Computers and Chemical Engineering 97 (2017) 104–113

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Automatic model reduction of differential algebraic systems by proper orthogonal decomposition

Dmytro Khlopov[a,*], Michael Mangold[b]

[a] Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstraße 1, 39106 Magdeburg, Germany
[b] Technische Hochschule Bingen, Berlinstr. 109, 55411 Bingen, Germany

ARTICLE INFO

Article history:
Received 4 August 2016
Received in revised form 7 October 2016
Accepted 6 November 2016
Available online 15 November 2016

Keywords:
Nonlinear model reduction
Proper orthogonal decomposition
Empirical interpolation
Computer aided modeling
Differential algebraic systems

ABSTRACT

Proper orthogonal decomposition (POD) is an attractive way to obtain nonlinear low-dimensional models. This article reports on the automatization of the mentioned reduction method. An automatic procedure for the reduction of differential algebraic systems is presented, which is implemented in the modeling and simulation environment ProMoT/Diana. The software tool has been applied to a nonlinear heat conduction model and a continuous fluidized bed crystallizer model. The automatically generated reduced models are significantly smaller than the reference models, while the loss of accuracy is negligible.

1. Introduction

Many modern mathematical models of real-life processes impose difficulties when it comes to their numerical solution. This holds especially for models represented by nonlinear distributed parameter systems, which are frequent in engineering. Usually, for the numerical solution of distributed parameter systems the original system of infinite order is approximated by one with a finite system order by a semi-discretization, which results in a system of differential algebraic equations. The resulting number of degrees of freedom is usually very high and makes the use of the discretized model inconvenient for model-based process design, process control and optimization (Shi et al., 2006). Thus there is a need for reduced models. Through model reduction, a small system with reduced number of equations is derived. The numerical solution of reduced models should be much easier and faster than the solution of the original problem. On the other hand, the reduced model should be able to reproduce the system behavior with sufficient accuracy in the relevant window of operation conditions and in the relevant range of system parameters.

Various methods for nonlinear and linear model reduction have been proposed, particularly in the areas of electrical and mechanical engineering, control design and computational fluid dynamics. Some of them are based on physical simplifications like assumption of perfect mixing, introduction of compartments, equilibrium assumptions, etc. This approach requires physical insight of the modeler and hence is hard to automatize. Another successful approach, which may also be considered as a physical model reduction method, is based on nonlinear wave propagation theory (Marquardt, 1990; Kienle, 2000). It produces reduced model by approximation of the spatially distributed solution by profile with a given shape. As in the previous case, this method requires physical process understanding from the user and can be applied only for special systems. The generalized method of moments (Marchisio and Fox, 2005; Lebaz et al., 2016) is a widely used mathematical reduction technique for population balance equations. In this case, the reduced model does not preserve full information on spatial profile. Another mathematical possibility to obtain reduced models is to separate fast and slow subsystems. Slow manifold approximation (Christofides and Daoutidis, 1997) requires complicated symbolic operations, which impose difficulties on the automatization of this method. To sum up, widely used methods for nonlinear model reduction require experienced user; automatic application and integration in a simulation tool is a difficult and challenging task, which has hardly been attempted to our knowledge. On the other hand, there are linear model reduction techniques like balanced truncation (Benner et al., 2000; Heinkenschloss et al., 2011), which are applicable to high order systems and can be automatized quite easily. However, the resulting linear reduced models are only

[*] Corresponding author.
E-mail addresses: khlopov@mpi-magdeburg.mpg.de (D. Khlopov), mangold@mpi-magdeburg.mpg.de (M. Mangold).

http://dx.doi.org/10.1016/j.compchemeng.2016.11.004
0098-1354/© 2016 Elsevier Ltd. All rights reserved.

valid locally and not able to capture nonlinear properties of the original system.

In this work proper orthogonal decomposition (POD) (Kunisch and Volkwein, 2002; Park and Cho, 1996; Sirovich, 1987; Antoulas, 2005) is used for the development of an automatic procedure for model reduction. This method has been successfully applied for numerous problems in the fields of fluid dynamics, optimal control, and for population balance systems like crystallizers (Krasnyk and Mangold, 2010; Mangold et al., 2015), and granulators (Mangold, 2012). To put it in other words, the model reduction by POD is a proven approach. Nevertheless, applying model reduction by POD manually to complex engineering models is a challenging and tedious task. The idea of this work is to provide a software environment that performs the model reduction by POD automatically with minimal additional input from the user.

The work is structured as follows. Section 2 discusses the model reduction method. Technical details of the developed software tool for automatic model reduction are described in Section 3. Section 4 shows the developed software tool in action by applying it to two test models: a nonlinear heat conductor and a continuous fluidized bed crystallizer.

2. Mathematical model reduction method

2.1. Reference model representation

Before applying a reduction procedure to the reference model, it has to be transformed into a spatially discretized form by applying the method of lines (Schiesser, 1991). Discretization results in a system of differential algebraic equations, which may be written as

$$B\frac{dx}{dt}(t) = f(x(t)) = Ax(t) + c + g(x(t)), \qquad (1)$$

where $x(t)$ is the discretized state vector, B and A are the system matrices, where B may be singular, c is a constant vector, and $g(x(t))$ is a function that comprises the nonlinearities of the system.

2.2. POD method

In this work the proper orthogonal decomposition method (Kunisch and Volkwein, 2002; Park and Cho, 1996; Sirovich, 1987; Antoulas, 2005) is used for the development of an automatic procedure for the model reduction. The basic idea of this method is to approximate the model solution by a linear combination of time independent basis functions weighted by time dependent coefficients. The basis functions are constructed from numerical simulation results of the detailed reference model. Applying Galerkin's method of weighted residuals produces the reduced model equations. At this point the offline phase of the reduction procedure ends, which can be extremely computationally intensive depending on the complexity of the reference model. But these efforts pay off in the second fast and cheap step, the online phase. In the online phase only a differential algebraic system of low order has to be solved.

As a starting point of the offline phase, the detailed reference model has to be solved numerically. Snapshots for the model states $x(t_1), x(t_2), \ldots$ and for the right-hand sides $f(t_1), f(t_2), \ldots$ are stored in matrices $X = (x(t_1), x(t_2), \ldots)$ and $F = (f(t_1), f(t_2), \ldots)$, correspondingly.

A reduced basis for the snapshots vectors is constructed from the singular value decomposition (SVD) of X with

$$X = U\Sigma V^T, \qquad (2)$$

where U is a unitary matrix containing the left singular vectors or POD modes, which are already ordered by the singular values, V^T is a unitary matrix containing the right singular vectors and Σ is a pseudo-diagonal matrix with the descending singular values as entries. The singular values are a measure for the truncation error and hence determine the order of the reduced model.

Consequently the basis vectors for the orthogonal projection are taken as

$$\Psi^x_i = U_i, \quad i = 1, \ldots, N^x, \qquad (3)$$

where U_i denotes the ith column of U, and N^x is the dimension of the reduced basis and correspondingly the order of the resulting reduced model.

The state vector $x(t)$ is approximated by the following expression:

$$x(t) \approx \Psi^x \phi^x(t), \qquad (4)$$

where $\Psi^x = (\Psi^x_1, \ldots, \Psi^x_{N^x})$, and $\phi^x(t)$ is the coefficient vector of the reduced basis and the state of the reduced model.

In order to obtain equations for $\phi^x(t)$, the approximation for the state vector (4) is inserted into the discretized differential equation (1). To make the projection of the residuals on the reduced basis vanish, Galerkin's method of weighted residuals is applied, which leads to

$$\underbrace{\Psi^{xT} B \Psi^x}_{=:B_{red}} \frac{d\phi^x}{dt}(t) = \underbrace{\Psi^{xT} A \Psi^x}_{=:A_{red}} \phi^x(t) + \underbrace{\Psi^{xT} c}_{=:c_{red}} + \Psi^{xT} g(\Psi^x \phi^x(t)) \qquad (5)$$

The matrices B_{red}, A_{red} and the vector c_{red} from Eq. (5) have to be evaluated only once for a fixed reduced basis, because they do not depend on the reduced state vector $\phi^x(t)$.

2.3. Empirical interpolation

The nonlinear term on the right-hand side of Eq. (5) still depends on the high order state vector of the reference model, bringing additional complexity during the runtime of the reduced model. Clearly, more efficient approaches are needed. There are several methods in literature on how to handle the nonlinear terms in the context of POD model reduction effectively, whose basic idea is to approximate also the nonlinearities by basis vectors constructed from snapshots (Grepl et al., 2007; Nguyen et al., 2008).

In this work the empirical interpolation method (EI) (Grepl et al., 2007) is used. Its algorithm uses specially selected interpolation indices to specify an interpolation-based projection instead of a more costly orthogonal projection. Thus, the nonlinearity is projected onto a subspace spanned by a basis, which approximates the solution space of the nonlinearity. The basis vectors $\Psi^g_i, i = 1, \ldots, N^g$ for the available snapshots $g(t_i) = f(t_i) - (Ax(t_i) + c)$ are constructed by the iterative procedure in Grepl et al. (2007). During runtime of the reduced model, the nonlinearity is approximated as a linear combination of time independent basis functions $\Psi^g = (\Psi^g_1, \ldots, \Psi^g_{N^g})$ weighted by time dependent coefficients $\phi^g(t)$, which follow from the linear equation system

$$\underbrace{\Psi^g_k}_{=:D_{red}} \phi^g(t) = f_k(x(t)) - (\underbrace{A_k \Psi^x}_{=:E_{red}} \phi^x(t) + c_k) \qquad (6)$$

The indices k from Eq. (6) are the output of the EI algorithm described in Grepl et al. (2007) and chosen in such a way that the approximation error is minimized. This is achieved by placing new interpolation points where the residual between the input basis and its approximation by former interpolation points is largest.

In summary, the resulting reduced model consists of the differential equations

$$\underbrace{\Psi^{xT} B \Psi^x}_{=:B_{red}} \frac{d\phi^x}{dt}(t) = \underbrace{\Psi^{xT} A \Psi^x}_{=:A_{red}} \phi^x(t) + \underbrace{\Psi^{xT} c}_{=:c_{red}} + \underbrace{\Psi^{xT} \Psi^g}_{=:G_{red}} \phi^g(t) \qquad (7)$$