of finding vectors $\mathcal{V}_m^s$ such that (11) is satisfied is not trivial, because each vector $\mathcal{V}_m^s$ represents a linear model. We propose a procedure in the parameter space of the vectors $\mathcal{V}_m^s$ to compute the required component models. This procedure is based on the assumption that if for all $k = 1, \ldots, N$

$$\mathcal{V}_{\psi,k} \approx \sum_{m=1}^{M} \beta_m(k) \mathcal{V}_m^s \tag{21}$$

then we also have that

$$A_k^{\psi} \approx \sum_{m=1}^{M} \beta_m(k) A_m, \tag{22}$$

$$B_k^{\psi} \approx \sum_{m=1}^{M} \beta_m(k) B_m, \tag{23}$$

$$x_{off,k}^{\psi} k \approx \sum_{m=1}^{M} \beta_m(k) x_{off,m} \tag{24}$$

and thus

$$\psi_k \approx \sum_{m=1}^{M} \beta_m(k) \left(A_m x_{red}(k) + B_m u(k) + x_{off,m} \right). \tag{25}$$

Procedure 1: The following algorithm can be used to determine parameter vectors $\mathcal{V}_1^s, \ldots, \mathcal{V}_M^s$ such that (11) holds:

1) Set $\lambda = 1$
2) Compute parameter vectors $\mathcal{V}_1^s, \ldots, \mathcal{V}_\lambda^s$ and coefficients $\beta_1(k), \ldots, \beta_\lambda(k)$ for $k = 1, \ldots, N$ such that:

$$\begin{aligned} &\mathcal{V}_1^s, \ldots, \mathcal{V}_\lambda^s, \beta_1(1), \ldots, \beta_\lambda(N) \\ &= \arg \min_{\tilde{\mathcal{V}}_1^s, \ldots, \tilde{\mathcal{V}}_\lambda^s, \tilde{\beta}_1(1), \ldots, \tilde{\beta}_\lambda(N)} \sum_{k=1}^{N} \left\| \mathcal{V}_{\psi,k} - \sum_{m=1}^{\lambda} \tilde{\beta}_m(k) \tilde{\mathcal{V}}_m^s \right\|^2, \end{aligned} \tag{26}$$

The criterium above results in those parameter vectors $\mathcal{V}_1^s, \ldots, \mathcal{V}_\lambda^s$ such that averaged over the identification set Z^N we can construct the parameter vectors $\mathcal{V}_{\psi,k}$ for $k = 1, \ldots, N$ as a linear combination of the λ parameter vectors $\mathcal{V}_1^s, \ldots, \mathcal{V}_\lambda^s$ with the smallest possible error. The parameter vectors $\mathcal{V}_1^s, \ldots, \mathcal{V}_\lambda^s$ and coefficients $\beta_1(k), \ldots, \beta_\lambda(k)$ for $k = 1, \ldots, N$ can be efficiently computed using an SVD operation.

3) Compute the prediction error of the short expansion specified by $\mathcal{V}_1, \ldots, \mathcal{V}_\lambda$ and coefficients $\beta_1(k), \ldots, \beta_\lambda(k)$. If the computed error satisfies (11) then $M = \lambda$ and stop. Otherwise, $\lambda = \lambda + 1$ and goto step 2.

The advantage of using the parameter space criterion is that the parameter vectors $\mathcal{V}_1^s, \ldots, \mathcal{V}_\lambda^s$ and functions $\beta_1(k), \ldots, \beta_\lambda(k)$ can be computed efficiently using the SVD. Using the vectors $\mathcal{V}_\psi(1), \ldots, \mathcal{V}_\psi(N)$, we can construct a data matrix X:

$$X = [\mathcal{V}_{\psi,1} \ \cdots \ \mathcal{V}_{\psi,N}]. \tag{27}$$

If the singular value decomposition of X is denoted as:

$$X = USV^T, \tag{28}$$

then the parameter vectors $\mathcal{V}_1^s, \ldots, \mathcal{V}_\lambda^s$ which satisfy (26) are equivalent to the first λ columns of U.

The corresponding matrices A_m, B_m, x_{off_m}, $m = 1, \ldots, \lambda$ can be determined by applying the inverse of the vec$(\cdot)$ operator to the obtained vectors:

$$[A_m \ B_m \ x_{off,m}] = \text{vec}^{-1}(\mathcal{V}_m^s). \tag{29}$$

The coefficients $\beta_1(k), \ldots, \beta_\lambda(k)$ in (26) can be easily computed via:

$$\beta_m(k) = \mathcal{V}_m^{s \ T} \mathcal{V}_{\psi,k}. \tag{30}$$

Note that using our procedure in the parameter space allows us to easily determine the number of component models M, which is often a problem is other algorithms.

Secondly, also note that the described method of determining the expansion (11) is not guaranteed to result in the shortest possible expansion that satisfies (11). However the expansions obtained using procedure 1 will in general satisfy (11) with $M << N$.

VI. Step 4: Identification of structured scheduling functions $\phi_i(x_{red}(k), u(k), \theta)$

In the previous steps matrices $A_0, B_0, x_{off,0} \ldots, A_M, B_M, x_{off,M}$ were determined such that for the available identification data there exists a scheduling coefficients $\beta_m(k)$ such that the one step ahead prediction error (11) was smaller that a chosen value α. The criterion was met using an unstructured set of scheduling coefficients $\beta_m(k)$ as computed in (30).

In practice the optimal scheduling functions of the final qLPV models should be known structured function of both the reduced order $x_{red}(k)$ and inputs $u(k)$. In this fourth step of the identification procedure structured scheduling functions $\phi_m(x_{red}(k), u(k), \theta_m)$ will be identified using the available simulation data Z^N.

For this purpose we first choose a model structure for the functions $\phi_m(x_{red}(k), u(k), \theta_m)$. Since the functions

$\phi_m(\cdot, \theta_m)$ are not required to have any physical interpretation, we are free to chose any model structure we would like. The choice for structure of these functions is a tradeoff between complexity and flexibility.

An example of a simple model structure is:

$$\phi_m(x_{red}(k), u(k), \theta_m) = [x_{red}(k)^T \ u(k)^T \ 1]\theta_m. \quad (31)$$

The simple affine model structure for $\phi_m(\cdot, \theta_m)$ has as main advantages that the structure is linear in its parameter vector θ_m and the number of parameters is relatively small. Drawback of this model structure is that such a structure may not be flexible enough to allow for an accurate model (7).

If more complex scheduling functions are required, it is possible to use more complex structures such as radial basis functions [13] or fuzzy membership functions [12]. To select an appropriate structure for the scheduling functions, it is often helpful to examine plots of the computed coefficients $\beta_m(k)$ (see (30)) as a function of the states and inputs $\{x_{red}(k), u(k)\} \in Z^N$.

Once a structure has been selected, all that remains is to estimate the parameter vectors θ_m. This can be accomplished by minimizing the prediction error criterion for the available data Z^N:

$$\theta_1, \ldots, \theta_M = \arg \min_{\theta_1, \ldots, \theta_M} \frac{1}{N} \sum_{k=1}^{N} \| f_{red}(x_{red}(k), u(k)) - f_{id}(x_{red}(k), u(k), \theta_1, \ldots, \theta_M) \|^2, \quad (32)$$

with $f_{id}(\cdot)$ defined as in (7) using matrices $A_m, B_m, x_{off,m}$ as determined in section V.

Note that if the scheduling functions $\phi_m(\cdot, \theta_m)$ are chosen linear in θ_m such as in (31), the resulting minimization problem is a linear least squares problem which can be easily solved. For more complex structures that are not linear in θ_m, the parameters of the scheduling functions have to be determined using nonlinear optimization techniques. For the previously mentioned structures involving radial basis functions or fuzzy membership functions, good initial conditions can be obtained by applying clustering methods on the computed coefficients $\beta_m(k)$.

VII. Simulation example

To illustrate the effectiveness of the presented quasi-LPV identification method, the method will be applied in a simulation example. In the simulation example we use the techniques outlined in sections II through VI to approximate a finite element model of a heated iron slab that can be heated or cooled at its edges. To prevent confusion, we shall use (p, q) instead of the more common (x, y) for horizontal and vertical coordinates within the slab. If we assume that heat is only exchanged in the $p-q$ plane the first principles model for the slab can be easily derived using the following energy balance:

$$\rho c_p (dp)(dq) h \frac{dT(p, q, t)}{dt} = \nabla \cdot J(p, q, t) \quad (33)$$

with c_p the specific heat of the iron, h the height of the slab, $T(p, q)$ the temperature at location (p, q) and $J(p, q, t)$ a vector containing the flow of thermal energy. The energy flow $J(p, q, t)$ is given by:

$$J(p, q) = \lambda(T(p, q, t)) \nabla(T(p, q, t)), \quad (34)$$

with $\lambda(T(p, q, t))$ a temperature dependent heat conductivity coefficient. For constant $\lambda(T(p, q, t))$, the heated plate model would be a linear model. In this example however, we chose

$$\lambda(T) = 80 + \frac{3}{8} T + \frac{1}{2560} T^3, \quad (35)$$

and thus the heated slab model in nonlinear. To derive a model for the heat distribution of the entire slab, the equations (33)-(35) are solved on a 32×32 spatial grid, using a finite differences approach. After numerical integration using a step size of 20 seconds using a simple explicit Euler method, the model can be written in the form (1). This model was finally reduced to a model of order 25 using a POD technique [1]. The resulting model is of the form (3), with $\dim(x_{red}(k)) = 25$ and $\dim(u(k)) = 4$ (i.e. the temperature at each each side of the slab).

To approximate the reduced order first principles model of the heated slab with a qLPV model of the form (7), we first generated $N = 3000$ pairs of simulation data $Z^N = \{u(1), x_{red}(1), \ldots, u(N), x_{red}(N)\}$.

After generating the required simulation data the first step in the qLPV identification algorithm is to estimate a global linear model with matrices $A_0, B_0, x_{off,0}$. These matrices were estimated by solving the least squares problem (13). The global linear model is used to compute the residual $\psi(k)$. Then in the second step of the identification algorithm the residual $\psi(k)$ is rewritten function of time-varying matrices A_k^ψ, B_k^ψ and $x_{off,k}^\psi$. For this purpose the reduced order nonlinear model was linearized for all $\{x_{red}(k), u(k)\} \in Z^N$. The obtained expansion (10) was used to construct the matrices $A_m, B_m, x_{off,m}$ for the shorter expansion, for $m = 1, \ldots M$ using Procedure 1. This resulted in $M = 8$ component models for $\alpha = 0.1$. The prediction error of the shorter expansion (11) over the simulation data Z^N using time functions $\beta_m(k)$ computed via (30) is 0.078. To put this number in perspective, the prediction error of the linear model specified by $A_0, B_0, x_{off,0}$ is 4.5, a factor 58 times larger!

The final step in the qLPV identification algorithm is to identify the scheduling functions $\phi_m(\cdot)$. To choose an appropriate structure for the scheduling functions we used examined plots of the coefficients $\beta_m(k)$ as a function of states $x_{red}(k)$ and inputs $u(k)$. From these plots we concluded that the functions $\beta_m(k)$ mainly seem to depend linearly on the first eight rows of the state vector $x_{red}(k)$ (denoted by $[x_{red}(k)]_{1:8}$) and all inputs $u(k)$. Apart from this linear dependence, functions $\beta_m(k)$ appear to depend on the both the first four rows $[x_{red}(k)]_{1:4}$) and inputs $u(k)$ to the power of 3. This was of course to be expected given that the heat conductivity function $\lambda(T)$ (see (35)) is a third order polynomial. As a result, the following structure for

TABLE I

AVERAGED PREDICTION ERRORS AND COMPUTATION TIME OF CONSTRUCTED APPROXIMATE FASTER MODELS OF HEATED PLATE MODELS. AVERAGED PREDICTION ERRORS PER GRID CELL AND COMPUTATION TIME WERE DETERMINED FOR 1000 POINTS OF VALIDATION DATA.

Model	Err	CPU time
$f_{red}(\cdot)$	0	307 s.
Identified Linear model	4.8	3 s.
Identified quasi-LPV model	0.1	36 s.

$\phi_m(\cdot)$ was selected:

$$\phi_m(x_{red}(k), u(k), \theta_m) = \left[1 \; [x_{red}]_{1:8}^T \; [(x_{red})^3]_{1:4}^T \; u(k)^T \; u(k)^{3^T}\right], \quad (36)$$

with $\theta_m \in \mathbb{R}^{21\times 1}$ parameter vector that is still to be determined. Since the chosen model structure for $\phi_m(\cdot)$ is linear in θ_m, the parameter vectors can be estimated by solving linear least squares problem (32).

To test the quality of the identified qLPV model, 1000 points of new validation data has been generated and the prediction error of the identified qLPV model has been determined. For reference the prediction error of the linear model specified by only $A_0, B_0, x_{off,0}$ was also computed. Results are presented in Table I. In the table we see that even though the linear model is already fairly accurate, the qLPV model is approximately 48 times more accurate than the linear model. The Table also lists the computation times to generate the predictions using each of the models. The CPU time for the qLPV model is more that 8 times lower than the original reduced order model, even though we only used a very simple explicit Euler solver to derive our first principles model.

VIII. CONCLUSION

This papers describes a method of approximating a known complex first principles model (obtained with finite element/differencing methods) with a simpler structures. The chosen structure of the approximation models is a state-dependent linear combination of affine component models, a structure also used in LPV and fuzzy identification literature.

To optimally use knowledge contained in the known first principles models, a new procedure is introduced to derive the approximation model. Instead of using experimental data, only simulation data generated with the known first principles model is used. Apart from using simulation data instead of experimental data, this paper also introduces a novel method that allows using the known simulation model to easily determine the optimal component models.

REFERENCES

[1] P. Holmes, J. L. Lumley, and G. Berkooz, *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge University Press, 1996.

[2] J. M. A. Scherpen, "Balancing for nonlinear systems," *Systems and Control Letters*, pp. 143–153, 1993.

[3] S. Lall, J. E. Marsden, and S. Glavaski, "A subspace approach to balanced truncation for model reduction of nonlinear control systems," *International Journal on Robust and Nonlinear Control*, vol. 12, pp. 519–535, 2002.

[4] W. Marquardt, "Nonlinear model reduction for optimization based control of transient chemical processes," *AIChE Symp. Ser. 326*, vol. 98, pp. 12–42, 2002.

[5] R. Bos, X. Bombois, and P. Van den Hof, "Accelerating large-scale non-linear models for monitoring and control using spatial and temporal correlations," in *Proceedings of American Control Conference 2004*, Boston, 2004, pp. 3705–3710.

[6] P. Astrid, "Reduction of process simulation models - a proper orthogonal decomposition approach," Ph.D. dissertation, Eindhoven University of Technology, 2004.

[7] L. Huisman and S. Weiland, "Identification and model predictive control of an industrial glass-feeder," in *Proceedings 13th IFAC Symposium on System Identification*, Rotterdam, 2003, pp. 1685–1689.

[8] V. Verdult, L. Ljung, and M. Verhaegen, "Identification of composite local linear state-space models using a projected gradient search," *Int. J. Control*, vol. 75, pp. 1385–1398, 2002.

[9] B. Bamieh and L. Giarré, "Identification of linear paramter varying models," *Int. J. Nonlinear Control*, vol. 12, pp. 841–853, 2002.

[10] L. H. Lee and K. Poolla, "Identification of linear parameter-varying systems using nonlinear programming," *J. of Dynamic Systems, Measurement and Control*, vol. 121, pp. 71–78, 1999.

[11] T. A. Johansen and B. A. Foss, "Identification of non-linear system structure and parameters using regime decomposition," *Automatica*, vol. 31, no. 2, pp. 321–326, 1995.

[12] R. Babuška, *Fuzzy Modeling for Control*. Boston/Dordrecht/London: Kluwer Academic, 1998.

[13] V. Verdult, "Nonlinear System Identification - A State-Space Approach," Ph.D. dissertation, University of Twente, 2002.