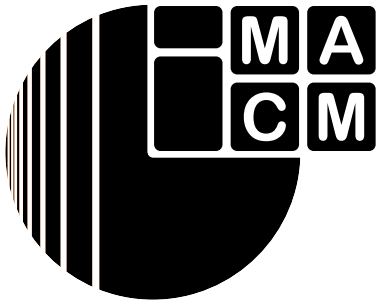

Bergische Universität Wuppertal

Fachbereich Mathematik und Naturwissenschaften

Institute of Mathematical Modelling, Analysis and Computational Mathematics (IMACM)

Preprint BUW-IMACM 14/16

Kai Gausling, Andreas Bartel

Analysis of the Contraction Condition in the Co-Simulation of a Specific Electric Circuit

June 4, 2014

http://www.math.uni-wuppertal.de

Analysis of the Contraction Condition in the Co-Simulation of a Specific Electric Circuit

Kai Gausling and Andreas Bartel

Abstract The convergence for a co-simulation method is commonly based on an error recursion. Usually the contraction condition itself is obtained by some estimations (standard theory). This paper takes a closer look at the coupling structure of co-simulation model for a simple electric circuit. It is shown that with standard theory for our example no contraction could be inferred. However, co-simulation converges. By a detailed analysis, we can prove convergence in this case.

1 Introduction

Co-simulation is an important method for coupled systems in time domain. In particular, if the monolithic description of a dynamic system is not feasible and/or dedicated simulation tools for the subsystems are available, then it is a relevant option. In practice co-simulation is frequently applied to electrical circuits. Seminal approaches in this field were already specified in [6]. Furthermore this simulation methodology is capable of multirate, multimethod, multiorder (and so on). However, convergence can only be achieved by solving multiple times the subsystems. To enhance convergence, the whole simulation time is split into time windows. Co-simulation applied to coupled ordinary differential equations (ODEs) always convergences [4]. This is not the case for coupled differential-algebraic equations (DAEs). There convergence can only be guranteed if a contraction condition is fulfilled, see e.g. [1]. It can be shown that the convergence and stability of co-

Kai Gausling,
Lehrstuhl für Angewandte Mathematik/Numerische Analysis, Bergische Universität Wuppertal, D-42119 Wuppertal, e-mail: gausling@math.uni-wuppertal.de

Andreas Bartel,
Lehrstuhl für Angewandte Mathematik/Numerische Analysis, Bergische Universität Wuppertal, D-42119 Wuppertal e-mail: bartel@math.uni-wuppertal.de

simulation is directly influenced by the order of computation and by the coupling interface, see e.g. [3].

In fact, a co-simulation computes the solutions of the coupled subsystems separately on windows $[T_n, T_n + H]$. We follow the Gauss-Seidel approach. Let (k) denote the current iteration, also old iterates $(k-1)$ are involved. Such a co-simulation scheme can be encode by splitting functions $\mathbf{F}, \mathbf{G}$:

$$\begin{aligned} \dot{\mathbf{y}} &= \mathbf{f}(\mathbf{y}, \mathbf{z}) \qquad \leftrightarrow \quad \dot{\tilde{\mathbf{y}}} = \mathbf{F}\left(\tilde{\mathbf{y}}^{(k)}, \tilde{\mathbf{z}}^{(k)}, \tilde{\mathbf{y}}^{(k-1)}, \tilde{\mathbf{z}}^{(k-1)}\right) \\ 0 &= \mathbf{g}(\mathbf{y}, \mathbf{z}) \qquad\quad\; 0 = \mathbf{G}\left(\tilde{\mathbf{y}}^{(k)}, \tilde{\mathbf{z}}^{(k)}, \tilde{\mathbf{y}}^{(k-1)}, \tilde{\mathbf{z}}^{(k-1)}\right) \end{aligned}$$

Then the contraction condition reads:

$$\alpha := \|\mathbf{G}^{-1}_{z^{(k)}} \mathbf{G}_{z^{(k-1)}}\|_2 < 1, \tag{1}$$

where $\mathbf{G}_{\mathbf{z}^{(k)}}$, $\mathbf{G}_{\mathbf{z}^{(k-1)}}$ denote partials Jacobians of $\mathbf{G}$, see e.g. [1], [2].

Our paper is outlined as follows: We consider a linear test system, where the standard contraction condition (1) is not fulfilled. In a numerical treatment, we observe convergence. Then convergence for this test case is proven by an exact fine structure analysis. Finally, we discuss the connection of both types analysis.

2 Circuit Modeling and Test Circuit

Classically, a mathematical model for an electric network can be obtained via modified nodal analysis, see e.g. [5]. This gives a DAE:

$$\mathbf{E}\dot{\mathbf{x}} + \mathbf{A}\mathbf{x} = \mathbf{f}(t),$$

where $\mathbf{E}$ contains the dynamic components, $\mathbf{A}$ static components and $\mathbf{f}$ time depended sources. The unknowns $\mathbf{x}$ are the node voltages and some branch currents.

We investigate the simple RL circuits depicted in Fig. 1. Modified nodal analysis yields an index-1 DAE. By applying the strategy of source coupling (see e.g. [2]),

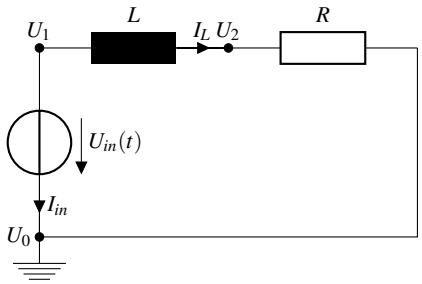

Fig. 1 RL circuit applied by supply voltage $U_{in}(t)$ (reference model).

we can model this circuit as two coupled networks as given in Fig. 2. This example serves as our test case for co-simulation. Notice that the monolithic circuit (Fig. 1) is almost the same as the subsystem 2 (Fig. 2). This is due to the fact that we aimed at an example as simple as possible. This makes our model rather academic, however it shows the divergence between analysis and application of co-simulation, which we want to highlight. Now, the two subsystems for our co-simulation read:

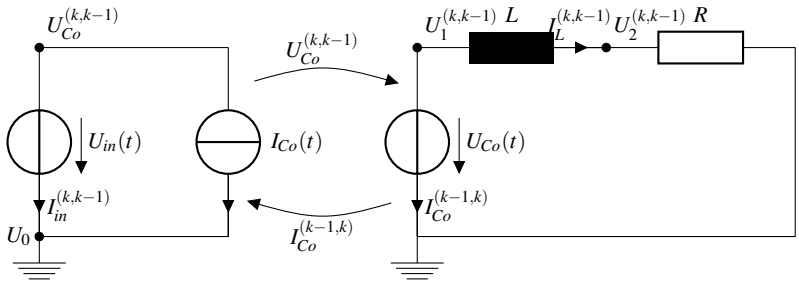

Fig. 2 Decoupled RL network using source-coupling in a co-simulation of Gauss-Seidel type. The first/second notation index denotes the old and new differential and algebraic variables for subsystem 1/ subsystem 2 first.

$$\begin{aligned}
&\textbf{Subsystem 1:} \quad 0 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} U_{Co} \\ I_{in} \end{pmatrix} - \begin{pmatrix} -I_{Co}(t) \\ -U_{in}(t) \end{pmatrix}, \\
&\textbf{Subsystem 2:} \quad 0 = \begin{pmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 0&0&0&0 \\ 0&0&0&L \end{pmatrix} \begin{pmatrix} \dot{U}_1 \\ \dot{U}_2 \\ \dot{I}_{Co} \\ \dot{I}_L \end{pmatrix} + \begin{pmatrix} 0&0&1&1 \\ 0&G&0&-1 \\ -1&0&0&0 \\ -1&1&0&0 \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ I_{Co} \\ I_L \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ -U_{Co}(t) \\ 0 \end{pmatrix},
\end{aligned} \tag{2}$$

with inductance L, conductance $G = 1/R$, given voltage source $U_{in} = U_{in}(t)$, unknown node potentials U_1, U_2, U_{Co} and unknown currents I_{in}, I_L, I_{Co}. U_{Co} and I_{Co} are additional variables needed for the source coupling. The application of a Gauss-Seidel type of co-simulation demands to choose a system, which is computed first.

3 Standard Abstract Co-simulation Analysis

Next we use standard theory [1, 2] to analyze the coupled system (2). To this end, we generalize the system (2) to the following semi-explicit form:

$$0 = \mathbf{g}_1(\mathbf{z}_1, \mathbf{z}_2), \qquad \dot{\mathbf{y}}_2 = \mathbf{f}_2(\mathbf{y}_2, \mathbf{z}_2), \quad 0 = \mathbf{g}_2(\mathbf{z}_1, \mathbf{y}_2, \mathbf{z}_2), \tag{3}$$

where subsystem 1 (subindex '1') is merely a system of linear equations and subsystem 2 is a DAE. The variables of the subsystems are

$$\mathbf{z}_1 := \left[U_{Co},\, I_{in}\right]^T, \quad \mathbf{y}_2 := I_L, \quad \mathbf{z}_2 := \left[U_1,\, U_2,\, I_{Co}\right]^T.$$

Since $\partial \mathbf{g}_i / \partial \mathbf{z}_i$ are not singular in (2), the subsystems and the overall system are index-1. Thus $\mathbf{y_i}$ defines the differential and $\mathbf{z_i}$ the algebraic components. Notice, the semi-explicit form (3) encodes, which type of variables occur in the submodels.

Now, we start co-simulation with the subsystem 1 first and obtain the corresponding splitting functions:

$$\begin{aligned} \mathbf{F}(\mathbf{y}^{(k)}, \mathbf{y}^{(k-1)}, \mathbf{z}^{(k)}, \mathbf{z}^{(k-1)}) &:= \left[\mathbf{f}_2(0, 0, \mathbf{y}_2^{(k)}, \mathbf{z}_2^{(k)})\right], \\ \mathbf{G}(\mathbf{y}^{(k)}, \mathbf{y}^{(k-1)}, \mathbf{z}^{(k)}, \mathbf{z}^{(k-1)}) &:= \begin{bmatrix} \mathbf{g}_1(0, \mathbf{z}_1^{(k)}, 0, \mathbf{z}_2^{(k-1)}) \\ \mathbf{g}_2(0, \mathbf{z}_1^{(k)}, \mathbf{y}_2^{(k)}, \mathbf{z}_2^{(k)}) \end{bmatrix}. \end{aligned} \tag{4}$$

Notice the old algebraic iterate $\mathbf{z}_2^{(k-1)}$ ($I_{Co}^{(k-1)}$) enters algebraic equations. The reversed computational order gives us the splitting functions (subsystem 2 first):

$$\begin{aligned} \mathbf{F}(\mathbf{y}^{(k)}, \mathbf{y}^{(k-1)}, \mathbf{z}^{(k)}, \mathbf{z}^{(k-1)}) &:= \left[\mathbf{f}_2(0, 0, \mathbf{y}_2^{(k)}, \mathbf{z}_2^{(k)})\right], \\ \mathbf{G}(\mathbf{y}^{(k)}, \mathbf{y}^{(k-1)}, \mathbf{z}^{(k)}, \mathbf{z}^{(k-1)}) &:= \begin{bmatrix} \mathbf{g}_1(0, \mathbf{z}_1^{(k)}, 0, \mathbf{z}_2^{(k)}) \\ \mathbf{g}_2(0, \mathbf{z}_1^{(k-1)}, \mathbf{y}_2^{(k)}, \mathbf{z}_2^{(k)}) \end{bmatrix}. \end{aligned} \tag{5}$$

Also here depends an algebraic constraint on old algebraic iterates (subsystem 2 depends on $\mathbf{z}_1^{(k-1)}$, i.e., $U_{Co}^{(k-1)}$). Thus the contraction factor α does not vanish for both pairs of splitting functions (4) and (5). Consequently, stability and contraction cannot be guaranty without previously estimated contraction factor α. Therefore we calculate the matrices $\mathbf{G}_{z^{(k)}}^{-1}, \mathbf{G}_{z^{(k-1)}}$ needed in (1). Splitting the Jacobian of $\mathbf{G}(\mathbf{y}^{(k)}, \mathbf{y}^{(k-1)}, \mathbf{z}^{(k)}, \mathbf{z}^{(k-1)})$ into parts of $\mathbf{G}_{\mathbf{y}^{(k)}}, \mathbf{G}_{\mathbf{y}^{(k-1)}}, \mathbf{G}_{\mathbf{z}^{(k)}}$ and $\mathbf{G}_{\mathbf{z}^{(k-1)}}$, we obtain:

$$\textbf{Subsystem 1 first: } \mathbf{G}_{\mathbf{z}^{(k)}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & G & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \end{pmatrix} \Rightarrow \mathbf{G}_{\mathbf{z}^{(k)}}^{-1} = \begin{pmatrix} 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & R & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}, \mathbf{G}_{\mathbf{z}^{(k-1)}} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$\textbf{Subsystem 2 first: } \mathbf{G}_{\mathbf{z}^{(k)}} = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & G & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \end{pmatrix} \Rightarrow \mathbf{G}_{\mathbf{z}^{(k)}}^{-1} = \begin{pmatrix} 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & R & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \end{pmatrix}, \mathbf{G}_{\mathbf{z}^{(k-1)}} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

Thus we obtain for the contraction conditions for both splitting schemes:

$$\begin{aligned} \textbf{Subsystem 1 first}: &\quad \|\mathbf{G}_{\mathbf{z}^{(k)}}^{-1} \mathbf{G}_{\mathbf{z}^{(k-1)}}\|_2 = \| \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \end{pmatrix}^T \|_2 = 1, \\ \textbf{Subsystem 2 first}: &\quad \|\mathbf{G}_{\mathbf{z}^{(k)}}^{-1} \mathbf{G}_{\mathbf{z}^{(k-1)}}\|_2 = \| \begin{pmatrix} -1 & 0 & 0 & 0 & 0 \end{pmatrix}^T \|_2 = 1, \end{aligned} \tag{6}$$

i.e., stability and contraction cannot be inferred for our co-simulation model directly by using standard theory. Notice that standard theory gives only a rough inside into the co-simulation.

4 Numerical Results

Now we analyze the above RL circuit numerically using MATLAB®[1]. For this purpose, we employ the following parameters: resistance $R = 10\text{k}\Omega$, inductance $L = 1\text{mH}$, and capacitance $C = 1\text{nF}$. The circuit is operated by a supply voltage $U_{in}(t) = 1\text{V} \cdot \cos(\omega t)$ with an angular frequency $\omega = 2\pi \cdot 5 \cdot 10^3$ Hz.

To investigate contraction and convergency, a co-simulation is studied in one time window $[t_0, t_0+H]$ with $t_0 = 0.4$ ms and time window size $H = 10^{-4}$ s. The accuracy of the solutions on the n-th time window after k iterations $\tilde{\mathbf{X}}^{(k)}(t)$ is measured by comparing with a reference solution $\mathbf{X}_m(t)$ computed by a monolithic simulation: $\Delta_n^{(k)}(t) = \mathbf{X}_m(t) - \tilde{\mathbf{X}}_c^{(k)}(t)$, $\delta_n^{(k)} := \|\Delta_n^{(k)}\|_2$. For both splitting schemes (4) and (5), a constant extrapolation of the initial value is employed for the initial guess $\tilde{\mathbf{X}}^{(0)}(t)$ on time window H is used. This is the most common choice for an initial guess.

Fig. 3 shows convergence and contraction for both splitting schemes (4) and (5). Thus we have convergence even so the estimate (6) does not indicate this behavior. Additionally, we observe two different convergency orders. For subsystem 1 first,

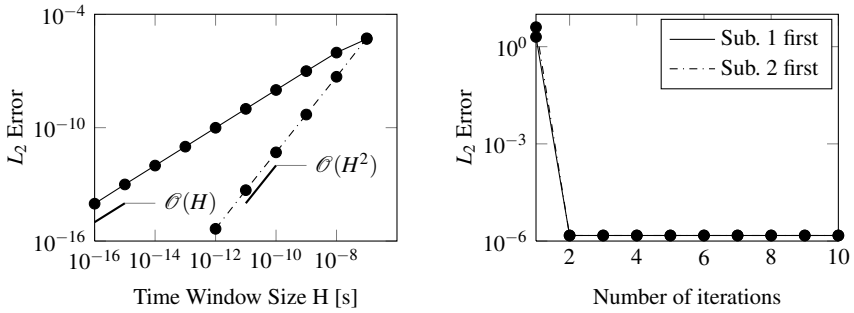

Fig. 3 Convergence and contraction of co-simulation applied to the test circuit in Fig. 2. Solid lines indicate subsystem 1 first. Left: L^2 error versus window size H for one iteration and one window. Right: L^2 error versus the total number of iterations.

we get order $\mathcal{O}(H)$, whereas for subsystem 2 first $\mathcal{O}(H^2)$ is achieved. This can be explained as follows: Constant extrapolation produces an error of $\mathcal{O}(H)$. For subsystem 1 first, the coupling parameter I_{Co} is constantly extrapolated. Since system 1 is just an algebraic equation, their is no improvement during time integration. For subsystem 2 first U_{Co} is constantly extrapolated. This parameter is coupled to the algebraic unknown U_1. However, subsystem 2 has a dynamic element, which is defined by the coupling current. This current is improved during time integration.

[1] Version: MATLAB R2013a, http://www.mathworks.de.

5 Exact Fine Structure Error Propagation

For our test circuit Fig. 2, we aim at calculating a recursion matrix $\mathbf{K}_e$ explicitly for all unknowns in order to verify the above numerical results. To this end, $\Delta_{\mathbf{X}}^{(k)} X_i := X_i^{(k)}(t) - \tilde{X}_i^{(k)}(t)$ measures the difference of of two waveforms on the n-th time window after k iterations. For simplicity of notation the index n is skipped.

We derive the fine structure recursion for our test circuit where the Gauss-Seidel iteration begins with subsystem 1, see (4). For the algebraic variables we find from (2) following the relations to old and new iterates by taking differences

$$\begin{aligned} &\Delta_{\mathbf{z_1}}^{(k)} I_{in} = -\Delta_{\mathbf{z_1}}^{(k-1)} I_{Co}, \quad \Delta_{\mathbf{z_1}}^{(k)} U_{Co} = \Delta U_{in} = 0, \\ &\Delta_{\mathbf{z_2}}^{(k)} I_{Co} = -\Delta_{\mathbf{y_2}}^{(k)} I_L, \qquad \Delta_{\mathbf{z_2}}^{(k)} U_1 = \Delta_{\mathbf{z_1}}^{(k)} U_{Co} = 0, \qquad \Delta_{\mathbf{z_2}}^{(k)} U_2 = \frac{1}{G} \Delta_{\mathbf{y_2}}^{(k)} I_L. \end{aligned} \tag{7}$$

Notice that $U_{Co}^{(k)} = U_{in}(t)$ means that there is no error in the coupling variable $U_{Co}^{(k)}$. From the differential equation for I_L, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\Delta_{\mathbf{y_2}}^{(k)} I_L\right) = \frac{\Delta_{\mathbf{z_2}}^{(k)} U_1 - \Delta_{\mathbf{z_2}}^{(k)} U_2}{L} = \frac{1}{G \cdot L} \Delta_{\mathbf{y_2}}^{(k)} I_L$$

and thus we find for any $t \in [T_n, T_n + H]$

$$|\Delta_{\mathbf{y_2}}^{(k)} I_L(t)| = |\Delta_{\mathbf{y_2}}^{(k)} I_L(t_n)| \cdot e^{(t-t_n)/(G \cdot L)} = |\Delta_{\mathbf{y_2}}^{(k-1)} I_L(t_n)| \cdot e^{(t-t_n)/(G \cdot L)}. \tag{8}$$

Putting (7) and (8) together and using absolute values, we finally find the exact error propagation (for subsystem 1 first):

$$\begin{bmatrix} |\Delta_{\mathbf{y_2}}^{(k)} I_L| \\ |\Delta_{\mathbf{z_1}}^{(k)} I_{in}| \\ |\Delta_{\mathbf{z_1}}^{(k)} U_{Co}| \\ |\Delta_{\mathbf{z_2}}^{(k)} I_{Co}| \\ |\Delta_{\mathbf{z_2}}^{(k)} U_1| \\ |\Delta_{\mathbf{z_2}}^{(k)} U_2| \end{bmatrix} = \underbrace{\begin{bmatrix} 0&0&0&0&0&0 \\ 0&0&0&1&0&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \end{bmatrix}}_{=\mathbf{K}_e} \begin{bmatrix} |\Delta_{\mathbf{y_2}}^{(k-1)} I_L| \\ |\Delta_{\mathbf{z_1}}^{(k-1)} I_{in}| \\ |\Delta_{\mathbf{z_1}}^{(k-1)} U_{Co}| \\ |\Delta_{\mathbf{z_2}}^{(k-1)} I_{Co}| \\ |\Delta_{\mathbf{z_2}}^{(k-1)} U_1| \\ |\Delta_{\mathbf{z_2}}^{(k-1)} U_2| \end{bmatrix} + e^{(t-t_n)/(G \cdot L)} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ \frac{1}{G} \end{bmatrix} |\Delta_{\mathbf{y_2}}^{(k-1)} I_L(t_n)|. \tag{9}$$

Now, the spectral radius of the recursion matrix is zero $\rho(\mathbf{K}_e) = 0$, since all eigenvalues are zero. Hence, $\mathbf{K}_e$ satisfies the contraction condition, i.e., $\rho(\mathbf{K}_e) < 1$, for splitting scheme (4). An analogous computation verifies contraction for the reversed order of computation. Thus this analysis agrees with our numerical observation of convergence.

Clearly, the relation to the standard theory is the lumping of differential and algebraic components in the error recursion (9). Applying the maximum norm, we

obtain the estimate

$$\begin{bmatrix} |\Delta^{(k)}\mathbf{y}| \\ |\Delta^{(k)}\mathbf{z}| \end{bmatrix} \leq \mathbf{K} \begin{bmatrix} |\Delta^{(k-1)}\mathbf{y}| \\ |\Delta^{(k-1)}\mathbf{z}| \end{bmatrix} + \gamma := \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} |\Delta^{(k-1)}\mathbf{y}| \\ |\Delta^{(k-1)}\mathbf{z}| \end{bmatrix} + \begin{bmatrix} C \\ C \end{bmatrix} |\Delta^{(k-1)}\mathbf{y}(t_n)|,$$

with $C = (1 + \frac{1}{G})e^{(t-t_n)/(G \cdot L)}$ and $\rho(\mathbf{K}) = 1$. Thus without fine structure analysis, the contraction disappears from the estimate even for our simple test circuit.

6 Conclusions

We have shown that standard co-simulation theory may not always detects convergence. This holds apply already for a simple electrical circuits, which we have investigated. Therefore we analyzed our model by express the exact error propagation (fine structure analysis) and proved stability and thus contraction for our example. In fact convergence holds for both orders of computation.

Clearly, the information about stability and contraction disappeared during lumping, which we have demonstrated for our example. It is a future aim to investigate stability and contraction derived directly from the network structure and thus to generalize convergence results from standard co-simulation theory.

Acknowledgements This work is supported by the German Federal Ministry of Education and Research (BMBF) in the research project SIMUROM project (grant number 05M13PXB). In addition, we acknowledge the support from the project nanoCOPS, Nanoelectronic COupled Problems Solutions (FP7-ICT-2013-11/619166), http://www.fp7-nanoCOPS.eu/.

References

1. Arnold, M., Günther, M.: Preconditioned Dynamic Iteration for Coupled Differential-Algebraic Systems. BIT, vol. 41, pp. 1–25. (2001)
2. Bartel, A., Brunk, M., Günther, M. and Schöps, S.: Dynamic Iteration for Coupled Problems of Electric Circuits and Distributed Devices. SIAM J. Sci. Comput., vol. 35, No. 2, pp. B315–B335 (2013).
3. Bartel, A., Brunk, M., Schöps, S.: On the convergence rate of dynamic iteration for coupled problems with multiple subsystems. J. Comput. Appl. Math., vol. 262, pp. 14–24 (2014)
4. Burrage, K.: Parallel Methods for Systems of Ordinary Differential Equations. Clarendon Press, Oxford, 1995.
5. Feldmann, U. and Günther, M.: CAD-based electric-circuit modeling in industry I: mathematical structure and index of network equations. Surv. Math. Ind. 8:2, pp. 97–129 (1999).
6. Lelarasmee, E., Ruehli, A. E., Sangiovanni-Vincentelli, A.: The Waveform Relaxation Method for Time Domain Analysis of Large Scale Integrated Circuits. IEEE Computer-Aided Design of Integrated Circuits and Systems, vol. 1, pp. 131 - 145 (1982).