Proper Orthogonal Decomposition Model Order Reduction of Nonlinear IC Models

A. Verhoeven[1], M. Striebel[2], J. Rommes[3], and E.J.W. ter Maten[1,3], and T. Bechtold[3]

[1] Eindhoven University of Technology, The Netherlands, Arie.Verhoeven@na-net.ornl.gov
[2] Technische Universität Chemnitz, Germany, Michael.Striebel@nxp.com
[3] NXP Semiconductors, Eindhoven, The Netherlands, Jan.ter.Maten@nxp.com, Joost.Rommes@nxp.com, Tamara.Bechtold@nxp.com

Summary. We demonstrate Model Order Reduction for a nonlinear system of differential-algebraic equations of a diode chain by Proper Orthogonal Decomposition with Adapted Missing Point Estimation. The collected time snapshots also allow for an efficient impression of the sensitivity of objective functions.

1 Introduction

Future simulation for nanoelectronics requires that circuit equations can be coupled to electromagnetics, to semiconductor equations, and to heat transfer. The consequence is that one has to deal with large systems. Model Order Reduction (MOR) is a means to speed up simulation of large systems. Existing MOR techniques mostly apply to linear problems and even then they have to be generalized to become applicable to a resulting system of (Partial) Differential-Algebraic Equations (DAEs, PDAES). To make MOR applicable to industrial applications one has to address nonlinearity and parameterization. Here we consider Proper Orthogonal Decomposition (POD) to reduce the system size. An adaption is presented to also reduce the complexity in evaluating functions and Jacobian matrices.

The problem of reducing nonlinear systems can be described as follows. Given a, possibly large-scale, nonlinear time-invariant dynamical system $\Sigma = (\mathbf{g}, \mathbf{f}, \mathbf{h}, \mathbf{x}, \mathbf{u}, \mathbf{y}, t)$

$$\Sigma = \begin{cases} \frac{d\mathbf{g}(\mathbf{x}(t))}{dt} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) = \mathbf{h}(\mathbf{x}, \mathbf{u}) \end{cases}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^m$, $\mathbf{y}(t) \in \mathbb{R}^p$, $\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \mathbf{g}(\mathbf{x}(t)) \in \mathbb{R}^n$, $\mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)) \in \mathbb{R}^p$, find a reduced model $\widetilde{\Sigma} = (\tilde{\mathbf{g}}, \tilde{\mathbf{f}}, \tilde{\mathbf{h}}, \tilde{\mathbf{x}}, \mathbf{u}, \tilde{\mathbf{y}}, t)$

A.D. Fitt et al. (eds.), *Progress in Industrial Mathematics at ECMI 2008*, Mathematics in Industry 15, DOI 10.1007/978-3-642-12110-4_68,

$$\widetilde{\Sigma} = \begin{cases} \frac{d\tilde{\mathbf{g}}(\tilde{\mathbf{x}}(t))}{dt} = \tilde{\mathbf{f}}(\tilde{\mathbf{x}}(t), \mathbf{u}(t)) \\ \tilde{\mathbf{y}}(t) = \tilde{\mathbf{h}}(\tilde{\mathbf{x}}, \mathbf{u}) \end{cases}$$

where $\tilde{\mathbf{x}}(t) \in \mathbb{R}^r$, $\mathbf{u}(t) \in \mathbb{R}^m$, $\tilde{\mathbf{y}}(t) \in \mathbb{R}^p$, $\tilde{\mathbf{f}}(\tilde{\mathbf{x}}(t), \mathbf{u}(t)), \tilde{\mathbf{g}}(\tilde{\mathbf{x}}(t)) \in \mathbb{R}^r$, $\tilde{\mathbf{h}}(\tilde{\mathbf{x}}(t), \mathbf{u}(t)) \in \mathbb{R}^p$, such that $\tilde{\mathbf{y}}(t)$ can be computed in much less time than $\mathbf{y}(t)$ and the approximation error $\mathbf{y}(t) - \tilde{\mathbf{y}}(t)$ is small.

In the context of circuit simulation the dynamical systems we are dealing with are circuit blocks or subcircuits. Connection to and communication with a block's environment is done via its terminals, i.e. external nodes. Therefore, we can assume that the currents or voltages are always injected linearly into the circuit under consideration. A similar reasoning applies for the determination of the output signal $\mathbf{y}(t)$, which is also assumed to be not explicitly dependent on the input $\mathbf{u}(t)$. Hence, in the remainder of this document, we assume the dynamical systems to be of the form

$$\Sigma = \begin{cases} \frac{d\mathbf{g}(\mathbf{x}(t))}{dt} = \mathbf{f}(\mathbf{x}(t)) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}^T\mathbf{x} \end{cases}$$

where $\mathbf{B} \in \mathbb{R}^{n\times m}$ and $\mathbf{C} \in \mathbb{R}^{n\times p}$.

The two best-known methods for reduction of nonlinear systems are Proper Orthogonal Decomposition (POD), and Trajectory PieceWise-Linear techniques (TPWL) [4,6,7] (and references cited there).

2 Proper Orthogonal Decomposition (POD)

Proper Orthogonal Decomposition extends the Petrov–Galerkin projection based methods that are used for linear systems to nonlinear system. By choosing a suitable $\mathbf{V} \in \mathbb{R}^{n\times r}$ and a test matrix $\mathbf{W} \in \mathbb{R}^{n\times r}$, where $\mathbf{W}$ and $\mathbf{V}$ are biorthonormal, i.e., $\mathbf{W}^T\mathbf{V} = \mathbf{I}_{r\times r}$, $r \leq n$, the reduced system is given by

$$\begin{cases} \mathbf{W}^T \frac{d\mathbf{g}(\mathbf{V}\tilde{\mathbf{x}}(t))}{dt} = \mathbf{W}^T\mathbf{f}(\mathbf{V}\tilde{\mathbf{x}}(t)) + (\mathbf{W}^T\mathbf{B})\mathbf{u}(t) \\ \tilde{\mathbf{y}}(t) = (\mathbf{C}^T\mathbf{V})\tilde{\mathbf{x}} \end{cases}$$

Similar to linear model order reduction, the idea is that $\mathbf{V}$ captures the dominant dynamics, i.e., the states of the original system are approximated well by $\mathbf{V}\tilde{\mathbf{x}} \approx \mathbf{x}$. The test matrix $\mathbf{W}$ is chosen such that the Petrov–Galerkin condition $\mathbf{r} = \frac{d\mathbf{g}(\mathbf{V}\tilde{\mathbf{x}}(t))}{dt} - \mathbf{f}(\tilde{\mathbf{x}}(t)) - \mathbf{B}\mathbf{u}(t) \perp \mathbf{W}$ is met.

POD constructs the matrix $\mathbf{V}$ as follows. A time domain simulation of the complete system is done and snapshots of the states at suitably chosen times t_i are collected in the state matrix $\mathbf{X}$

$$\mathbf{X} = [\mathbf{x}(t_0),\ \mathbf{x}(t_1),\ \mathbf{x}(t_2),\ \cdots \mathbf{x}(t_{N-1})] \in \mathbb{R}^{n\times N},$$

where N is the number of time points t_i. To extract the subspace that represents that dominant dynamics, the singular value decomposition of $\mathbf{X}$ is computed $\mathbf{X} = \mathbf{U}\Sigma\mathbf{T}$ where $\mathbf{U} \in \mathbb{R}^{n\times n}$, $\Sigma = [\mathrm{diag}(\sigma_1, \ldots, \sigma_n)\ 0_{n\times(N-n)}] \in$

$\mathbb{R}^{n\times N}$ (if $N > n$), and $\mathbf{T} \in \mathbb{R}^{N\times N}$. Let the singular values $\sigma_1 \geq \sigma_2 \cdots \sigma_r \gg \sigma_{r+1} > \cdots > \sigma_n$ be ordered in decreasing magnitude. POD chooses the matrix $\mathbf{V}$ to have as its columns the left singular vectors corresponding to the $r \ll n$ largest singular values

$$\mathbf{V} = [\mathbf{u}_1,\ \mathbf{u}_2, \cdots, \mathbf{u}_r] \in \mathbb{R}^{n\times r}.$$

The number k of vectors to choose can depend on a tolerance based criterion like $\sigma_{k+1} < \epsilon$, or on the relative difference between σ_k and σ_{k+1}. The test matrix $\mathbf{W}$ is taken as $\mathbf{W} = \mathbf{V}$, i.e., the residual is orthogonal to the reduced state space.

We stress that the reduction obtained from POD and similar projection based methods is solely in the number of states: r for the reduced systems vs. n for the original system and $r \ll n$. However, the costs for evaluating nonlinear terms such as $\mathbf{W}^T\mathbf{f}(\mathbf{V}\tilde{\mathbf{x}}(t))$ (and associated Jacobian matrices) will be larger than for the original system. Hence with respect to simulation times no reduction may be obtained unless additional measures are taken.

3 Missing Point Estimation/Adapted POD

We will present some results computed with the Missing Point Analysis/Adapted POD approach described in [3–5]. We reflect the basic idea with the case of a simple ODE

$$\frac{d}{dt}\mathbf{x} = \mathbf{f}(\mathbf{x}),$$

of dimension n with nonlinear right hand side $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$. The singular value decomposition $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$ of a matrix $\mathbf{X} \in \mathbb{R}^{n\times N}$ of N snapshots is computed, giving n singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$. The orthogonal matrix $\mathbf{L} = \mathbf{U} \cdot \mathrm{diag}(\sigma_1, \ldots, \sigma_n) \in \mathbb{R}^{n\times n}$ is introduced, with its columns $\mathbf{l}_1, \ldots, \mathbf{l}_n$ spanning the complete space $\mathbb{R}^n$. Hence, one can change to the new basis, i.e., $\mathbf{x} = \mathbf{L}\mathbf{y}$ and apply a Galerkin-like projection to the system

$$\mathbf{L}^T \frac{d}{dt}(\mathbf{L}\mathbf{y}) = \mathbf{L}^T\mathbf{f}(\mathbf{L}\mathbf{y}). \tag{1}$$

Strictly speaking we do not apply Galerkin projection as the columns of L are orthogonal, but not orthonormal.

Classical POD reduction acts on $\mathbf{x} = \mathbf{L}\mathbf{y}$ in the sense that the expansion of $\mathbf{x}$ in the basis $\mathbf{l}_1, \ldots, \mathbf{l}_n$ where $(\mathbf{l}_1, \ldots, \mathbf{l}_n) = \mathbf{L} = (\sigma_1 \cdot \mathbf{v}_1, \ldots, \sigma_n \cdot \mathbf{v}_n)$ with $(\mathbf{v}_1, \ldots, \mathbf{v}_n) = \mathbf{U}$ is truncated with respect to the magnitude of the singular values $\sigma_1, \ldots, \sigma_n$:

$$\begin{aligned}
\mathbf{x} = \mathbf{L}\mathbf{y} &= (\sigma_1\mathbf{v}_1)\cdot y_1 + \cdots + (\sigma_r\mathbf{v}_r)\cdot y_r + (\sigma_{r+1}\mathbf{v}_{r+1})\cdot y_{r+1} + \cdots + (\sigma_n\mathbf{v}_n)\cdot y_n \\
&\approx (\sigma_1\mathbf{v}_1)\cdot y_1 + \cdots + (\sigma_r\mathbf{v}_r)\cdot y_r + 0\cdot y_{r+1} + 0\cdot y_n \\
&= (\mathbf{l}_1, \ldots, \mathbf{l}_r, 0, \ldots, 0)\cdot\mathbf{y} \\
&= (\mathbf{L}\mathbf{P}_r^T\mathbf{P}_r)\cdot\mathbf{y}, \quad \text{with } \mathbf{P}_r = \left(\mathbf{I}_{r\times r}\ \mathbf{0}_{r\times(n-r)}\right) \in \{0,1\}^{r\times n} \\
&= (\mathbf{L}\mathbf{P}_r^T)\cdot(\mathbf{P}_r\mathbf{y}) = (\mathbf{L}\mathbf{P}_r^T)\cdot\mathbf{z}_r \quad \text{with } \mathbf{z}_r = (y_1, \ldots, y_r)^T \in \mathbb{R}^r
\end{aligned}$$

where r usually is chosen in such a way that $\sigma_{r+1} < \text{TOL}$ or $\sigma_{r+1} \ll \sigma_r$.

This procedure can also be interpreted as keeping the r most "dominant" columns of L and neglecting the rest, where a column's norm is taken as a criterion. That means, $\mathbf{L}$ is approximated by

$$\mathbf{L} \approx \mathbf{L}\mathbf{P}_r^T\mathbf{P}_r, \quad \text{with } \mathbf{P}_r \in \{0,1\}^{r\times n}. \tag{2}$$

where $\mathbf{P}_r = \left(\mathbf{I}_{r\times r}\ \mathbf{0}_{r\times(n-r)}\right)$ selects these columns. By construction of $\mathbf{L} = \mathbf{U}\cdot\text{diag}(\sigma_1,\ldots,\sigma_n)$, where $\mathbf{U}^T\mathbf{U} = \mathbf{I}_{n\times n}$, we have $\|\mathbf{v}_i\|_2 = \sigma_i$ for $i = 1,\ldots,n$. In this respect the r most dominant columns are therefore $\mathbf{l}_1,\ldots,\mathbf{l}_r$.

In the adapted POD presented in [4] this perception is carried over to the transposed $\mathbf{L}^T$. That means, one selects, again based on the norms, the $g \in \mathbb{N}$ most dominant columns $\{\tilde{\mathbf{l}}_{\mu_1},\ldots,\tilde{\mathbf{l}}_{\mu_g}\}$ of $\mathbf{L}^T = (\tilde{\mathbf{l}}_1,\ldots,\tilde{\mathbf{l}}_n)$ and neglects the rest:

$$\mathbf{L}^T \approx \mathbf{L}^T\mathbf{P}_g^T\mathbf{P}_g, \quad \text{with } \mathbf{P}_g \in \{0,1\}^{g\times n}. \tag{3}$$

First, these approximations to $\mathbf{L}$ and $\mathbf{L}^T$ from (2) and (3), respectively, are inserted into (1):

$$\mathbf{L}^T\mathbf{P}_g^T\mathbf{P}_g\frac{d}{dt}(\mathbf{L}\mathbf{P}_r^T\mathbf{P}_r\mathbf{y}) = \mathbf{L}^T\mathbf{P}_g^T\mathbf{P}_g\mathbf{f}(\mathbf{L}\mathbf{P}_r^T\mathbf{P}_r\mathbf{y}) \tag{4}$$

From (2) and (3) it follows that

$$\mathbf{L}^T \approx \mathbf{P}_r^T\mathbf{P}_r\mathbf{L}^T\mathbf{P}_g^T\mathbf{P}_g,$$

and multiplying with $\mathbf{P}_r$ (consider $\mathbf{P}_r\mathbf{P}_r^T = \mathbf{I}_{r\times r}$), the system (4) turns into

$$\mathbf{P}_r\mathbf{L}^T\mathbf{P}_g^T\mathbf{P}_g\frac{d}{dt}(\mathbf{L}\mathbf{P}_r^T\mathbf{P}_r\mathbf{y}) = \mathbf{P}_r\mathbf{L}^T\mathbf{P}_g^T\mathbf{P}_g\mathbf{f}(\mathbf{L}\mathbf{P}_r^T\mathbf{P}_r\mathbf{y})$$

As $\mathbf{L}\mathbf{P}_r^T = (\sigma_1\mathbf{v}_1,\ldots,\sigma_r\mathbf{v}_r) = \mathbf{U}_r\Sigma_r$ (for $\mathbf{U}_r = (\mathbf{v}_1,\ldots,\mathbf{v}_r)$, $\Sigma_r = \text{diag}(\sigma_1,\ldots,\sigma_r)$) we get

$$\Sigma_r\mathbf{U}_r^T\mathbf{P}_g^T\frac{d}{dt}[\mathbf{P}_g\mathbf{U}_r\Sigma_r\mathbf{P}_r\mathbf{y}] = \Sigma_r\mathbf{U}_r^T\mathbf{P}_g^T\mathbf{P}_g\mathbf{f}(\mathbf{U}_r\Sigma_r\mathbf{P}_r\mathbf{y}), \quad \mathbf{L}\mathbf{y} = \mathbf{x}.$$

The above equation states a system of dimension r for $\mathbf{y} \in \mathbb{R}^n$. Therefore, we introduce the reduced state vector $\mathbf{y}_r = \Sigma_r\mathbf{P}_r\mathbf{y} \in \mathbb{R}^r$ from which we can approximately reconstruct the coefficients of the full state in the basis spanned by the columns of $\mathbf{L}$ by $\mathbf{y} \approx \mathbf{P}_r^T\Sigma_r^{-1}\mathbf{y}_r$. This in turn lets us approximate the full state in the original basis $\mathbf{x} \approx \mathbf{U}_r\mathbf{y}_r$, because $\mathbf{x} = \mathbf{L}\mathbf{y} \approx \mathbf{L}\mathbf{P}_r^T\Sigma_r^{-1}\mathbf{y}_r = \mathbf{U}_r\Sigma_r\Sigma_r^{-1}\mathbf{y}_r$. This part is consistent with the classical POD.

In addition to the reduction in the state space the adapted POD downsizes $\mathbf{f}(\cdot)$ by considering that the term $\mathbf{P}_g\mathbf{f}(\cdot)$ corresponds to just including the g components $f_{\mu_1}(\cdot),\ldots,f_{\mu_g}(\cdot)$ of $\mathbf{f}(\cdot) = (f_1(\cdot),\ldots,f_r(\cdot))^T$. Hence, it suffices to evaluate the g-dimensional function

$$\bar{\mathbf{f}} : \mathbb{R}^n \to \mathbb{R}^g : \mathbf{x} \mapsto (f_{\mu_1}(\mathbf{x}),\ldots,f_{\mu_g}(\mathbf{x}))^T.$$

After scaling with Σ_r^{-1} the reduced system for the reduced state vector $\mathbf{y}_r \in \mathbb{R}^r$ becomes

$$\mathbf{U}_r^T \mathbf{P}_g^T \frac{d}{dt}[\mathbf{P}_g \mathbf{U}_r \mathbf{y}_r] = \mathbf{U}_r^T \mathbf{P}_g^T \bar{\mathbf{f}}(\mathbf{U}_r \mathbf{y}_r), \quad \mathbf{x} = \mathbf{U}_r \mathbf{y}_r \tag{5}$$

For the general case of having not an ODE (1) but a DAE

$$\frac{d}{dt}\mathbf{g}(\mathbf{x}) = \mathbf{f}(\mathbf{x}) + \mathbf{B}\mathbf{v}$$

to deal with, one gets a reduced problem

$$\mathbf{U}_r^T \mathbf{P}_g^T \frac{d}{dt}\bar{\mathbf{g}}(\mathbf{U}_r \mathbf{y}_r) = \mathbf{U}_r^T \mathbf{P}_g^T \bar{\mathbf{f}}(\mathbf{U}_r \mathbf{y}_r) + \mathbf{U}_r^T \mathbf{B}\mathbf{v}. \tag{6}$$

with $\bar{\mathbf{g}} : \mathbb{R}^n \to \mathbb{R}^g : \mathbf{x} \mapsto (g_{\mu_1}(\mathbf{x}), \ldots, g_{\mu_g}(\mathbf{x}))^T$.

We end this section with the observation that the collected time snapshots for POD also allow for an efficient first impression of the sensitivity of several objective functions (like consumed power) even in the case of many parameters [2].

4 POD Testcase: Diodechain

We consider the diode chain model shown in Fig. 1 (with the parameters I_s, V_T, R, C). Here the diode functionality is modelled by the current function $g(V_a, V_b)$ and the input function by $U_{\text{in}}(10^9 t)$, for $t \leq 70\,\text{ns}$, see [3–5],

$$g(V_a, V_b) = \begin{cases} I_s(e^{\frac{V_a - V_b}{V_T}} - 1) & \text{if } V_a - V_b > 0.5 \\ 0 & \text{otherwise} \end{cases} \quad U_{\text{in}}(\tau) = \begin{cases} 20 & \text{if } \tau \leq 10 \\ 170 - 15\tau & \text{if } 10 < \tau \leq 11 \\ 5 & \text{if } \tau > 11 \end{cases}$$

The state of the diode chain model consists of 302 elements but there is a lot of redundancy. The numerical solution (nodal voltage in each node) on the time interval $[0, 70\,\text{ns}]$ is computed by the Euler Backward method with fixed stepsizes of 0.1 ns. The full system was run in 42.01 s. Classic POD needed 35.51 s. The POD with Adapted MPE (reducing the state space to $r = 30$ and downsizing evaluations to $g = 35$), only required 5.12 s. No visible error can be seen in the approximative results (Fig. 2 (left)).

If the input changes to $7.5\cos(\frac{2\pi t}{60 \cdot 10^{-9}}) + 12.5$ this impression is confirmed (full system 40.22 s, Classic POD even 45.34 s, POD with Adapted POD 6.28 s; Fig. 2 (right)). This makes POD ca five times slower then TPWL, but much more accurate and more robust [3]. If we further increase the amplitude of the cosine to 9.5 POD is not able to properly recover the regions with higher amplitudes (but neither is TPWL) [5].

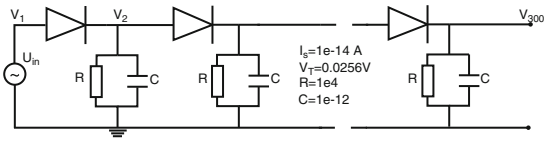

Fig. 1. Schematic of diode chain

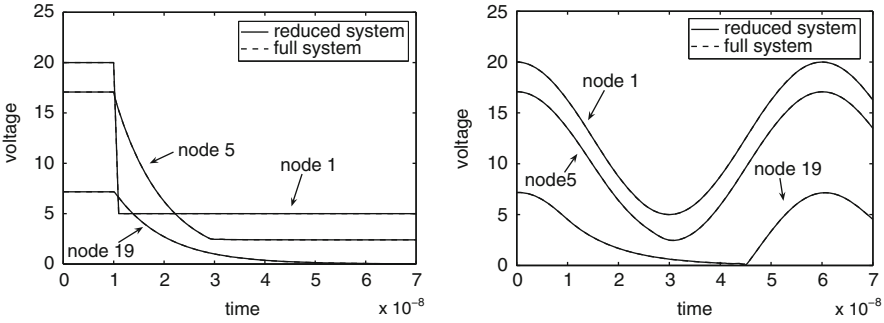

Fig. 2. *Left*: identical input; *Right*: changed input

References

1. Ciuprina, G., Ioan, D. (eds.): Scientific Computing in Electrical Engineering SCEE 2006, Series Mathematics in Industry, vol. 11. Springer, Berlin (2007)
2. Ilievski, Z., Xu, H., Verhoeven, A., ter Maten, E.J.W., Schilders, W.H.A., Mattheij, R.M.M.: Adjoint Transient Sensitivity Analysis in Circuit Simulation. In: [1], pp. 183–189. (2007)
3. Verhoeven, A., Voss, T., Astrid, P., ter Maten, E.J.W., Bechtold, T.: Model order reduction for nonlinear problems in circuit simulation, Proc. ICIAM-2007 in PAMM (Proc. in Appl. Maths. and Mech.), vol. 7(1), pp. 1021603–1021604, 2007/2008, DOI: 10.1002/pamm.200700537 (publ. online Sept. 18 2008)
4. Verhoeven, A., ter Maten, J., Striebel, M., Mattheij, R.: Model order reduction for nonlinear IC models. In: Korytowski, A., Malanowski, K., Mitkowski, W., Szymkat, M. (eds.): System Modeling and Optimization, Series IFIP AICT 312, pp. 476–491. Springer (2009)
5. Verhoeven, A., Striebel, M., ter Maten, E.J.W.: Model Order Reduction for nonlinear IC models with POD. In: Roos, J., Costa, L.R.J. (eds.): Scientific Computing in Electrical Engineering SCEE 2008, Series Mathematics in Industry, vol. 14, Springer (2010)
6. Voss, T., Pulch, R., ter Maten, E.J.W., El Guennouni, A.: Trajectory Piecewise Linear Approach for Nonlinear Differential-Algebraic Equations in Circuit Simulation. In: [1], pp. 167–173. (2007)
7. Voss, T., Verhoeven, A., Bechtold, T., ter Maten, J.: Model Order Reduction for Nonlinear Differential Algebraic Equations in Circuit Simulation. In: Bonilla, L.L., Moscoso, M., Platero, G., Vega, J.M. (eds.): Progress in Industrial Mathematics at ECMI 2006, Series Mathematics in Industry, vol. 12, pp. 518–523. Springer, Berlin (2007). ISBN 978-3-540-71991-5