Discrete Empirical Interpolation in POD Model Order Reduction of Drift-Diffusion Equations in Electrical Networks

Michael Hinze and Martin Kunkel

Abstract We consider model order reduction of integrated circuits with semiconductors modeled by modified nodal analysis and drift-diffusion (DD) equations. The DD-equations are discretized in space using a mixed finite element method. This discretization yields a high dimensional, nonlinear system of differential-algebraic equations. Proper orthogonal decomposition is used to reduce the dimension of this model. Since the computational complexity of the reduced order model through the nonlinearity of the DD equations still depends on the number of variables of the full model we apply the discrete empirical interpolation method to further reduce the computational complexity. We provide numerical comparisons which demonstrate the performance of this approach.

1 Introduction

In this article we investigate model order reduction (MOR) based on proper orthogonal decomposition (POD) for semiconductors in electrical networks using discrete empirical interpolation method (DEIM) to treat the reduction of nonlinear components. Electrical networks can be modeled efficiently by a differential-algebraic equation (DAE) which is obtained from modified nodal analysis. Often semiconductors themselves are modeled by electrical networks. These models are stored in a library and are stamped into the surrounding network in order to create a complete model of the integrated circuit. In [7] POD-based MOR (POD-MOR)

M. Hinze (✉)
Fachbereich Mathematik, Universität Hamburg, Bundesstr. 55, 20146 Hamburg, Germany
e-mail: michael.hinze@uni-hamburg.de

M. Kunkel
Institut für Mathematik, Universität Würzburg, Am Hubland, 97074 Würzburg, Germany
e-mail: martin.kunkel@mathematik.uni-wuerzburg.de

B. Michielsen and J.-R. Poirier (eds.), *Scientific Computing in Electrical Engineering SCEE 2010*, Mathematics in Industry 16, DOI 10.1007/978-3-642-22453-9_45,

is proposed to obtain a reduced surrogate model conserving as much of the drift-diffusion (DD) structure as possible in the reduced order model (ROM). This approach in [6] is extended to parametrized electrical networks using the greedy sampling proposed in [9]. Advantages of the POD approach are the higher accuracy of the model and fewer model parameters. On the other hand, numerical simulations are more expensive. For a comprehensive overview of the DD equations we refer to [2, 8, 11].

This paper is organized as follows. We describe the unreduced model in Sect. 2. In Sect. 3, we present the MOR method based on snapshot POD combined with DEIM. In Sect. 4 we present numerical experiments, and also discuss advantages and shortcomings of our approach.

2 Discretized Coupled Model

Using the notation introduced in [5, 13] the finite element method (FEM) discretization of one semiconductor with domain $\Omega \subset \mathbb{R}^d$ $(d = 1, 2, 3)$ in an electrical network leads to a nonlinear, fully coupled DAE system of the form

$$A_C \frac{d}{dt} q_C(A_C^\top e(t), t) + A_R g(A_R^\top e(t), t) + A_L j_L(t) + A_V j_V(t) + A_S j_S(t) + A_I i_s(t) = 0, \quad (1)$$

$$\frac{d}{dt} \phi_L(j_L(t), t) - A_L^\top e(t) = 0, \quad (2)$$

$$A_V^\top e(t) - v_s(t) = 0, \quad (3)$$

$$q_S(t) - \frac{dg_\psi}{dt}(t) = 0, \quad (4)$$

$$j_S(t) - C_1 J_n(t) - C_2 J_p(t) - C_3 q_S(t) = 0, \quad (5)$$

$$\begin{pmatrix} 0 \\ -M_L \frac{dn}{dt}(t) \\ M_L \frac{dp}{dt}(t) \\ 0 \\ 0 \\ 0 \end{pmatrix} + A_{FEM} \begin{pmatrix} \psi(t) \\ n(t) \\ p(t) \\ g_\psi(t) \\ J_n(t) \\ J_p(t) \end{pmatrix} + F(n(t), p(t), g_\psi(t)) - b(e(t)) = 0, \quad (6)$$

compare Fig. 1, and see [6, 7]. Here, (1)–(3) describe the electrical network with unknown node potentials e, and branch currents j_L of inductive, and j_V of voltage source branches, respectively. Equations (4)–(5) are discretized coupling conditions.

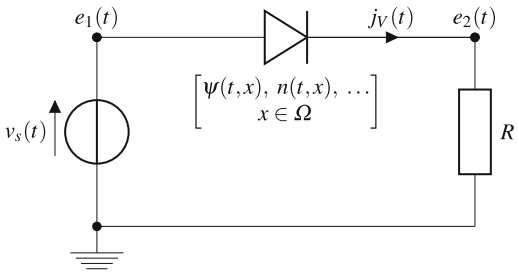

Fig. 1 Basic test circuit with one diode. The network is described by

$$A_V = \begin{pmatrix} 1, & 0 \end{pmatrix}^\top,$$
$$A_S = \begin{pmatrix} -1, & 1 \end{pmatrix}^\top,$$
$$A_R = \begin{pmatrix} 0, & 1 \end{pmatrix}^\top,$$
$$g(A_R^\top e, t) = \frac{1}{R} e_2(t).$$

The vector-valued function ψ contains the weights for the ansatz functions φ_i in the Galerkin ansatz

$$\psi^h(t,x) = \sum_{i=1}^{N} \psi_i(t)\varphi_i(x), \qquad x \in \Omega, \tag{7}$$

for the discretized potential of the semiconductor. Here, h denotes the discretization parameter and N denotes the number of finite elements. The discretized electron and hole concentrations $n^h(t,x)$ and $p^h(t,x)$, the electric field $-g_\psi^h(t,x)$ and the current densities $J_n^h(t,x)$ and $J_p^h(t,x)$ are defined likewise. The incidence matrix $A = [A_R, A_C, A_L, A_V, A_I, A_S]$ represents the network topology and is defined as usual. The matrices A_{FEM} and M_L are large and sparse. The voltage sources v_s and current sources i_s are considered as inputs of the network.

3 Model Order Reduction

We use POD-MOR applied to the DD part (6) to construct a dimension-reduced surrogate model for (1)–(6). For this purpose we run a simulation of the unreduced system and collect l snapshots $\psi^h(t_k,\cdot)$, $n^h(t_k,\cdot)$, $p^h(t_k,\cdot)$, $g_\psi^h(t_k,\cdot)$, $J_n^h(t_k,\cdot)$, $J_p^h(t_k,\cdot)$ at time instances $t_k \in \{t_1,\ldots,t_l\} \subset [0,T]$. The optimal selection of the time instances is not considered here. We use the time instances delivered by the DAE integrator. The snapshot variant of POD introduced in [12] finds a best approximation of the space spanned by the snapshots w.r.t. to the considered scalar product.

Since every component of the state vector $z := (\psi, n, p, g_\psi, J_n, J_p)$ has its own physical meaning we apply POD-MOR to each component separately. Among other

things this approach has the advantage of yielding a block-dense model and the approximation quality of each component is adapted individually.

The time-snapshot POD procedure delivers Galerkin ansatz spaces for ψ, n, p, g_ψ, J_n and J_p and we set $\psi^{POD}(t) := U_\psi \gamma_\psi(t)$, $n^{POD}(t) := U_n \gamma_n(t), \ldots$. The injection matrices $U_\psi \in \mathbb{R}^{N \times s_\psi}$, $U_n \in \mathbb{R}^{N \times s_n}, \ldots$, contain the (time independent) POD basis functions, and the vectors $\gamma_{(\cdot)}$ the corresponding time-variant coefficients. The numbers $s_{(\cdot)}$ denote the respective number of POD basis functions included. Assembling the POD system yields the ROM

$$A_C \frac{d}{dt} q_C(A_C^\top e(t), t) + A_R g(A_R^\top e(t), t) + A_L j_L(t) + A_V j_V(t) + A_S j_S(t) + A_I i_s(t) = 0, \quad (8)$$

$$\frac{d}{dt} \phi_L(j_L(t), t) - A_L^\top e(t) = 0, \quad (9)$$

$$A_V^\top e(t) - v_s(t) = 0, \quad (10)$$

$$q_S(t) - U_{g_\psi} \frac{dg_\psi}{dt}(t) = 0, \quad (11)$$

$$j_S(t) - C_1 U_{J_n} \gamma_{J_n}(t) - C_2 U_{J_p} \gamma_{J_p}(t) - C_3 q_S(t) = 0, \quad (12)$$

$$\begin{pmatrix} 0 \\ -\frac{d\gamma_n}{dt}(t) \\ \frac{d\gamma_p}{dt}(t) \\ 0 \\ 0 \\ 0 \end{pmatrix} + A_{POD} \begin{pmatrix} \gamma_\psi(t) \\ \gamma_n(t) \\ \gamma_p(t) \\ \gamma_{g_\psi}(t) \\ \gamma_{J_n}(t) \\ \gamma_{J_p}(t) \end{pmatrix} + U^\top F(U_n \gamma_n(t), U_p \gamma_p(t), U_{g_\psi} \gamma_{g_\psi}(t)) - U^\top b(e(t)) = 0, \quad (13)$$

with $A_{POD} = U^\top A_{FEM} U$ and $U = \mathrm{diag}(U_\psi, U_n, U_p, U_{g_\psi}, U_{J_n}, U_{J_p})$. All matrix-matrix multiplications are calculated in an offline-phase. The nonlinear function F has to be evaluated online which means that the computational complexity of the ROM still depends on the number of unknowns of the unreduced model. The nonlinearity in (13) is given by

$$U^\top F(U\gamma(t)) = \begin{pmatrix} 0 \\ U_n^\top F_n(U_n \gamma_n(t), U_p \gamma_p(t)) \\ U_p^\top F_p(U_n \gamma_n(t), U_p \gamma_p(t)) \\ 0 \\ U_{J_n}^\top F_{J_n}(U_n \gamma_n(t), U_{g_\psi} \gamma_{g_\psi}(t)) \\ U_{J_p}^\top F_{J_p}(U_n \gamma_p(t), U_{g_\psi} \gamma_{g_\psi}(t)) \end{pmatrix},$$

see e.g. [6]. The subsequent considerations apply for each block component of F. For the sake of presentation we only consider the second block

$$\underbrace{U_n^\top}_{\text{size } s_n \times N} \quad \underbrace{F_n}_{N \text{ evaluations}} \quad \Big(\underbrace{U_n}_{\text{size } N \times s_n} \gamma_n(t), \underbrace{U_p}_{\text{size } N \times s_p} \gamma_p(t) \Big), \tag{14}$$

and its derivative with respect to γ_p,

$$\underbrace{U_n^\top}_{\text{size } s_n \times N} \quad \underbrace{\frac{\partial F_n}{\partial p}(U_n\gamma_n(t), U_p\gamma_p(t))}_{\text{size } N \times N, \text{ sparse}} \underbrace{U_p}_{\text{size } N \times s_p}.$$

Here, the matrices $U_{(\cdot)}$ are dense and the Jacobian of F_n is sparse. The evaluation of (14) is of computational complexity $O(N)$. Furthermore, we need to multiply large dense matrices in the evaluation of the Jacobian. Thus, the POD-MOR may become inefficient.

To overcome this problem, we apply DEIM, proposed in [3], which we now describe briefly. The snapshots $\psi^h(t_k,\cdot)$, $n^h(t_k,\cdot)$, $p^h(t_k,\cdot)$, $g^h_\psi(t_k,\cdot)$, $J^h_n(t_k,\cdot)$, $J^h_p(t_k,\cdot)$ are collected at time instances $t_k \in \{t_1,\dots,t_l\} \subset [0,T]$ as before. Additionally, we collect snapshots $\{F_n(n(t_k), p(t_k))\}$ of the nonlinearity. DEIM approximates the projected function (14) such that

$$U_n^\top F_n(U_n\gamma_n(t), U_p\gamma_p(t)) \approx U_n^\top V_n (P_n^\top V_n)^{-1} P_n^\top F_n(U_n\gamma_n(t), U_p\gamma_p(t)),$$

where $V_n \in \mathbb{R}^{N\times\tau_n}$ contains the first τ_n POD basis functions of the space spanned by the snapshots $\{F_n(n(t_k), p(t_k))\}$ associated with the largest singular values. The selection matrix $P_n = \left(e_{\rho_1},\dots,e_{\rho_{\tau_n}}\right) \in \mathbb{R}^{N\times\tau_n}$ selects the rows of F_n corresponding to the so-called DEIM indices $\rho_1,\dots,\rho_{\tau_n}$ which are chosen such that the growth of a global error bound is limited and $P_n^\top V_n$ is regular, see [3] for details.

The matrix $W_n := U_n^\top V_n (P_n^\top V_n)^{-1} \in \mathbb{R}^{s_n\times\tau_n}$ as well as the whole interpolation method is calculated in an offline phase. In the simulation of the ROM we instead of (14) evaluate:

$$\underbrace{W_n}_{\text{size } s_n \times \tau_n} \quad \underbrace{P_n^\top F_n}_{\tau_n \text{ evaluations}} \quad \Big(\underbrace{U_n}_{\text{size } N \times s_n} \gamma_n(t), \underbrace{U_p}_{\text{size } N \times s_p} \gamma_p(t) \Big), \tag{15}$$

with derivative

$$\underbrace{W_n^\top}_{\text{size } s_n \times \tau_n} \quad \underbrace{\frac{\partial P_n^\top F_n}{\partial p}(U_n\gamma_n(t), U_p\gamma_p(t))}_{\text{size } \tau_n \times N, \text{ sparse}} \underbrace{U_p}_{\text{size } N \times s_p}.$$

In the applied FEM a single functional component of F_n only depends on a small constant number $c \in \mathbb{N}$ components of $U_n\gamma_n(t)$. Thus, the matrix-matrix multiplication in the derivative does not really depend on N since the number of entries per row in the Jacobian is at most c.

But there is still a dependence on N, namely the calculation of $U_n\gamma_n(t)$. To overcome this dependency we identify the required components of the vector $U_n\gamma_n(t)$ for the evaluation of $P_n^\top F_n$. This is done by defining selection matrices $Q_{n,n} \in \mathbb{R}^{c\tau_n \times s_n}$, $Q_{n,p} \in \mathbb{R}^{c\tau_p \times s_p}$ such that

$$P_n^\top F_n(U_n\gamma_n(t), U_p\gamma_p(t)) = \hat{F}_n(Q_{n,n}U_n\gamma_n(t), Q_{n,p}U_p\gamma_p(t)),$$

where $\hat{F}_n$ denotes the functional components of F_n selected by P_n restricted to the arguments selected by $Q_{n,n}$ and $Q_{n,p}$.

Supposed that $\tau_n \approx s_n \ll N$ we obtain a ROM which does not depend on N any more.

4 Numerical Investigation

The discussed method is implemented in C++ based on the FEM library deal.II [1]. The high dimensional DAE is integrated using the DASPK software package [10]. The derivative of the nonlinear functional is hard to compute and thus we calculate the Jacobians by automatic differentiation with the package ADOL-C [14]. The Newton systems which arise from the BDF method are solved with the direct sparse solver SuperLU [4].

A basic test circuit with a single 1-dimensional diode is depicted in Fig. 1. The parameters of the diode are summarized in [6]. The input $v_s(t)$ is chosen to be sinusoidal with amplitude 5 V. In the sequel the frequency of the voltage source will be considered as a model parameter.

We first validate the ROM at a fixed reference frequency of $5 \cdot 10^9$ Hz. Figure 2 shows the development of the relative error between the POD reduced, the POD-

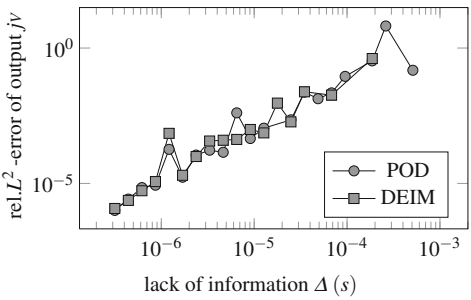

Fig. 2 Relative error between reduced and unreduced problem at the fixed frequency $5 \cdot 10^9$ Hz

DEIM reduced and the unreduced numerical solutions, plotted over the lack of information Δ of the POD basis functions with respect to the space spanned by the snapshots. The figure shows that the approximation quality of the POD-DEIM reduced model is comparable with the more expensive POD reduced model. The number of POD basis functions $s_{(\cdot)}$ for each variable is chosen such that the indicated approximation quality is reached, i.e. $\Delta := \Delta_\psi \simeq \Delta_n \simeq \Delta_p \simeq \Delta_{g_\psi} \simeq \Delta_{J_n} \simeq \Delta_{J_p}$. The numbers $\tau_{(\cdot)}$ of POD-DEIM basis functions are chosen likewise.

In Fig. 3 the simulation times are plotted versus the lack of information Δ. The POD ROM does not reduce the simulation times significantly for the chosen parameters. The reason for this is the dependency on the number of variables of the unreduced system. Here, the unreduced system contains 1000 finite elements which yields 12012 unknowns. The POD-DEIM ROM behaves very well and leads to a reduction in simulation time of about 90% without reducing the accuracy of the ROM. However, we have to report a minor drawback; not all tested ROMs converge for large $\Delta(s) \geq 3 \cdot 10^{-5}$. This is indicated in the figures by missing squares.

In Fig. 4 we plot the corresponding total number of required POD basis functions. It can be seen that with the number of POD basis functions increasing linearly, the lack of information tends to zero exponentially. Furthermore, the number of DEIM interpolation indices behaves in the same way.

In Fig. 5 we investigate the dependence of the ROMs on the number of finite elements N. One sees that the simulation times of the unreduced model depends linearly on N. The POD ROM still depends on N linearly with a smaller constant. The dependence on N of our POD-DEIM implementation is negligible.

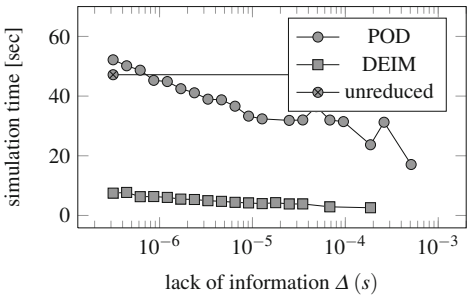

Fig. 3 Time consumption for simulation runs of Fig. 2. The horizontal line indicates the time consumption for the simulation of the original full system

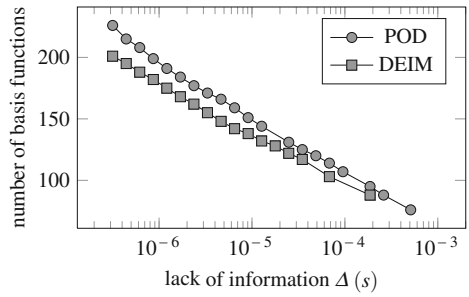

Fig. 4 The number of required POD basis function and DEIM interpolation indices grows only logarithmically with the requested information content

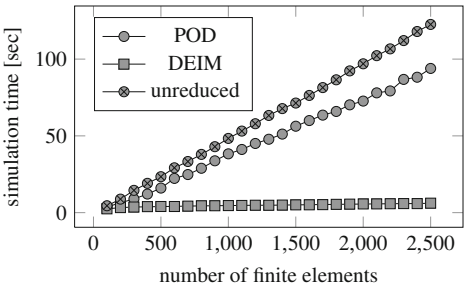

Fig. 5 Computation times of the unreduced and the reduced order models plotted versus the number of finite elements

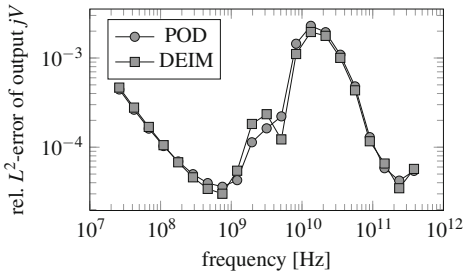

Fig. 6 The reduced models are compared with the unreduced model at various input frequencies

Finally, we in Fig. 6 analyze the behaviour of the models with respect to parameter changes. We consider the frequency of the sinusoidal input voltage as model parameter. The ROMs are created based on snapshots gathered in a full simulation at a frequency of $5 \cdot 10^9$ Hz. We see that the POD model and the POD-DEIM model behave very similarly. The adaptive refinement of the ROM is discussed in [6].

Summarizing all numerical results we conclude that the significantly faster POD-DEIM reduction method yields a ROM with the same qualitative behaviour as the ROM obtained by classical POD-MOR.

Acknowledgements The work reported in this paper was supported by the German Federal Ministry of Education and Research (BMBF), grant no. 03HIPAE5. Responsibility for the contents of this publication rests with the authors.

References

1. Bangerth, W., Hartmann, R., Kanschat, G.: deal.II – a General Purpose Object Oriented Finite Element Library. ACM Trans. Math. Softw. **33**(4) (2007)
2. Brezzi, F., Marini, L., Micheletti, S., Pietra, P., Sacco, R., Wang, S.: Discretization of Semiconductor Device Problems. I. Schilders, W. H. A. et al. (ed.) Handbook of Numerical Analysis, vol. XIII, pp. 317–441, Elsevier/North Holland, Amsterdam (2005)
3. Chaturantabut, S., Sorensen, D.C.: Nonlinear Model Reduction via Discrete Empirical Interpolation. SIAM J. Scientific Comput. **32**(5), 2737–2764 (2010)

4. Demmel, J.W., Eisenstat, S.C., Gilbert, J.R., Li, X.S., Liu, J.W.H.: A Supernodal Approach to Sparse Partial Pivoting. SIAM J. Matrix Anal. Appl. **20**(3), 720–755 (1999)
5. Günther, M., Feldmann, U., ter Maten, J.: Modelling and Discretization of Circuit Problems. Schilders, W.H.A. et al. (ed.) Handbook of Numerical Analysis, vol. 13, pp. 523–629, Elsevier/North Holland, Amsterdam (2005)
6. Hinze, M., Kunkel, M.: Residual Based Sampling in POD Model Order Reduction of Drift-Diffusion Equations in Parametrized Electrical Networks. J. Appl. Math. Mech. **91**(9), 1–14 (2011) URL http://arxiv.org/abs/1003.0551
7. Hinze, M., Kunkel, M., Vierling, M.: POD model order reduction of drift-diffusion equations in electrical networks. In: ter Maten, E. Jan W. (eds.) Lecture Notes in Electrical Engineering, vol. 74, Benner, Peter; Hinze, Michael (2011)
8. Markowich, P.: The Stationary Semiconductor Device Equations. Computational Microelectronics. Springer, New York (1986)
9. Patera, A., Rozza, G.: Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations. Version 1.0. Copyright MIT 2006–2007, to appear in (tentative rubric) MIT Pappalardo Graduate Monographs in Mechanical Engineering (2007)
10. Petzold, L.R.: A Description of DASSL: A Differential/Algebraic System Solver. IMACS Trans. Scientific Comput. **1**, 65–68 (1993)
11. Selberherr, S.: Analysis and Simulation of Semiconductor Devices. Springer, New York (1984)
12. Sirovich, L.: Turbulence and the Dynamics of Coherent Structures I: Coherent Structures. II: Symmetries and Transformations. III: Dynamics and Scaling. Q. Appl. Math. **45**, 561–590 (1987)
13. Tischendorf, C.: Coupled Systems of Differential Algebraic and Partial Differential Equations in Circuit and Device Simulation. Habilitation thesis, Humboldt-University of Berlin (2003)
14. Walther, A., Griewank, A.: A Package for Automatic Differentiation of Algorithms written in C/C++. URL https://projects.coin-or.org/ADOL-C