# Chapter 7
# Simple Circuits

This chapter is devoted to present numerical simulation results obtained with the SICONOS platform, on several simple circuits: the first circuit has been built to show that conventional analog simulators fail to converge; the other circuits are classical diode-bridge wave rectifiers, and the last one is a circuit that exhibits a sliding mode. In this chapter and in Chap. 8, five simulation software packages were used:

SICONOS: the platform developed at INRIA Grenoble Rhône-Alpes (France) dealing with nonsmooth dynamical systems with dedicated time integrators and algorithms to solve sets of equations and inequalities (for instance LCP: linear complementarity problems).

NGSPICE: an open-source version of the original SPICE3F5 software developed by Berkeley university. Even if this version may differ from existing commercial ones, it shares with them a common set of models and the solving algorithms belong also to the same class that deals with regular functions.

SMASH: a commercial version of SPICE developed by Dolphin Integration (see http://www.dolphin.fr).

ELDO: a commercial version of SPICE with Newton-Raphson and OSR (one step relaxation) algorithms developed by Mentor Graphics (http://www.mentor.com).

PLECS: a SIMULINK/MATLAB toolbox dedicated to the simulation of power electronics circuits (see http://www.plexim.com). The models and algorithms come from the hybrid approach. In our work the freely available demonstration version of PLECS has been used.

## 7.1 Maffezzoni's Example

This section is devoted to the modeling and the simulation of the circuit in Fig. 7.1. In Maffezzoni et al. (2006) it is shown that Newton-Raphson based methods fail to converge on such a circuit, with the switch model as in (1.46). The diode model is the equivalent resistor model in Fig. 1.2(d). On the contrary the OSNSP solver correctly behaves on the same model, as demonstrated next.

V. Acary et al., *Nonsmooth Modeling and Simulation for Switched Circuits,*
Lecture Notes in Electrical Engineering 69,
DOI 10.1007/978-90-481-9681-4_7, 

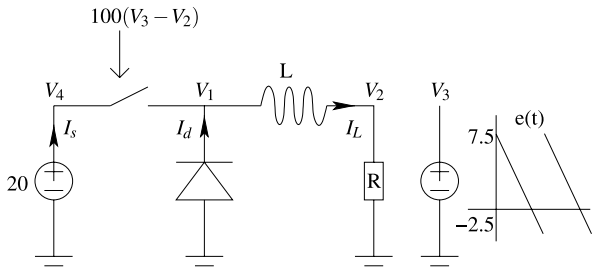



**Fig. 7.1** A simple switched circuit

### *7.1.1 The Dynamical Model*

The dynamics of the circuit in Fig. 7.1 is obtained using the algorithm of automatic circuit equation formulation of Chap. 6. In a first step, the vector of unknown variables is built, in a second step, the dynamical system is written, and in a last step, the nonsmooth laws are added. Applying the automatic equations generation algorithm leads to the following 9-dimensional unknown (dynamic and algebraic unknown variables) vector: $X = (V_1\ V_2\ V_3\ V_4\ I_L\ I_{03}\ I_{04}\ I_s\ I_d)^T$ in the system (5.1) or $x = (I_L)$ and $z = (V_1\ V_2\ V_3\ V_4\ I_{03}\ I_{04}\ I_s\ I_d)^T$ in the system (5.14), where the potentials and the currents are depicted in Fig. 7.1. Building the dynamical equations consists in writing the Kirchhoff current laws at each node, the constitutive equation of the smooth branch, and the nonsmooth law of the other branches. The two nonsmooth devices are the diode and the switch. It yields the following system, that fits within the general framework in (3.32). For the semi-explicit DAE, we obtain:

$$\begin{cases} L\frac{dI_L}{dt}(t) = V_1(t) - V_2(t), \\ I_d(t) + I_s(t) - I_L(t) = 0, \quad I_L(t) - \frac{V_2(t)}{R} = 0, \\ I_{03}(t) = 0, \qquad I_{04}(t) - I_s(t) = 0, \\ V_4(t) = 20, \qquad V_3 = e(t). \end{cases} \tag{7.1}$$

For the input/output relations of the nonsmooth components, we get:

$$\begin{cases} V_1(t) = \frac{1}{2}(\tau_1(t) - 1)R_{\mathsf{off}} I_d(t) - \frac{1}{2}(\tau_1(t) + 1)R_{\mathsf{on}} I_d(t), \\ 2(V_4(t) - V_1(t)) = [(1 + \tau_2(t))R_{\mathsf{off}} + (1 - \tau_2(t))R_{\mathsf{on}}]I_s(t). \end{cases} \tag{7.2}$$

Finally, the inclusion rule is written as:

$$\begin{cases} V_1(t) \in -\mathbb{N}_{[-1,1]}(\tau_1(t)) \\ 100(V_3(t) - V_2(t)) \in -\mathbb{N}_{[-1,1]}(\tau_2(t)). \end{cases} \tag{7.3}$$

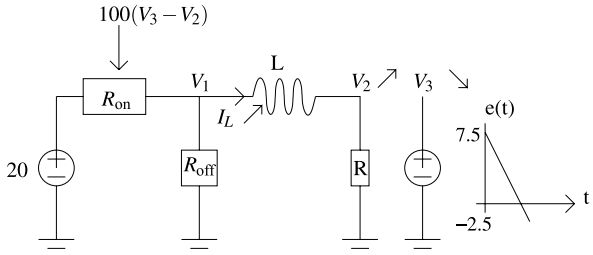



**Fig. 7.2** Equivalent linear circuit

On this example, the fully implicit ($\theta = 1$) Moreau's time-stepping scheme reads as:

$$\begin{cases} L(I_{L,k+1} - I_L) = h(V_{1,k+1} - V_{2,k+1}), \\ I_{d,k+1} + I_{s,k+1} - I_{L,k+1} = 0, \quad I_{L,k+1} - \frac{1}{R} V_{2,k+1} = 0, \\ I_{03,k+1} = 0, \qquad I_{04,k+1} - I_{s,k+1} = 0, \\ V_{4,k+1} = 20, \qquad V_{3,k+1} = e(t_{k+1}), \\ 2V_{1,k+1} = (\tau_{1,k+1} - 1) R_{\text{off}} I_{d,k+1} - (\tau_{1,k+1} + 1) R_{\text{on}} I_{d,k+1}, \\ 2(V_{4,k+1} - V_{1,k+1}) = [(1 + \tau_{2,k+1}) R_{\text{off}} + (1 - \tau_{2,k+1}) R_{\text{on}}] I_{s,k+1}, \\ V_{1,k+1} \in -\mathbb{N}_{[-1,1]}(\tau_{1,k+1}), \\ 100(V_{3,k+1} - V_{2,k+1}) \in -\mathbb{N}_{[-1,1]}(\tau_{2,k+1}). \end{cases} \tag{7.4}$$

## *7.1.2 Simulation Results: Failure of the Newton-Raphson Algorithm*

The simulation consists of two phases: a linear behavior followed by a change in the state of the switch.

### 7.1.2.1 A Linear Behavior

The time step has been fixed to 0.1 µs and the initial state is $\{0, 7.5, 0\}$. While the value of $V_3 - V_2$ is positive, the Newton-Raphson algorithm converges in one iteration. It is a linear circuit shown in Fig. 7.2. During this period, $I_L$ and $V_2$ are increasing and $V_3$ is decreasing. When $V_2$ becomes equals to $V_3$, the switch will change its state.

### 7.1.2.2 The Switch Change of State

We denote $N$ the integer such that the switch will change its state on $[t_N, t_{N+1}]$. Figure 7.3 depicts the initial linearized circuit used by the Newton-Raphson iterations to compute the state at $t_{N+1}$. Setting $V_3$ to the value $e(t_{N+1})$, that is 1.20 V, leads to change the switch's state. Figure 7.4 shows the equivalent circuit after the first

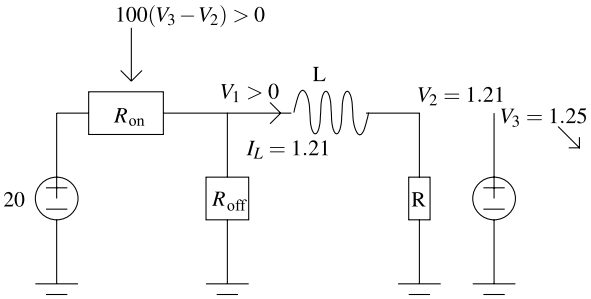



**Fig. 7.3** Circuit state at $t = t_N$

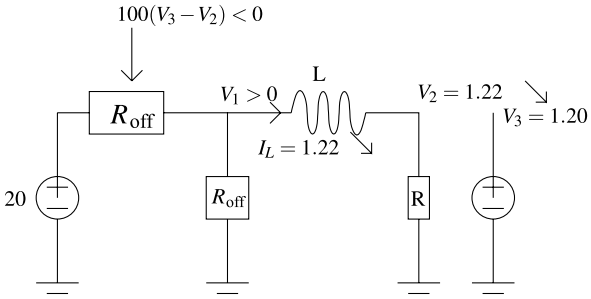



**Fig. 7.4** Equivalent linear model, first step

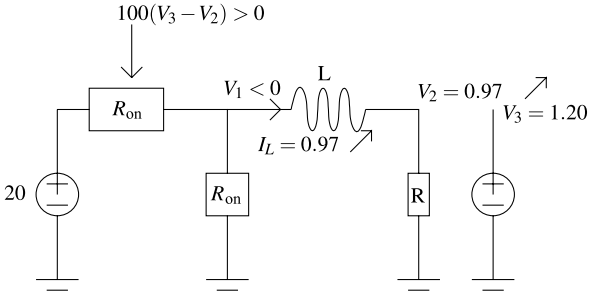



**Fig. 7.5** Equivalent linear model, second step

Newton-Raphson iteration. We note that the switch is now OFF, due to the negative value of $V_2 - V_3$. The second Newton-Raphson iteration causes the decreasing of $V_2$ resulting in changing the states of the diode and of the switch. Figure 7.5 points out the new state of the circuit. The third iteration results in a new setting of both the switch and diode components depicted in Fig. 7.6. On this example, the linearization performed at each Newton-Raphson iteration leads to an oscillation between two incorrect states and never converges to the correct one. The Newton-Raphson iterations enter into an infinite loop without converging.

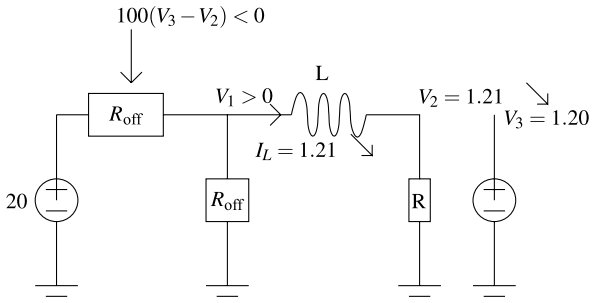



**Fig. 7.6** Equivalent linear model, third step

#### 7.1.2.3 The Newton-Raphson Iterations at $t = t_N$

The next table summarizes the oscillation between two incorrect states:

| | $k=0$ | $k=1$ | $k=2$ | $k=3$ | $k=4$ | ... | Solution |
|---|---|---|---|---|---|---|---|
| $S$ | *ON* | *OFF* | *ON* | *OFF* | *ON* | ... | *OFF* |
| $D$ | *OFF* | *OFF* | *ON* | *OFF* | *ON* | ... | *ON* |

(7.5)

### *7.1.3 Numerical Results with* SICONOS

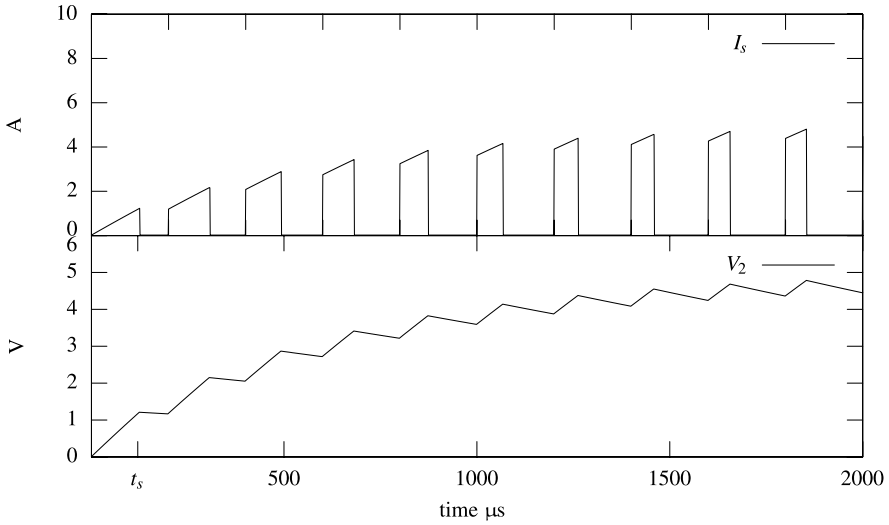
The time step has been fixed to 0.1 µs, the values of the parameters are $R = 1\ \Omega$, $R_{\text{on}} = 0.001\ \Omega$, $R_{\text{off}} = 1000\ \Omega$, $L = 2.10^{-4}$ H and the initial condition is $I_L(0) = 0$ A. Figure 7.7(a) depicts the current evolution through the inductor $L$. Using the NSDS approach the OSNSP solver converges and computes the correct state. For such a simple system, any OSNSP solver gives a correct solution. We have used indifferently PATH and a SEMISMOOTH Newton method.

*Remark 7.1* In Maffezzoni et al. (2006) an event-driven numerical method is proposed to solve the non convergence issue. However it is reliable only if the switching times can be precisely estimated, a shortcoming not encountered with the NSDS and the Moreau's time-stepping method.

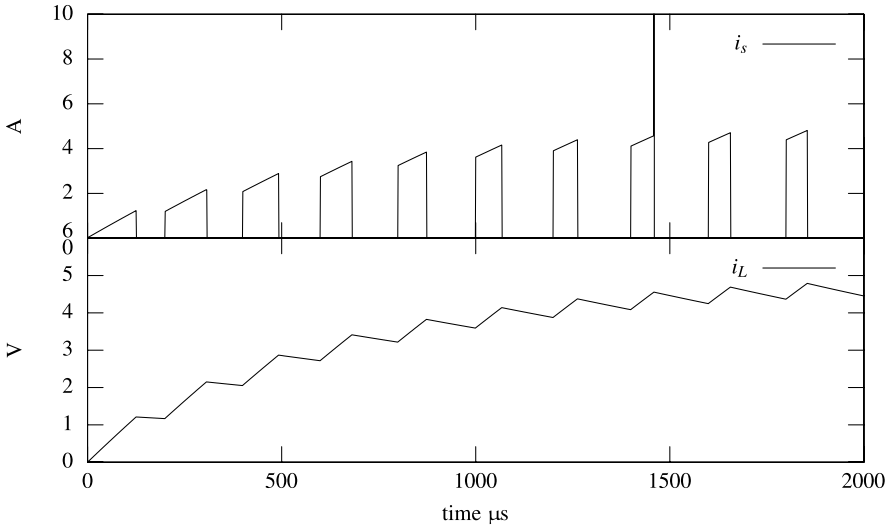
### *7.1.4 Numerical Results with* ELDO

ELDO does not provide any nonsmooth switch model. But it furnishes the 'VSWITCH' one described in (7.6), where $R_S$ is the controlled resistor value of the switch, and $V_c$ the voltage control yielding to the model:

$$R_S(t) = \begin{cases} R_{\text{on}} & \text{if } V_c(t) \geqslant V_{\text{on}}, \\ R_{\text{off}} & \text{if } V_c(t) \leqslant V_{\text{off}}, \\ (V_c(t)(R_{\text{off}} - R_{\text{on}}) + R_{\text{on}}\, V_{\text{off}} & \\ \quad - R_{\text{off}}\, V_{\text{on}})/(V_{\text{off}} - V_{\text{on}}) & \text{otherwise.} \end{cases} \tag{7.6}$$




(a) SICONOS simulation




(b) ELDO simulation

**Fig. 7.7** Switched circuit simulations

Setting $V_{\text{off}}$ to 0 and choosing a small value for $V_{\text{on}}$ lead to a model close to (1.46) for the chosen parameters. Simulations have been done using different sets of parameters. It is noteworthy that the behavior of ELDO depends on these values. For example, using a backward scheme Euler with the time step fixed to 0.1 µs and $V_{\text{on}} = 10^{-4}$ V, $V_{\text{off}} = 0$ V, $R_{\text{off}} = 1000\ \Omega$, $R_{\text{on}} = 0.001\ \Omega$ causes trouble during the ELDO simulation: '`Newton no-convergence`' messages appear. Figure 7.7(b)

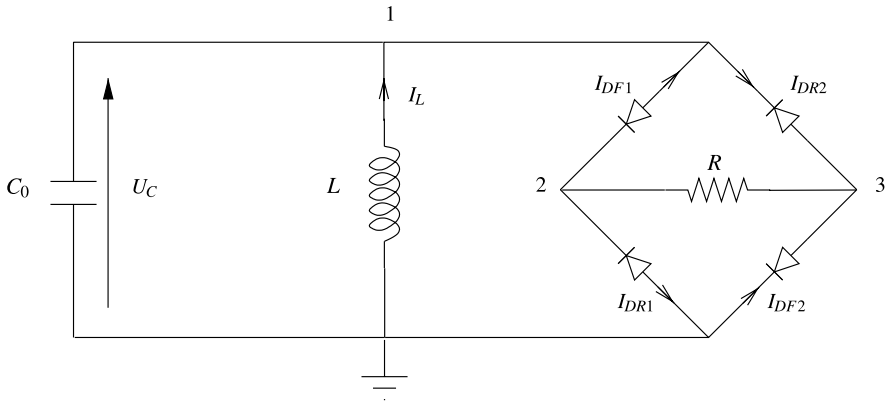



**Fig. 7.8** A 4-diode bridge wave rectifier

shows the ELDO simulation. The values are very close to the SICONOS simulation, except for the steps corresponding to the '`no-convergence`' messages. In this case, the resulting current value is absurd.

This academic example demonstrates that standard analog tools (SPICE-like simulators) can fail to simulate a switched circuit.

## 7.2 A First Diode-Bridge Wave Rectifier

The chosen example is a four-diode bridge wave rectifier as shown in Fig. 7.8. In this sample, an LC oscillator initialized with a given voltage across the capacitor and a null current through the inductor provides the energy to a load resistance through a full-wave rectifier consisting of a 4-ideal-diode bridge. Both waves of the oscillating voltage across the LC are provided to the resistor with current flowing always in the same direction. The energy is dissipated in the resistor resulting in a damped oscillation. This section presents the modeling and the simulation of this circuit using the SICONOS platform and the automatic circuit equation formulation presented in Chap. 6.

### *7.2.1 Dynamical Equations*

The automatic circuit equation formulation leads to the system (7.7), (7.9) and (7.8). The vector of unknown variables is $(U_C, I_L, V_1, V_2, V_3, I_{DF1}, I_{DF2}, I_{DR1}, I_{DR2})^T \in \mathbb{R}^9$. The potentials $V_1$, $V_2$, $V_3$ are the potentials at the points indicated on the figure. We obtain:

$$\begin{cases} -C_0\dot{U}_C(t) + I_L(t) - I_{DR2}(t) + I_{DF1}(t) = 0, \\ L\dot{I}_L(t) + V_1(t) = 0, \\ \frac{V_3(t)-V_2(t)}{R} - I_{DR1}(t) - I_{DF1}(t) = 0, \\ \frac{V_2(t)-V_3(t)}{R} + I_{DR2}(t) + I_{DF2}(t) = 0, \\ U_C(t) - V_1(t) = 0, \end{cases} \tag{7.7}$$

$$\begin{cases} 0 \leqslant \lambda_1(t) \perp V_2(t) - V_1(t) \geqslant 0, \\ 0 \leqslant \lambda_2(t) \perp -V_3(t) \geqslant 0, \\ 0 \leqslant \lambda_3(t) \perp V_2(t) \geqslant 0, \\ 0 \leqslant \lambda_4(t) \perp V_1(t) - V_3(t) \geqslant 0, \end{cases} \tag{7.8}$$

with:

$$\lambda_1 = -I_{DF1} \qquad \lambda_2 = -I_{DF2} \qquad \lambda_3 = -I_{DR1} \qquad \lambda_4 = -I_{DR2}. \tag{7.9}$$

One may identify this dynamics with the canonical form in (2.54), where $x$ may be chosen as the above vector of unknown variables. One has:

$$E = \begin{pmatrix} -C_0 & 0 & 0 & \cdots & 0 \\ 0 & L & 0 & \cdots & 0 \\ 0 & \cdots & & & 0 \\ & & \vdots & & \\ 0 & \cdots & & & 0 \end{pmatrix} \in \mathbb{R}^{9\times 9}, \qquad M = I_4,$$

$$C = \begin{pmatrix} 0 & 0 & -1 & 1 & 0 & \cdots & & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & \cdots & 0 \end{pmatrix} \in \mathbb{R}^{4\times 9}.$$

Here $I_4$ is the $4 \times 4$ identity matrix.

### 7.2.2 Simulation Results

Figure 7.9 shows a simulation with SICONOS using the following numerical values: $L = 10^{-2}$ H, $C = 10^{-6}$ H, $R = 10^3\ \Omega$, $V_1(0) = 10$ V. The initial time is zero and the total simulation time is $5 \times 10^{-3}$ s with a step of $10^{-6}$ s. The nonsmooth problem is written as a Mixed Linear Complementarity Problem (MLCP). It has been solved using indifferently PATH and SEMISMOOTH methods. Obviously an enumerative solver is also convenient for a problem of such a size. The comparison is made with the SPICE simulator SMASH. In this case, the system is equivalent to an ODE with Lipschitz right-hand-side. The simulation with standard SPICE simulator together with low order schemes (Backward Euler) still works.

It is possible to modify the time-stepping algorithm described in Chap. 5 so that the time-step is adapted according to the local error. A practical error estimation is based on halved time-steps. Figure 7.10 presents the result of the step control mechanism using $10^{-3}$ for the relative and absolute tolerance. It is noteworthy that the time step is not decreased to pass through the diodes switching.

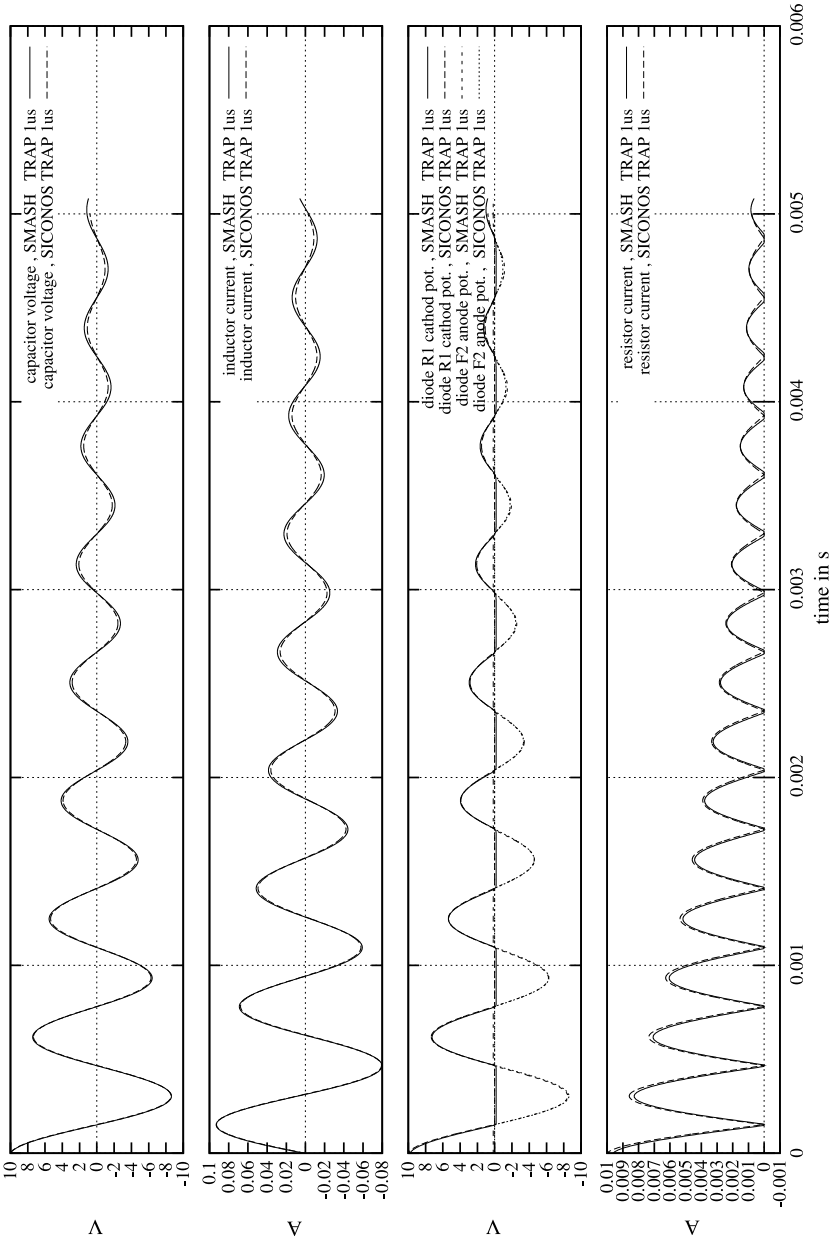



**Fig. 7.9** Bridge wave rectifier simulation results

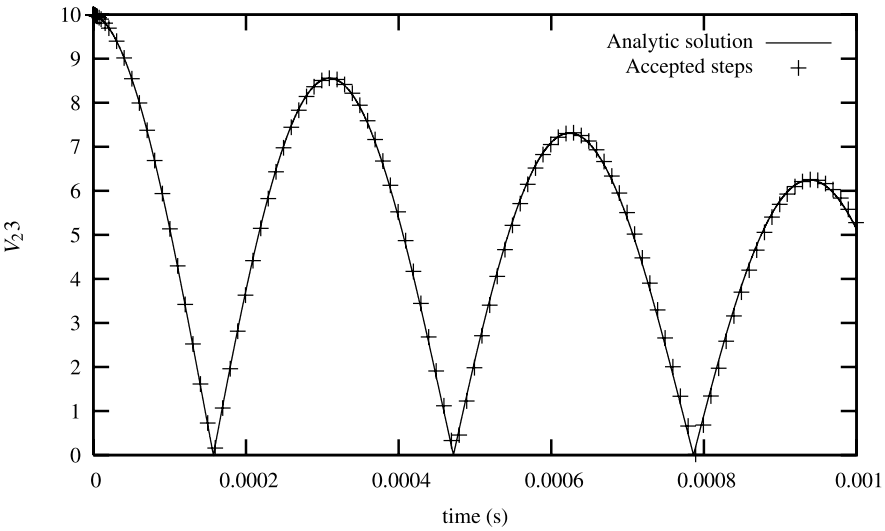



**Fig. 7.10** Step size control using a $10^{-3}$ tolerance. 193 steps +95 rejected

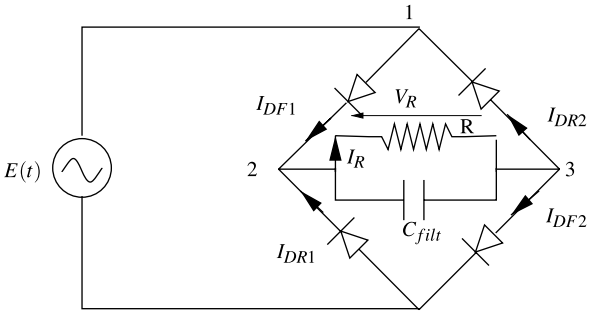



**Fig. 7.11** Filtered full wave rectifier

## 7.3 A Second Diode-Bridge Wave Rectifier

A little bit more complex example was simulated: a sinusoidal voltage supply providing energy to a resistor through a 4-diode bridge full-wave rectifier filtered with a capacitor (see Fig. 7.11). The automatic circuit equation formulation leads to the system (7.10) with the complementarity constraints in (7.9) and (7.8). The vector of unknown variables is $(U_C, V_1, V_2, V_3, I_E, I_{DF1}, I_{DF2}, I_{DR1}, I_{DR2})^T \in \mathbb{R}^9$. The potentials $V_1$, $V_2$, $V_3$ are the potentials at the points indicated in the figure. We obtain:

$$\begin{cases} -C\dot{U}_C(t) + \frac{V_2(t)-V_3(t)}{R} + I_{DR1}(t) + I_{DF1}(t) = 0, \\ -I_E(t) + I_{DR1}(t) - I_{DF2}(t) = 0, \\ I_{DF2}(t) - I_{DR1}(t) + I_{DR2}(t) - I_{DF1}(t) = 0, \\ V_1(t) = e(t), \\ U_C(t) = V_3(t) - V_2(t). \end{cases} \tag{7.10}$$

One may once again identify the dynamics of this circuit with the MLCS dynamics in (2.54), choosing for $x$ the above vector of unknown variables. Figures 7.12, 7.13 and 7.14 show what happens with the SPICE algorithms when the time step is forced to a "high value" (here 10 µs): the SPICE simulator seems to converge but the results are erroneous while the nonsmooth approach provides accurate results. The SPICE package that has been used for these simulations is SMASH and the time-integration scheme is the trapezoidal rule. The results are taken from Denoyelle and Acary (2006).

Figures 7.15 and 7.16 show a comparison between SMASH results with a time step of 0.1 µs and SICONOS results with respectively time steps of 2 µs and 1 µs. The 2 µs results are already very close to SMASH ones. At 1 µs the differences are almost unnoticeable, whereas a factor of 10 is gained on the time step.

These results suggest that with a small number of stiff components in a circuit, the convergence of the Newton-Raphson algorithm is already impaired, even if several tricks were added in the SPICE software to help it. When the integration time period becomes too large, some diodes may be completely blocked at a time step and completely passing at the next time step. The SPICE algorithms are not designed to handle such a case: they need to step a sufficient number of times to cover properly all the switching period which is very short here.
On the contrary, the nonsmooth approach is able to compute a consistent solution with relatively far time steps, assuming that it exists and it is unique, which is true here.

Similar circuits like parallel resonant converters can be modeled and simulated in a similar way by SICONOS,[1] showing the wide range of applicability of the NSDS method in this field of electrical engineering.

## 7.4 The Ćuk Converter

In this section, we are interested in a special type of DC/DC power converter: the Ćuk converter (Middlebrook and Ćuk 1976). The circuit is described in Fig. 7.17. The converter is supplied by a constant voltage supply $E = 10$ V and loads a resistor $R = 50\ \Omega$. The switch is modeled by a linear MOS transistor described in Sect. 4.7.5 with two hyperplanes. The parameters of the nMOS model are $V_T = 5$ V and $K = 10$. The diode threshold voltage is 0.2 V. The capacitance are $C_1 = C_2 = 10$ µF and the inductances are $L_1 = L_2 = 250$ µH. There is no feedback on the regulation of the switch. The voltage $V_G$ at the gate of nMOS model is given a periodic door

[1] See http://siconos.gforge.inria.fr/Examples/EMPowerConverter.html.

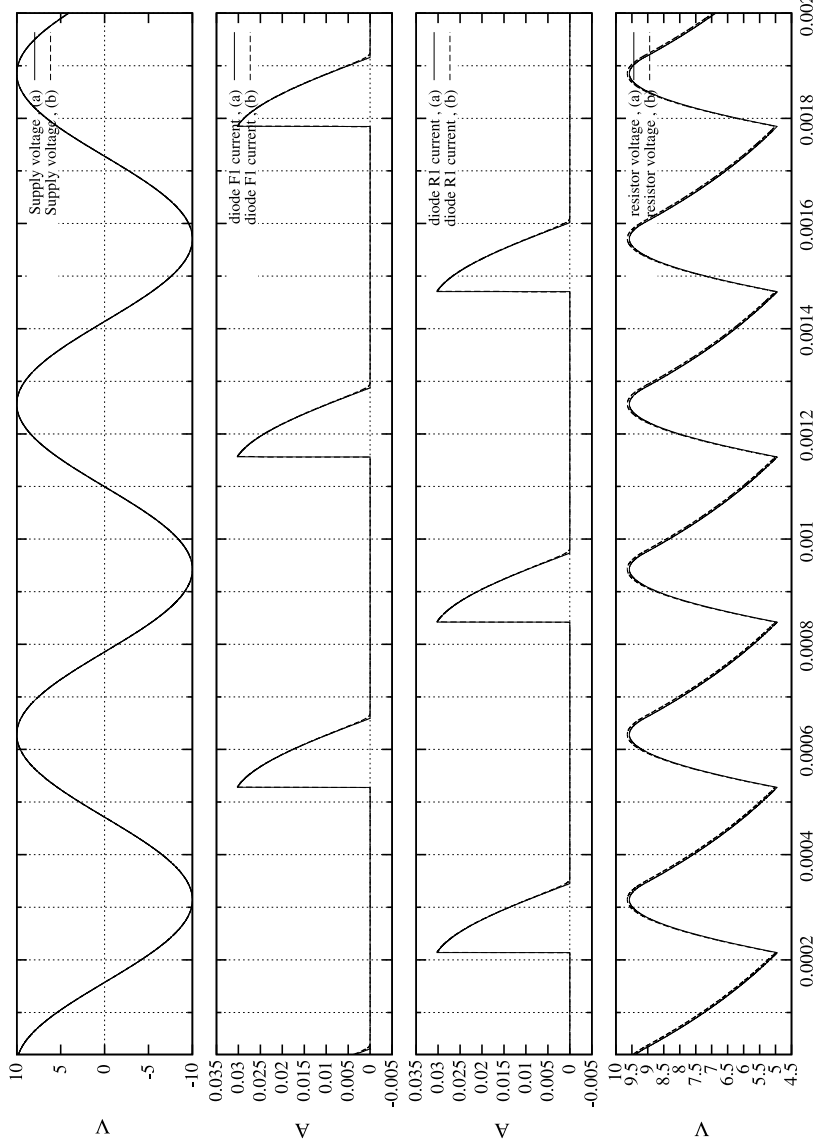



**Fig. 7.12** SICONOS (**a**) and SMASH (**b**) simulations of the diode-bridge circuit, 0.1 µs time step

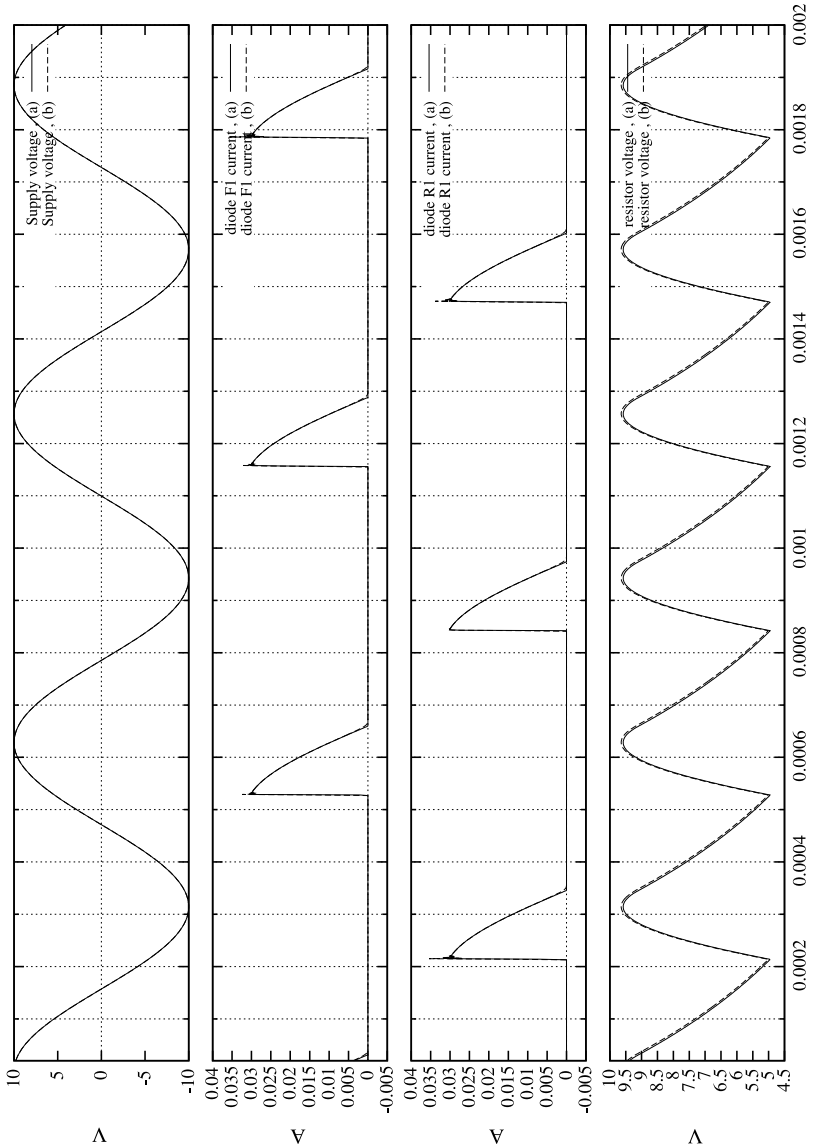



**Fig. 7.13** SICONOS (**a**) and SMASH (**b**) simulations of the diode-bridge circuit, 1 µs time step

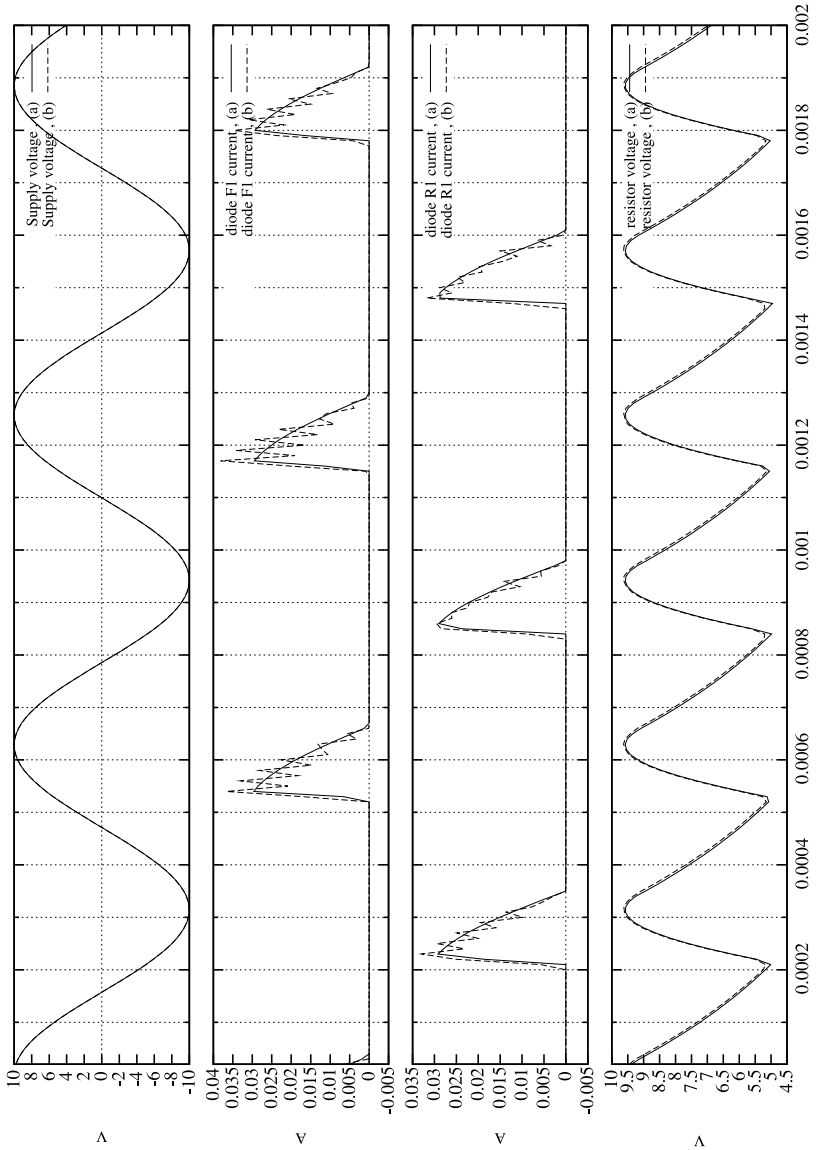



**Fig. 7.14** SMASH (**b**) and SICONOS (**a**) simulations of the diode-bridge circuit, 10 µs time step

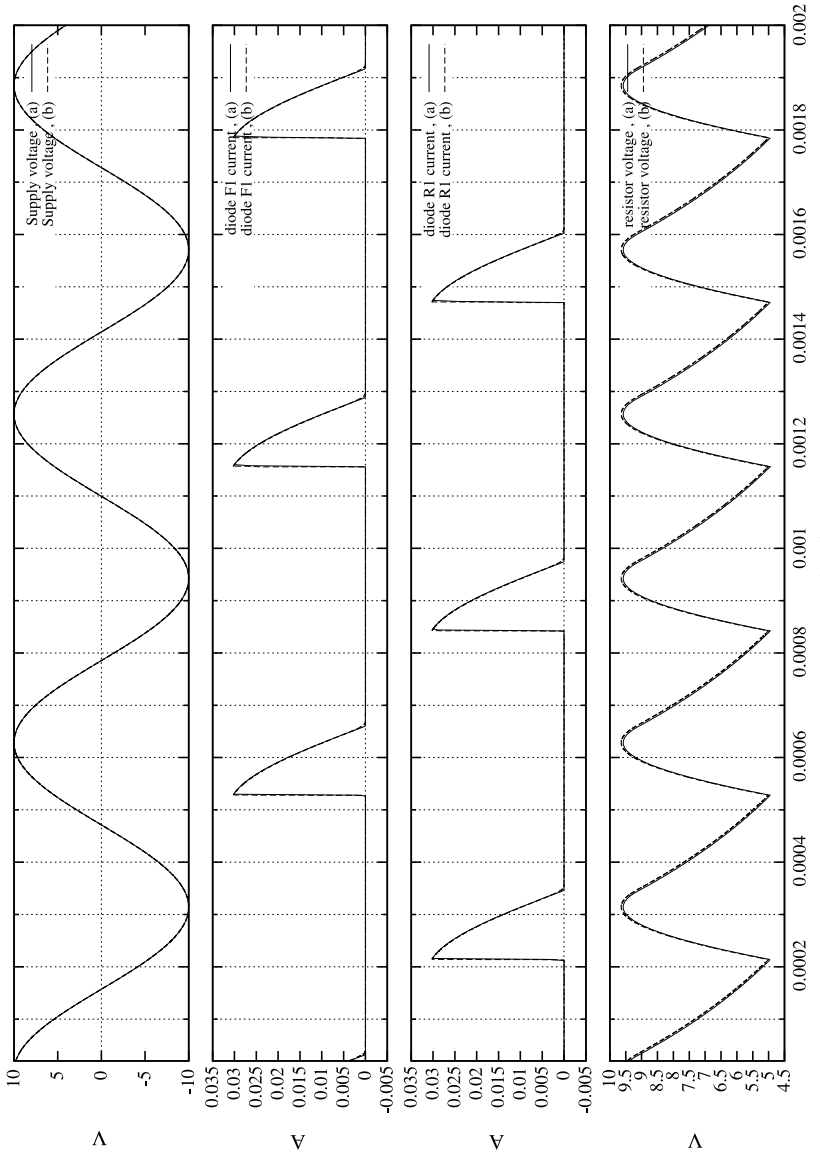



**Fig. 7.15** Simulation results of the diode-bridge circuit. (**a**) SICONOS with $h = 2$ µs (**b**) SMASH with $h = 0.1$ µs

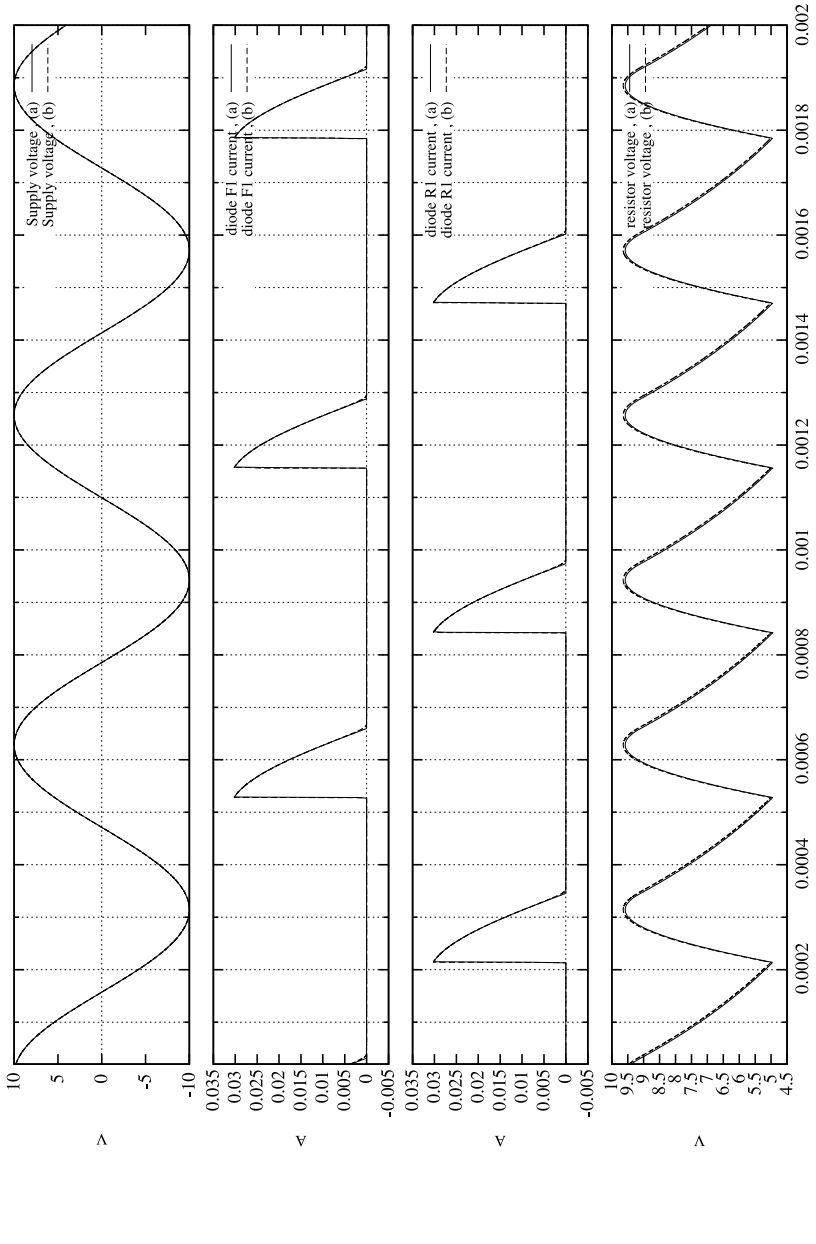



**Fig. 7.16** Simulation results of the diode-bridge circuit. **(a)** SICONOS with $h = 1$ µs **(b)** SMASH with $h = 0.1$ µs

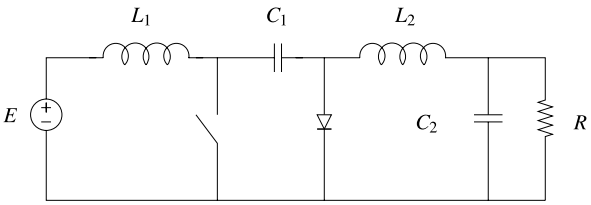



**Fig. 7.17** The Ćuk converter

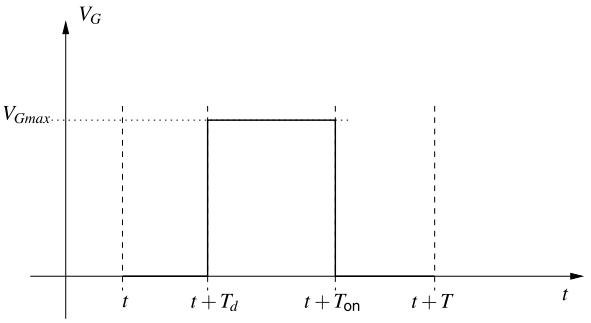



**Fig. 7.18** Regulation function of the switch $V_G(t)$

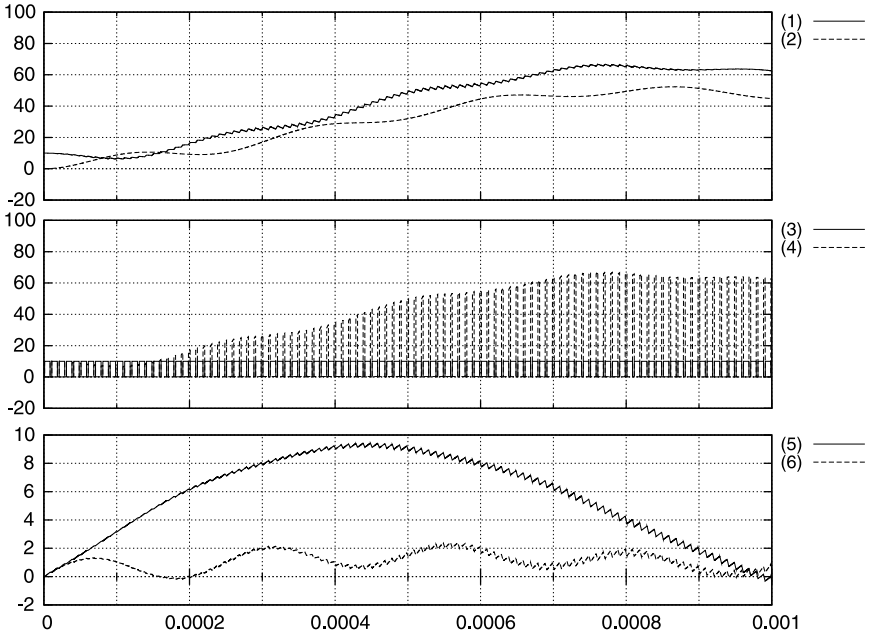
function of period $T$ described in Fig. 7.18. In our numerical simulation, the parameters of the control of the switch are $V_{Gmax} = 10$, $T = 10$ µs, $T_d = 0.95$ µs, $T_{\text{on}} = 7.5$ µs.

The results are displayed in Fig. 7.19 up to $= 0.001$ s and in Fig. 7.20 for the remaining simulation up to $t = 0.01$ s. The standard behaviour of the Ćuk converter is found with the SICONOS software as well as with the SPICE solver and the results are very similar. The main discrepancy between the nonsmooth approach and the SPICE approach is the choice of the time-step. The SICONOS simulations are performed with a time-step $h = 10^{-7}$ s and the SPICE simulations are performed with a time-step of $10^{-10}$ s. This last choice is mainly motivated by the numerical convergence problem of the Newton method when large time-steps are chosen. The gap between this two time-steps results in a gain of CPU time.

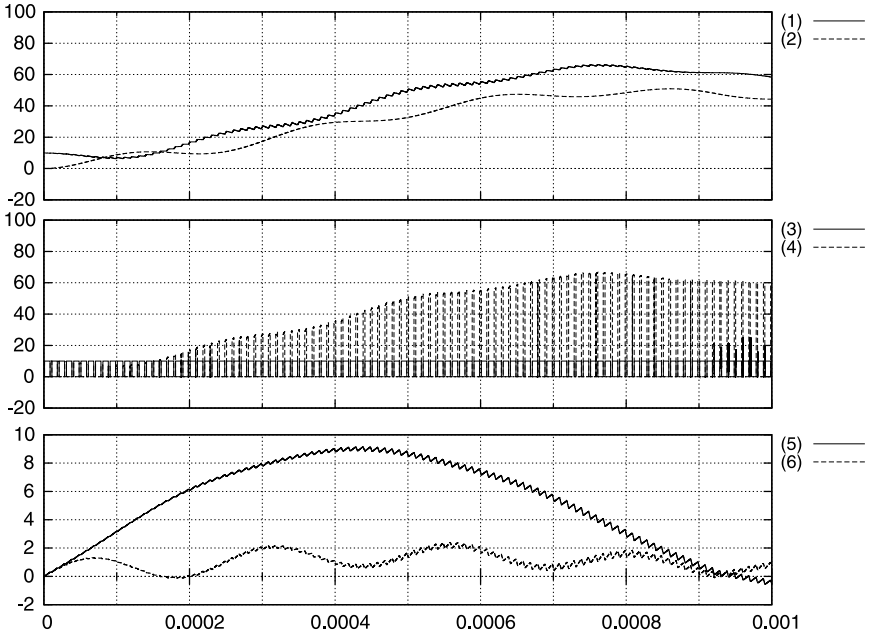
The nonsmooth approach allows the use of larger time-step for the same accuracy in avoiding the numerical convergence problem of the Newton method when the electrical characteristics are stiff. This results in more robustness of the simulating process and in lower simulation times.

## 7.5 A Circuit Exhibiting Sliding Modes

The goal of this section is to focus on the very interesting feature of the nonsmooth approach: the possibility to simulate consistently multivalued components and then

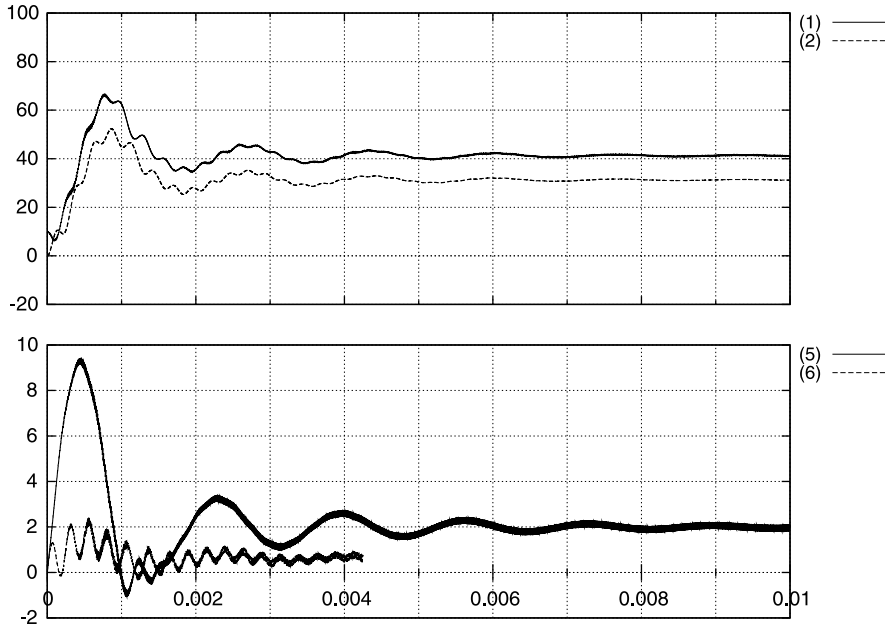



(a) Siconos simulation results

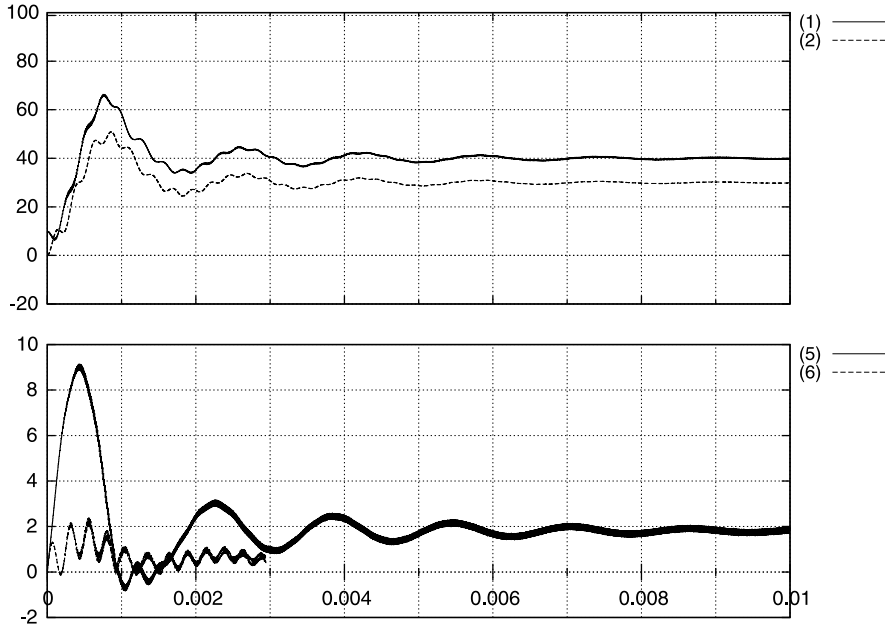



(b) SPICE simulation results

**Fig. 7.19** Simulation values versus time (s) for the Ćuk Converter. (*1*) $C_1$ voltage, (*2*) $C_1$ voltage, (*3*) $V_G$ gate voltage, (*4*) MOS switch drain voltage, (*5*) $L_1$ current and (*6*) $L_2$ current




(a) Siconos simulation results




(b) SPICE simulation results

**Fig. 7.20** Simulation values versus time (s) for the Ćuk Converter. (*1*) $C_1$ voltage, (*2*) $C_1$ voltage, (*5*) $L_1$ current and (*6*) $L_2$ current

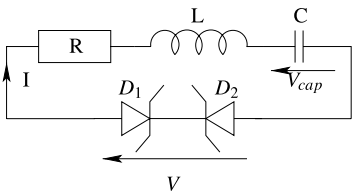



**Fig. 7.21** RLC Zener-diodes circuit

coherently ideal components. The point is not to show that ideal components are better for the physical modeling accuracy, but rather that the regularization or standard hybrid approaches are not convenient for a high-level description and design.

> Our goal is to show that it is better to have a right simulation with an ideal model rather than a simulation which does not work correctly with a regularized model and pseudo-physics.

Let us consider a multivalued component, more precisely, we choose a multivalued behavior of a couple of Zener-diodes as the one which has been already introduced in Sect. 2.5.8. The goal is to outline the efficiency of the nonsmooth model by inclusions to handle such a behavior, even when a sliding mode occurs. In this context, the sliding mode has to be understood as a mode whose operating point of the component is inside the multivalued part of the graph in Fig. 2.24. The circuit is depicted in Fig. 7.21.

## *7.5.1 Models and Dynamical System*

The choice of the vector of unknowns is $[I, V_{cap}]^T$ and it yields:

$$L\frac{dI(t)}{dt} = -RI(t) - V_{cap}(t) + V(t), \qquad C\frac{dV_{cap}(t)}{dt} = I(t). \tag{7.11}$$

### 7.5.1.1 Nonsmooth Model of Double Opposite Zener Diodes

For simplicity's sake, the two Zener diodes $D_1$ and $D_2$ in Fig. 7.21 are modeled as a single component. The behavior of the whole component is given by (see Fig. 2.24):

$$-I \in \mathbb{N}_{[-V_z, V_z]}(V), \tag{7.12}$$

which can be equivalently written as a complementarity problem (see the material in Sect. 2.4.6, especially (2.90)):

$$\begin{cases} V = \lambda_2 - V_z, \\ y_1 = V_z - V, \\ y_2 = I + \lambda_1. \end{cases} \quad \text{and} \quad 0 \leqslant \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \perp \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} \geqslant 0. \tag{7.13}$$

#### 7.5.1.2 A Hybrid Single Valued Ideal Model in VERILOG/ELDO

Using a Netlist and a VERILOG description of the relation (7.14) to represent the couple of Zener diodes, a hybrid single valued model reads as:

$$V = \begin{cases} V_z & \text{if } I < 0, \\ 0 & \text{if } I = 0, \\ -V_z & \text{if } I > 0. \end{cases} \tag{7.14}$$

#### 7.5.1.3 A Smooth Model Using Hyperbolic Tangent Function in ELDO

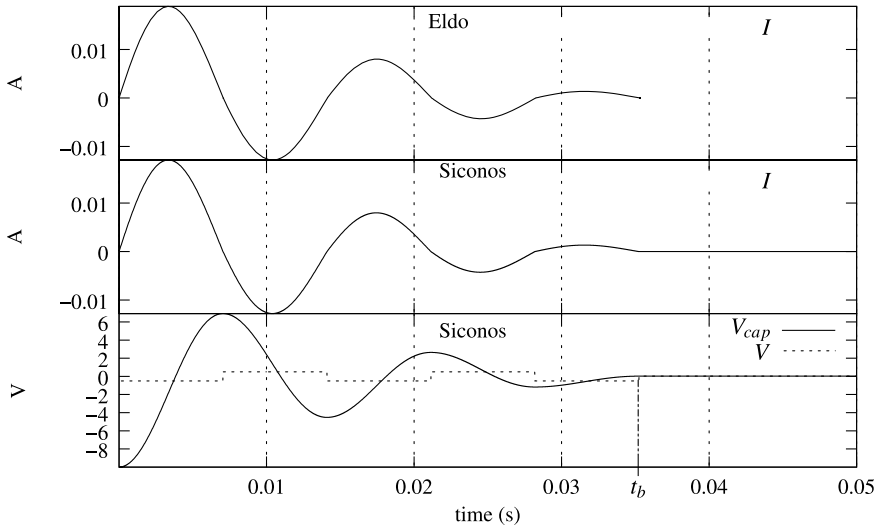
Another approach consists in using the hyperbolic tangent function to approximate the multivalued components $D_1$ and $D_2$, *i.e.* the vertical branch in Fig. 2.24 or in Fig. 1.8(b), is replaced by a stiff smooth curve. The relation between $V$ and $I$ is $V = -V_z \tanh(a\,v)$. The coefficient $a$ is chosen sufficiently large in order to neglect the influence of the regularization. We chose the value $10^5$ to simulate the circuit using ELDO.

### *7.5.2 Simulation and Comparisons*

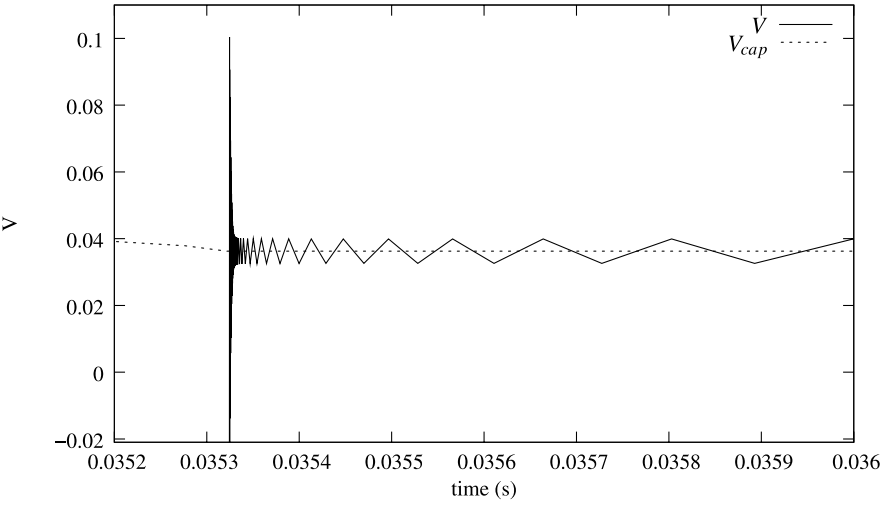
The initial conditions are chosen as $I(0) = 0$ A, $V_{cap}(0) = -10$ V and the value of $V_z$ is 0.5 V. The simulation using the LCP is successfully achieved with SICONOS. The result is shown in Fig. 7.22(a). This electrical circuit dissipates some energy, so $V_{cap}$ oscillates with a decreasing amplitude up to a threshold value $V_z$. After the first event at $t = t_b$, the current $i$ vanishes and the voltage through the capacitor $V_{cap}$ is stabilized to a nonzero value, equal to $v$ through the double Zener diodes component. Notice that this equilibrium point is located in the multivalued part of the characteristic.

Such a behaviour exactly corresponds to the dynamics (7.11–7.12), for which the segment $\{(I, V_{cap}) \mid I = 0, -V_z \leqslant V_{cap} \leqslant V_z\}$ is an attractive sliding surface, attained in a finite time. When $I = 0$, it follows from (7.12) that $V_{cap} \in [-V_z, V_z]$. Figure 7.22(a) shows also the ELDO simulation using the Netlist and the VERILOG relation (7.14). In this case the simulation is correctly done until $t_b$. At the first event at time $t_b$, the simulation cannot be continued because the equilibrium point of the circuit is not handled by the model.

The simulation using the hyperbolic function has been made using ELDO. We focus our attention on the difference due to the regularization of the multivalued




(a) Nonsmooth Model in SICONOS and hybrid model in VERILOG/ELDO simulation.




(b) Regularized model in ELDO. Zoom in the neighborhood of the switching time.

**Fig. 7.22** Simulation of the circuit with a sliding mode

model. Figure 7.22(b) zooms on the moment where the current vanishes. At this instant, the circuit is equivalent to an RLC circuit where the value of $R$ is the coefficient of the tangent to the hyperbolic curve $a$. Note that using a coefficient $a$ larger than $10^5$ leads to an artificial $v$ oscillation around the value of $V_{cap}$. The conclusion is that we cannot expect to observe the convergence toward an ideal behavior with such a regularization.

The NSDS method with Moreau's implicit time-stepping scheme allows one to perfectly simulate sliding mode phenomena. This is not the case for the other approaches.

More details on the simulation of sliding mode systems may be found in Acary and Brogliato (2010).