HAL
open science

# The nonsmooth dynamical systems approach for the analog simulation of switched circuits within the Siconos framework

Vincent Acary, Olivier Bonnefon, Bernard Brogliato

## To cite this version:

Vincent Acary, Olivier Bonnefon, Bernard Brogliato. The nonsmooth dynamical systems approach for the analog simulation of switched circuits within the Siconos framework. [Research Report] RR-7061, INRIA. 2009, pp.26. inria-00423955

**HAL Id: inria-00423955**

**https://inria.hal.science/inria-00423955**

Submitted on 13 Oct 2009

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

# *The nonsmooth dynamical systems approach for the analog simulation of switched circuits within the Siconos framework*

Vincent Acary — Olivier Bonnefon — Bernard Brogliato

**N° 7061**

October 13, 2009

Thème NUM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--7061--FR+ENG

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE
INRIA
centre de recherche GRENOBLE - RHÔNE-ALPES

# The nonsmooth dynamical systems approach for the analog simulation of switched circuits within the Siconos framework

Vincent Acary*, Olivier Bonnefon* , Bernard Brogliato*

Thème NUM — Systèmes numériques
Équipe-Projet Bipop

Rapport de recherche n° 7061 — October 13, 2009 — 26 pages

**Abstract:** The numerical integration of switching circuits is known to be a tough issue when the number of switches is high, or when sliding modes exist. Then classical analog simulators may behave poorly, or even fail. In this paper it is shown on two examples that the nonsmooth dynamical systems (NSDS) approach, which is made of 1) a specific modelling of the piecewise-linear electronic devices (ideal diodes, Zener diodes, transistors), 2) the Moreau's time-stepping scheme, and 3) specific iterative one-step solvers, supersedes simulators of the SPICE family and hybrid simulators. An academic example constructed in [Maffezzoni et al, IEEE Trans. on CADICS, Vol 25, No 11, November 2006], so that the Newton-Raphson scheme does not converge, and the buck converter, are used to make extensive comparisons between the NSDS method and other methods of the SPICE family and a hybrid-like method. The NSDS method, implemented in the Siconos platform developed at INRIA, proves to be on these two examples much faster and more robust with respect to the models parameters variations.

**Key-words:** Switching circuits, complementarity problems, backward Euler algorithm, power converters, complementarity dynamical systems, analog simulation, multivalued systems, unilateral state constraints.

* Electronic mail: `firstname.lastname@inrialpes.fr`

Centre de recherche INRIA Grenoble – Rhône-Alpes
655, avenue de l'Europe, 38334 Montbonnot Saint Ismier
Téléphone : +33 4 76 61 52 00 — Télécopie +33 4 76 61 52 52

**Résumé :** L'intégration numérique en temps des circuits commutés est connue pour être un problème difficile lorsque le nombre de commutations est élévé, ou si des modes glissants apparaissent. Dans ces cas, les simulateurs analogues classiques se comportent mal, ou parfois même échoues. Dans ce rapport, on démontre sur des exemples, que l'approche des systèmes dynamiques non réguliers, constituée de 1) une modélisation spécifique des composants électroniques définis par morceaux (diode idéal, diode Zener, transistors) , 2) du schéma à pas de temps de Moreau et 3) d'algorithmes de résolution itératifs dépasse les capacités des simulateurs classiques SPICE et des simulateurs hybrides. Un exemple académique construit dans [Maffezzoni et al, IEEE Trans. on CADICS, Vol 25, No 11, November 2006] de sorte à faire échouer la convergence du schéma de Newton–Raphson et un convertisseur de puissance de type "buck" sont utilisés pour des comparaisons fouillées entre l'approche des systèmes dynamiques non réguliers et les autres méthodes. L'approche des systèmes dynamiques non réguliers, mis en œuvre dans la plate–forme Siconos développée à l'INRIA, démontre sur ces deux exemples qu'elle est beaucoup efficace et robuste par rapport aux variations de paramètres des modèles.

**Mots-clés :** Circuits commutés, problème de complémentarité, algorithme d'Euler implicite, convertisseurs de puissance, systèmes dynamiques de complémentarité, simulation analogique, systèmes multivalués, contraintes unilatérales sur l'état.

# Contents

**1 Introduction** **4**

**2 The nonsmooth dynamical systems approach.** **5**
2.1 nonsmooth electronic devices modelling . . . . . 5
2.1.1 nonsmooth diodes . . . . . 6
2.1.2 nonsmooth switches . . . . . 7
2.1.3 nonsmooth MOSFET transistors . . . . . 8
2.1.4 nonsmooth comparator . . . . . 10
2.2 The dynamical equations . . . . . 10
2.2.1 The nonsmooth DAE formulation . . . . . 10
2.3 Moreau's time-stepping scheme . . . . . 11
2.4 Numerical solvers for the OSNSP (21) . . . . . 11
2.4.1 $K$ is a finite representable convex set . . . . . 12
2.4.2 $K$ is a generalized box . . . . . 12

**3 Automatic circuit equation generation and Software implementation** **12**
3.1 Automatic generation of the dynamical equations . . . . . 12
3.2 Software aspects . . . . . 13

**4 An elementary switching circuit** **14**
4.1 The dynamical system . . . . . 14
4.2 Numerical results with Siconos . . . . . 15
4.3 Numerical results with Eldo . . . . . 16

**5 Results on the buck converter** **16**
5.1 The dynamical equations . . . . . 17
5.2 Numerical results with Siconos, and comparisons . . . . . 17
5.2.1 Simulation with Siconos . . . . . 17
5.2.2 Simulation with Spice . . . . . 19
5.2.3 Simulation comparisons . . . . . 20
5.2.4 Sliding mode using a multi-valued comparator . . . . . 20
5.2.5 Simulation with Plecs . . . . . 22

**6 Conclusions** **22**

# 1 Introduction

It is well know that conventional accurate analog simulation tools, which are based on the Newton–Raphson nonlinear solver, can have serious drawbacks when they are used for the integration of nonsmooth circuits, containing switches and piecewise linear components (like ideal diodes and transistors). This is especially true when the number of events becomes too large, or when sliding modes exist, which is common in practice. Then analog (Spice-like) tools may become very time consuming, or provide very poor results with chattering [31], or even fail [40, 56, 41, 19, 13]. The same applies to "hybrid" integrators that consider an exhaustive enumeration of all the system's modes, which have a very limited scope of application because of the exponential growth of the number of modes that have to be simulated separately. Along the same lines, event-driven schemes can hardly simulate systems with large number of events, because they soon become quite time-consuming and do not allow for accumulations of events [2].

It is therefore clear that other types of numerical schemes have to be applied for highly nonsmooth switching circuits. Since a numerical method always relies on a specific modelling approach, a logical path is to first reconsider the models of nonsmooth components (diodes, switches, transistors, *etc*) so that efficient numerical solvers can be applied. The nonsmooth dynamical systems (NSDS) approach, which is the one chosen in this paper, appears to be a suitable framework for the simulation of nonsmooth circuits, allowing one to efficiently simulate systems with very large number of events, and sliding mode trajectories. It consists of modelling nonsmooth components as piecewise linear functions, with possible vertical branches (inducing some unilaterality in the system, hence possible state jumps, when these branches are infinite). The time-discretization of such nonsmooth systems then yields various types of so-called One-Step NonSmooth Problems (OSNSP), for instance (linear) complementarity problems or nonlinear (or quadratic) programs with equality-inequality constraints. The NSDS approach may then take advantage of the quite important works that have been led by the Nonlinear Programming community concerning the development of efficient solvers for complementarity problems [27] and optimization tools [35], and also by the Contact Mechanics community [2], where Moreau and Jean developed the so-called Nonsmooth Contact Dynamics (NSCD) method within the theoretical framework of Moreau's sweeping process [44, 36, 45]. The numerical method that is used in this paper, owes a lot to the NSCD method of mechanics, and will be named *Moreau's time-stepping scheme*. As alluded to above, nonsmooth components are often represented with piecewise-linear functions, or with complementarity relations, or with inclusions into normal cones. The piecewise-linear modelling approach in nonsmooth electrical circuits has been pioneered by Chua et al in [37, 18, 17], and complementarity problems have been introduced in [51, 50, 52], followed by the works of Leenaerts and van Bokhoven [39, 38], Vlach et al [9, 54, 55, 59]. Camlibel et al [28, 14] studied the convergence of backward Euler methods, and comparisons with other (analog and hybrid) integrators are proposed in [53]. Glocker et al [33, 42] led interesting developments showing the analogy between mechanics and electricity for various types of nonsmooth components, and also proposed a time-stepping method inspired by Moreau's algorithm for contact mechanics (consequently quite close to the algorithm used in this paper). Variational inequalities of the second kind and electrical superpotentials were recently introduced in electronics in [6, 34, 7] to study the existence and uniqueness of solutions for static circuits, or the equilibria of dynamical circuits with nonsmooth devices. Other works may be found in [26, 8, 32].

The objective of this paper is twofold: firstly it is shown on an academic example taken from [40] that the NSDS approach allows one to simulate a nonsmooth system for which conventional analog methods fail (roughly speaking, the iterative solver for complementarity problems converges, whereas Newton-Raphson's method does not); secondly, numerical results for a buck converter are presented and comparisons with other (analog and hybrid) tools are done. The buck converter example in fact demonstrates on a significant case study that the proposed time-stepping method is efficient for systems with a large number of events. Compared to previous works [33, 53], the ideal switches are here modelled and simulated for the first time in a completely implicit way, the advantage of which will be explained. The simulations are done with the Siconos software

platform[1] of the INRIA [2, 4, 5], that is an open-source software package dedicated to nonsmooth dynamical systems. The paper is organized as follows: in Section 2 the modelling and general time-discretization frameworks are recalled; in Section 3 the automatic circuit equation generation and software aspects are outlined; in Section 4 an elementary closed-loop switching circuit taken from [40] is simulated; in Section 5 the buck converter example is treated and comparisons are presented. Conclusions end the paper.

**Notation:** The following tools will be used in this paper. Let $K \subseteq I\!R^n$ be a non empty convex set. The normal cone to $K$ at $x \in I\!R^n$ is $N_K(x) = \{z \in I\!R^n | \langle z, \zeta - x\rangle \leqslant 0 \text{ for all } \zeta \in K\}$. The projection in the euclidean metric of a vector $x \in I\!R^n$ onto $K$ is denoted as $\text{proj}[K; x]$. A singleton is denoted as $\{t\}$. The identity matrix of $I\!R^{m\times m}$ is denoted by $\mathsf{I}_m$ and the zero vector in $I\!R^m$ by $0_m$.

The following standard mathematical programming problems will be used throughout this paper.

**Definition 1 (Variational Inequality [27])** *Given a function $\mathsf{F} : I\!R^p \to I\!R^p$, and $\Omega$ a non empty subset of $I\!R^p$, the Variational Inequality (VI) problem is to find a vector $z \in I\!R^p$ such that*

$$\mathsf{F}^T(z)(y - z) \geqslant 0, \forall y \in \Omega. \tag{1}$$

❑

**Definition 2 (Inclusion into a normal cone [47])** *Given a function $\mathsf{F} : I\!R^p \to I\!R^p$, and $K$ a non empty convex subset of $I\!R^p$, the inclusion into a normal cone problem is to find a vector $z \in I\!R^p$ such that*

$$0 \in \mathsf{F}(z) + N_K(z) \tag{2}$$

❑

If $K = \Omega$ is a convex set, the inclusion (2) and the VI (1) are equivalent.

**Definition 3 (Mixed Complementarity Problem [24])** *The Mixed Complementarity Problem (MCP) is defined as follows. Given a function $\mathsf{F} : I\!R^p \to I\!R^p$, lower and upper bounds $l, u \in (I\!R \cup \{+\infty, -\infty\})^p$, find $z \in I\!R^p$, $w, v \in I\!R^p_+$, such that*

$$\begin{cases} \mathsf{F}(z) = w - v \\ l \leqslant z \leqslant u, \quad (z - l)^T w = 0, \quad (u - z)^T v = 0 \end{cases} \tag{3}$$

❑

Notice that a solution to the MCP satisfies the inclusion $-F(z) \in N_{[l,u]}(z)$. If the $\mathsf{F}(\,\cdot\,)$ in (3) is affine, *i.e.*

$$\begin{cases} Mz + q = w - v \\ l \leqslant z \leqslant u, \quad (z - l)^T w = 0, \quad (u - z)^T v = 0 \end{cases} \tag{4}$$

for some matrix $M \in I\!R^{p\times p}$ and some vector $q \in I\!R^m$, the MCP (3) defines a Mixed Linear Complementarity Problem (MLCP).

# 2 The nonsmooth dynamical systems approach.

## 2.1 nonsmooth electronic devices modelling

The NSDS approach for the modelling of piecewise linear components in electrical circuits has been described in detail in several of the above cited publications [2, 33, 44, 36, 45], and will just be recalled here for the sake of readability. The NSDS approach is a package that consists of: a) nonsmooth models, b) Moreau's time-stepping algorithm, c) OSNSP solvers. The current-voltage

[1] http://siconos.gforge.inria.fr/

laws of nonsmooth electronic devices may all be represented as inclusions into a normal cone to a convex set $K$, *i.e.* $0 \in \Phi(y, \lambda, t) + N_K(\lambda)$, where $\Phi(\,\cdot\,)$ is a function, $y$ and $\lambda$ are implicitly defined from $0 = H(X, \lambda, t)$ and $y = G(X, \lambda, t)$ for some functions $H(\,\cdot\,)$ and $G(\,\cdot\,)$, and $X$ is the state vector of the circuit, composed of branch voltages and currents. A crucial point for simulation efficiency, however, is to keep as less slack variables, $\lambda$ and $y$ as possible in the device representation. In addition some efficient OSNSP solvers (as they will be described in Section 2.4) use directly such inclusions into a normal cone to a convex set, or the equivalent VI formulation. This is the case for the direct MCP solvers that we used in our simulations. Finally, it is noteworthy that the inclusion modelling of the devices allows for nonlinear characteristics which may not be represented by complementarity relations.

Let us illustrate this on the above four examples (ideal diode, switch, transistor, comparator).

### 2.1.1 nonsmooth diodes

The notation for the currents and the potentials at the ports of the diode is depicted in Fig. 1. Four models of diodes are depicted in Fig. 2:

a) the smooth exponential Shockley model in Fig. 2(a) defined by the smooth constitutive equation,

$$i(t) = i_s \exp(-\frac{v(t)}{\alpha} - 1), \tag{5}$$

where $i_s$ and $\alpha$ are physical parameters of the diode,

b) ideal diodes with possible residual current $-a$ and voltage $-b$ in Fig. 2(b) defined by the following complementarity condition

$$0 \leqslant i(t) + a \perp v(t) + b \geqslant 0, \tag{6}$$

where the $x \perp y$ means that $x^T y = 0$ and $a$ and $b$ are the threshold values for $i$ and $v$,

c) the "hybrid" model which considers the two modes separately with for instance an associated Modelica [25] script in Fig. 2(c)

$$\begin{array}{rcl} \mathsf{off} & = & \mathsf{s} < 0 \\ v(t) & = & \textbf{if} \;\; \mathsf{off} \;\; \textbf{then} \;\; -\mathsf{s} \;\; \textbf{else} \;\; 0 \\ i(t) & = & \textbf{if} \;\; \mathsf{off} \;\; \textbf{then} \;\; 0 \;\; \textbf{else} \;\; \mathsf{s}, \end{array} \tag{7}$$

d) a piecewise-linear model in Fig. 2(d) defined by

$$v(t) = \begin{cases} -R_{\mathsf{on}}\, i(t) & \text{if } v(t) < 0 \\ -R_{\mathsf{off}}\, i(t) & \text{if } v(t) \geqslant 0 \end{cases}, \tag{8}$$

with $R_{\mathsf{on}} \ll 1$ and $R_{\mathsf{off}} \gg 1$ are the equivalent resistive values of each branches.

The ideal diode model in Fig. 2(b) is chosen in this paper. The drawbacks of the Shockley law is that it introduces high stiffness in the dynamical equations. The hybrid model becomes rapidly unusable if the number $m$ of diodes increases, since the number of modes to be described in the associated script varies as $2^m$. This will be shown on the converter example. The model 2(d) leads a badly conditionned algorithm used to solve the OSNSP in Section 2.4. On the contrary the ideal model of Fig. 2(b) yields, when introduced in the dynamics, well-conditioned complementarity problems, that yield time-stepping methods for which efficient solvers exist. Showing the efficiency of these methods is the object of this paper.

Quite similar developments and comments may be made for ideal Zener diodes, piecewise linear practical diodes and practical Zener diodes, see e.g. [2, 7]. From basic convex analysis one deduces that the ideal diode of Fig. 2 (b) has the following current/voltage law:

$$i(t) \in \{b\} + N_{]-\infty,a]}(v(t)) \;\Leftrightarrow\; v(t) \in \{a\} + N_{]-\infty,b]}(i(t)) \tag{9}$$

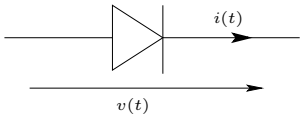
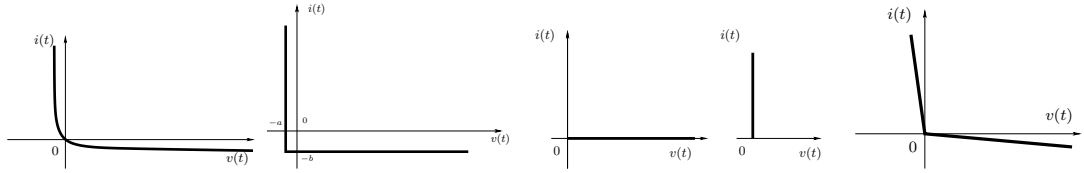



Figure 1: Diode symbol.




(a) smooth modeling (b) nonsmooth modeling (c) hybrid modelling (d) equivalent resistor model

Figure 2: Four models of diodes.

Similar inclusions for ideal Zener diodes may be found in [2, 7], that take the form $i(t) \in N_{[0,V_z]}(v(t))$ for some $V_z > 0$. The piecewise-linear diode of Fig. 2 (d) can be represented as:

$$\begin{cases} v(t) = \frac{1}{2}(\tau(t)-1)R_{\text{off}}i(t) - \frac{1}{2}(1+\tau(t))R_{\text{on}}i(t) \\ \tau(t) \in \text{sgn}(v(t)) \;\Leftrightarrow\; v(t) \in -N_{[-1,1]}(\tau(t)) \end{cases} \tag{10}$$

that is consistent with the MLCP formulation in (4). The function $\text{sgn}(\,\cdot\,)$ is the multi-valued sign function defined by

$$\text{sgn}(x) = \begin{cases} 1 \text{ if } x > 0 \\ -1 \text{ if } x < 0 \\ [-1,1] \text{ if } x = 0 \end{cases} . \tag{11}$$

The piecewise-linear model yields a condition number of the resulting MLCP matrix close to $R_{\text{off}}/R_{\text{on}}$, that causes trouble with the numerical algorithms that are used to solve the OSNSP. Inclusions as in (9) will be preferred as they can be directly used in the numerical algorithm for MCP, yielding well-posed and well-conditioned MCPs.

### 2.1.2 nonsmooth switches

The notation for the currents and the potentials at the ports of the ideal switch is depicted on Fig. 3. The switches are modelled in two ways in this paper. The first model, that is applied to the elementary example of Section 4, consists of:

$$v(t) = \begin{cases} R_{\text{off}}\; i(t) & \text{if } u_c(t) < 0 \\ R_{\text{on}}\; i(t) & \text{if } u_c(t) \geqslant 0 \end{cases} \tag{12}$$

where the voltage $u_c(\,\cdot\,)$ is a state variable of the overall dynamical system, $v(\,\cdot\,)$ is the voltage of the switch and $i(\,\cdot\,)$ is the current through the switch. The resistors $R_{\text{off}} \gg 1$ and $R_{\text{on}} \ll 1$ are chosen by the designer. In the case of the buck converter of Section 5, the switch is modelled with transistors, as is most common in the industrial practice. The switch in (12) is modeled as follows:

$$\begin{cases} v(t) = \frac{1}{2}(1+\tau(t))R_{\text{on}}i(t) + \frac{1}{2}(1-\tau(t))R_{\text{off}}i(t) \\ \\ \tau(t) \in \text{sgn}(u_c(t)) \;\Leftrightarrow\; u_c(t) \in -N_{[-1,1]}(\tau(t)) \end{cases} \tag{13}$$

The difference with respect to the diode (10) is that the "input" to the inclusion is an external voltage. It is noteworthy that the voltage $v(t)$ in (12) is discontinuous at $u_c(t) = 0$ for any $i(t) \neq 0$, the jump magnitude being equal to $|(R_{\text{off}} - R_{\text{on}})i(t)|$. The choice that is made in (13) implies that

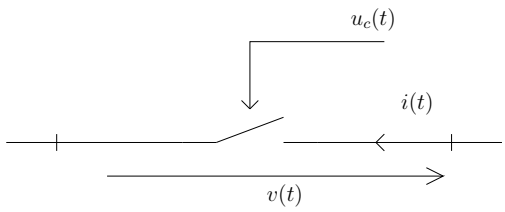



Figure 3: Ideal switch symbol.

the discontinuities are "filled-in" and the model is consequently multivalued at $u_c(t) = 0$, $i(t) \neq 0$. This is precisely what allows one to smoothly simulate the sliding-modes [3].

**Remark 1** *The ideal switch is modelled in [33] with a relay multifunction whose threshold may vary between 0 and $+\infty$, and the switch is controlled by a current variable of the circuit, in an explicit way. Compared to [53] our approach differs a lot since [53] models the switch through a so-called cone complementarity problems, with an exogenous excitation that makes the cones switch between $\{0\}$ and $\mathbb{R}$ or $\mathbb{R}^+$. The choice we made in this paper is motivated by the industrial practice and the way switches are modelled in Mentor Graphics'* ELDO *software package*[2]*, that is one of the main analog simulation tool of the market and may be considered as a reference for simulation results comparisons. Another way to model switches is to compute the topology changes after each "open" and "close" operation. As pointed out above such an approach rapidly becomes extremely time-consuming when the number of switches grows (the number of different topologies grows exponentially fast with the number of switches), and does not allow for finite accumulations of switches or sliding mode trajectories. An open issue would be the implicit discretization of the ideal switches models of [33] and [53] which is not directly possible and is not tackled in this paper.*

#### 2.1.3 nonsmooth MOSFET transistors

Following [38], let us consider the Sah model of the nMOS static characteristic:

$$I_{DS} = \frac{K}{2} \cdot (f(V_G - V_S - V_T) - f(V_G - V_D - V_T)) \tag{14}$$

with $K = \mu \frac{\epsilon_{OX}}{t_{OX}} \frac{W}{L}$, $\mu$ mobility of majority carriers, $W$ and $L$ channel width and length, $\epsilon_{OX}$ the permissivity of the silicon oxide of thickness $t_{Ox}$. The voltage $V_T$ is the threshold voltage depending on the technology.

The notation for the currents and the potentials at the ports of the nMOS is depicted on Fig. 4. The function $f : \mathbb{R} \longrightarrow \mathbb{R}$ in (14) is defined as:

$$f(x) = \begin{cases} 0 & \text{if } x < 0 \\ x^2 & \text{if } x \geqslant 0 \end{cases}$$

The piecewise and quadratic nature of this function is approximated by the following $s$ segments piecewise linear function [38]:

$$f_{\mathsf{pwl}}(x) = \alpha_i x + \beta_i, \text{ for } a_i \leqslant x \leqslant a_{i+1}, \quad i = -1 \ldots s+1 \tag{15}$$

with $a_{-1} = -\infty$ and $a_{s+1} = +\infty$. The complete model of the piecewise-linear nMOS transistor with $s$ segments in (15) can be recast under the following mixed linear complementarity form [38]:

[2] `http://www.mentor.com/products/ic_nanometer_design/analog-mixed-signal-verification/eldo/`

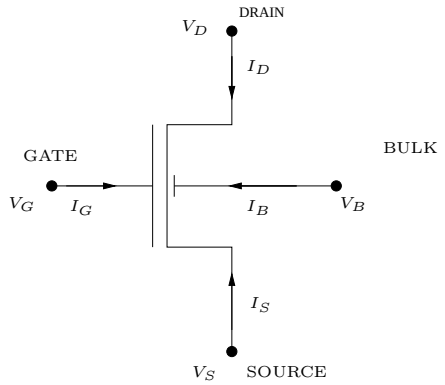



Figure 4: nMOS transistor symbol.

$$
\begin{aligned}
y(t) &= \left[ \underbrace{\begin{matrix} 0 & \dots & 0 \\ -b & \dots & -b \end{matrix}}_{\times s-1} \quad \underbrace{\begin{matrix} -b & \dots & -b \\ 0 & \dots & 0 \end{matrix}}_{\times s-1} \right]^T v(t) + \lambda(t) \\
&\qquad + \left[ \begin{matrix} h_1 & \dots & h_{s-1} & h_1 & \dots & h_{s-1} \end{matrix} \right]^T \\
0 &= \mathsf{I}_3\, i(t) + \left[ \begin{matrix} -c_1 & \dots & -c_{s-1} & c_1 & \dots c_{s-1} \\ 0 & 0 & 0 & 0 & 0 \\ c_1 & \dots & c_{s-1} & -c_1 & \dots - c_{s-1} \end{matrix} \right] \lambda(t) \\
0 &\leqslant y(t) \perp \lambda(t) \geqslant 0
\end{aligned}
\tag{16}
$$

with

$$
u(t) = \left[ \begin{matrix} U_{GD}(t) = V_G(t) - v_D(t) \\ U_{GS}(t) = V_G(t) - v_S(t) \end{matrix} \right], \quad i(t) = \left[ \begin{matrix} I_D(t) \\ I_G(t) \\ I_S(t) \end{matrix} \right]. \tag{17}
$$

The parameters are given as follows: $b = \frac{K}{2}$, $h_i = b(V_T + a_i), i = 1 \dots 5$. The values of $c_i$ are computed from the linear approximation $\alpha_i$ and $\beta_i$ in (15). Using some basic convex analysis, one obtains the compact formulation of (16) (17):

$$
\left\{ \begin{array}{l} -\lambda(t) + Bu(t) + h(t) \in N_K(\lambda(t)) \\ y(t) = Bu(t) + \lambda(t) + h(t) \\ 0 = i(t) + C\lambda(t) \end{array} \right. \tag{18}
$$

with $K = (I\!R_+)^{2(s-1)}$. In the case of the MOSFET transistor, the inclusion is an equality as expected since its piecewise-linear characteristic is single valued. The pMOS transistor is represented in the same way, changing the values of $h_i$, $i(t)$ to $-i(t)$ and $b$ to $-b$.

**Remark 2** *The piecewise-linear model in (15) has s segments. Multiple choices are possible in order to adjust the number of slack variables and consequently the size of the OSNSP-MLCP to be solved at each step with respect to the accuracy. In practice one should therefore be very careful about choosing a reasonable piecewise-linear approximation of the devices so that the MLCP size does not increase too much. In this work we have chosen a model using* 6 *segments. A study of the results accuracy and computation time as a function of the number of segments is outside the scope of this paper.*

#### 2.1.4 nonsmooth comparator

The comparator device as depicted in Fig. 5 is modelled as a piecewise-linear function whose value is $V_{min}$ if $x < -\epsilon$V and 3 if $x > \epsilon$V. Setting $\epsilon$ to 0 leads to a relay function that is multivalued at 0. In this case, similarly to the Zener diode the multivalued comparator is represented as:

$$V_{output} \in N_{[V_{\min},V_{\max}]}(V_+ - V_-), \tag{19}$$

where $V_{\min}$ and $V_{\max}$ are the saturation thresholds.

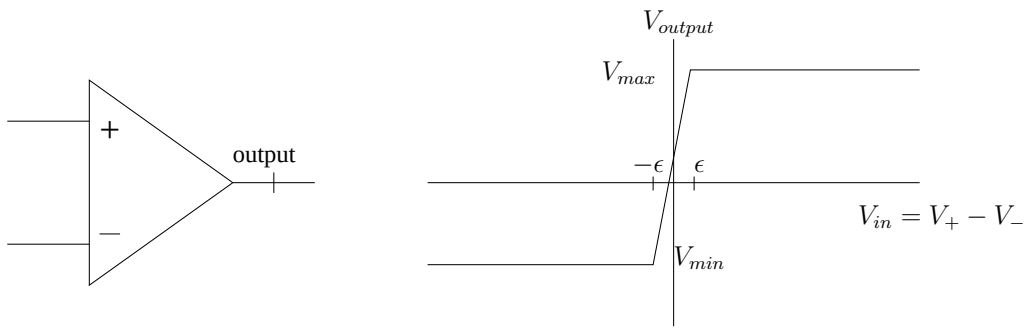



Figure 5: Comparator model.

## 2.2 The dynamical equations

Section 2.1 is devoted to present the electronic devices models and their mathematical representations to be inserted in the circuits dynamics in order to obtain a suitable formalism for the subsequent time-discretization. In particular the OSNSP solver to be used strongly influences the modelling choice. In this section we focus on the dynamical equations which are suitable for the NSDS approach.

#### 2.2.1 The nonsmooth DAE formulation

The circuit with nonsmooth components represented as inclusions and equalities, and the smooth non linear behavior of the network represented as Differential Algebraic Equations (DAE) can be written compactly as:

$$\left\{ \begin{array}{rl} M(X,t)\dot{X} = F(X,t) + U(t) \ ] & \text{DAE} \\ \left.\begin{array}{r} 0 = H(X,\lambda,t) \\ y = G(X,\lambda,t) \end{array}\right] & \begin{array}{l}\text{Input/output relations on} \\ \text{nonsmooth components}\end{array} \\ 0 \in \Phi(y,\lambda,t) + N_K(\lambda) & \text{"Inclusion rule"} \end{array} \right. \tag{20}$$

where $X \in I\!R^n$ is the state composed of the potentials and the currents in inductive, voltage–defined and non–smooth branches. The vectors $y, \lambda \in I\!R^m$ are the slack variable expressing the nonsmooth multi–valued models of the components. The functions $M : I\!R^n \times I\!R \to I\!R^{p\times n}$, $H : I\!R^n \times I\!R^m \times I\!R \to I\!R^{n-p}$, $G : I\!R^n \times I\!R^m \times I\!R \to I\!R^m$ and $\Phi : I\!R^m \times I\!R^m \times I\!R \to I\!R^m$ are assumed to be continuously differentiable functions. More details will be given on the choice of state variables and the structure of the functions in Section 3.1. One recognizes two basic ingredients: the DAE part, that is coupled to the nonsmooth electrical devices represented by inclusions into normal cone as those developed in Section 2.1. See Section 4 for a concrete example of (20).

## 2.3 Moreau's time-stepping scheme

Starting from the dynamics in (20) the Moreau's time-stepping scheme is as follows:

$$\left\{\begin{array}{l} M(X_{k+\theta},t_{k+\theta})(X_{k+1}-X_k) = hF(X_{k+\theta},t_{k+\theta}) \\ \qquad\qquad\qquad\qquad\qquad\qquad +hU(t_{k+\theta}) \\ 0 = H(X_{k+1},\lambda_{k+1},t_{k+1}) \\ y_{k+1} = G(X_{k+1},\lambda_{k+1},t_{k+1}) \\ \\ 0 \in \Phi(y_{k+1},\lambda_{k+1},t_{k+1}) + N_K(\lambda_{k+1}) \end{array}\right. , \tag{21}$$

for a time–step $h > 0$ and with the usual following notation. For a function $f(t)$, $f_{k+1} \approx f(t_{k+1})$ and $f_{k+\theta} = \theta f_{k+1} + (1-\theta) f_k$. The Moreau's time stepping algorithm is made of a $\theta$-method with $\theta \in [0,1]$ for the assumed sufficiently smooth terms, and a fully implicit scheme for the inclusion rule which can be non–smooth. This choice is led by two fundamental reasons. The first reason is the respect of the inclusion rule and its intrinsic multi–valued and unilateral character. As it has been shown in [43], only a fully implicit scheme can satisfy the unilateral constraints in discrete time. The second reason is the possible non–smoothness of the evolution which can be numerically integrated in a consistent way by implicit low order schemes only [2].

For the numerical purposes, let us rewrite the problem (21) in a global inclusion form

$$0 \in \mathsf{F}(z) + N_C(z), \tag{22}$$

where the variable $z = [X_{k+1}, y_{k+1}, \lambda_{k+1}]^T \in I\!R^{n+2m}$ and the function $\mathsf{F} : I\!R^{n+2n} \to I\!R^{n+2m}$ is defined by

$$\mathsf{F}(z) = \left[\begin{array}{l} M(\theta X_{k+1} + (1-\theta)X_k, t_{k+\theta})(X_{k+1}-X_k) \\ \quad -hF(\theta X_{k+1} + (1-\theta)X_k, t_{k+\theta}) - hU(t_{k+\theta}) \\ H(X_{k+1},\lambda_{k+1},t_{k+1}) \\ G(X_{k+1},\lambda_{k+1},t_{k+1}) - y_{k+1} \\ \Phi(y_{k+1},\lambda_{k+1},t_{k+1}) \end{array}\right]. \tag{23}$$

The normal cone $N_C$ is the normal cone to the following convex set

$$C = I\!R^n \times I\!R^m \times K \subset I\!R^{n+2m} \tag{24}$$

We will see in the next section that the nonlinearity of $\mathsf{F}(.)$ can be directly treated by the numerical one–step solver. Another approach is to perform an outer Newton linearization of this problem by searching the solution as the limit for $\alpha$ of the following linearized problem

$$0 \in \nabla_z \mathsf{F}(z^\alpha)(z^{\alpha+1} - z^\alpha) + \mathsf{F}(z^\alpha) + N_C(z^{\alpha+1}). \tag{25}$$

for a given $z^0$. At each time–step $k$ and at each Newton iteration $\alpha$, the problem (25) appears to be affine in $z$.

## 2.4 Numerical solvers for the OSNSP (21)

The problem (21) is a VI written in the form of an inclusion into a normal cone to a convex set as in (22). The choice of the numerical solver for (21) depends mainly on the structure of the convex set $K$. Indeed, from a very general convex set $K$ to a particular choice of $K$, the numerical solvers range from the numerical methods for VI to nonlinear equations, passing through various complementarity problems solvers. The convergence and the numerical efficiency are improved in proportion as the structure of $K$ becomes simpler. In the sequel, majors choices of $K$ will be given leading to various classes of well-known problems in mathematical programming theory. We refer to [27] for a thorough presentation of available numerical solvers and to [2, Chapter 12]) for a comprehensive summary of numerical algorithms. In the numerical example presented in this paper, various numerical methods described below are used according to the type of the one–step nonsmooth problem and will be further precised.

#### 2.4.1 $K$ is a finite representable convex set

In practice, the convex set is finitely represented by

$$K = \{\lambda \in I\!R^m, h(\lambda) = 0, g(\lambda) \geqslant 0\}, \tag{26}$$

where the functions $h : I\!R^m \to I\!R^m$, $g : I\!R^m \to I\!R^m$ are assumed to be smooth with non vanishing Jacobians. In this case, general algorithms for VI can be used. To cite a few, the minimization of the so-called regularized gap function [29, 57, 58] or generalized Newton methods [27, Chapter 7& 8] can be used. If $F(.)$ is affine (possibly after the linearization step described in (25)) and the functions $h(.)$ and $g(.)$ are also affine, the VI is said to be an affine VI for which the standard pivoting algorithms for LCP[20] has been extended in [15].

#### 2.4.2 $K$ is a generalized box

Let us consider the case that $K$ is a generalized box in $\overline{I\!R}^m = \{I\!R \cup \{+\infty, -\infty\}\}^m$, that is

$$K = \{\lambda \in I\!R^m, a_i \leqslant \lambda_i \leqslant b_i, a_i \in \overline{I\!R}, b_i \in \overline{I\!R}, i = 1 \dots m\}, \tag{27}$$

In this case, the problem (22-24) can be recast in a Mixed Complementarity Problem (MCP) by defining $p = n + m + m + m$ and the bounds $l, u$ as $l = [\ 0_n \quad 0_m \quad 0_m \quad a\ ]^T$ and $u = [\ 0_n \quad 0_m \quad 0_m \quad b\ ]^T$.

The MCP (3) can be solved by a large family of solvers based on Newton–type Methods and interior-points techniques. In contrast to the interior-point methods, it is not difficult to find comparisons of numerical methods based on Newton's method for solving MCP. We refer to [11] for an impressive comparison of the following algorithms: MILES [48], PATH [24], NE/SQP [30, 46], QPCOMP [12], SMOOTH [16], PROXI [10], SEMISMOOTH [22], SEMICOMP [10]. All of these comparisons, which have been made in the framework of the MCP (3) show that the PROXI, PATH and SMOOTH are superior on a large sample of test problems. For a comparison of the variants of the SEMISMOOTH algorithm, we refer to [21].

If $\mathsf{F}(.)$ is affine, the MLCP is equivalent to a box-constrained affine VI. For this problem, the standard pivoting algorithm such the Lemke's Method has been extended in [49].

A special case of a generalized box is the positive orthant of $I\!R^m$, that is $K = I\!R^m_+$. Standard theory and most of the numerical algorithms for LCPs apply in this MCLP case.

When the circuit is simple and of low size in terms of the number of state variables, it is sometimes possible to write the DAE as an ODE and perform the explicit substitution of $X$ by $y$ and $\lambda$ in the formulation (21). If the cone is also simply defined by a positive orthant, we arrive then at a standard LCP [23]. Unfortunately, the LCP formulation is not amenable for more complicated cases where an automatic circuit equation formulation (see the next section) is used.

# 3 Automatic circuit equation generation and Software implementation

In this section, the choice of the state variables and the formulation of the equations of motion are motivated by the compromise between the automatic character of the equation formulation and the efficiency of the numerical algorithm. The efficiency is based partly on the number of state and slack variables and partly on the conditioning of the formulation. Finally, some insights are given on the software implementation of the methods.

## 3.1 Automatic generation of the dynamical equations

Let us describe briefly how the dynamical equations are obtained for the two systems which are analyzed in this paper. There are basically three choices for the state variables, based on the charge approach, the flux approach, and the current-voltage approach. The latter is chosen here.

There are a lot of methods to build a smooth DAE formulation of standard electrical circuits. To cite a few of them, the Sparse Tableau Analysis (STA) and the modified Nodal Analysis (MNA) are the most widespread. An automatic circuit equation generation system extending the MNA has been developed at the INRIA, see the patent [1]. A straightforward extension of the MNA (or of the STA) can be performed by directly replacing the constitutive equations of the nonsmooth components with the corresponding inclusion rule yielding the system (20). Nevertheless, the fact that $M(X,t)$ is not a square matrix and the use of many superfluous variables and algebraic equations has the following drawbacks : a) the numerical efficiency of the algorithms is weakened by the larger size of the problem and b) the OSNSP solvers can be in trouble due to the redundancy of constraints, which is difficult to circumvent in the numerical procedure (mainly due to the machine accuracy constraint). Many alternate formulations have been tested. It has been concluded that a suitable adaptation of the MNA leads to the suitable following formulation

$$\left\{\begin{array}{ll} \left.\begin{array}{ll} \dot{x} & = f_1(x,z,t)+U(t) \\ 0 & = f_2(x,z,t) \end{array}\right] & \text{Semi-Explicit DAE} \\ \left.\begin{array}{ll} 0 & = h(x,z,\lambda,t) \\ y & = g(x,z,\lambda,t) \end{array}\right] & \begin{array}{l}\text{Input/output relations on} \\ \text{nonsmooth components}\end{array} \\ 0 \in \Phi(y,\lambda,t)+N_K(\lambda) & \text{"Inclusion rule"} \end{array}\right. \tag{28}$$

where $x \in I\!R^n$ corresponds to the current in the inductive branches and the voltages in the capacitive branches, $z \in I\!R^p$ collects all the node potentials, the currents in the voltage–defined and non–smooth branches and the currents in a subset of the capacitive branches. The choice and the construction of the latter subset of branches is described in details in [1]. The automatic circuit equation formulation starts from the MNA: it adds some unknowns to get a semi-explicit system, and replaces the constitutive equations of the nonsmooth components by the corresponding inclusion rule. Starting from (28), the numerical algorithm as explained in Section 2.4 is used in a similar manner on the time–discretized system,

$$\left\{\begin{array}{l} x_{k+1}-x_k = hf_1(x_{k+\theta},z_{k+\theta},t_{k+1})+hU(t_{k+\theta}) \\ 0 = f_2(x_{k+1},z_{k+1},t_{k+1}) \\ 0 = h(x_{k+1},z_{k+1},\lambda_{k+1},t_{k+1}), \\ y = g(x_{k+1},z_{k+1},\lambda_{k+1},t_{k+1}) \\ 0 \in \Phi(y_{k+1},\lambda_{k+1},t_{k+1})+N_K(\lambda_{k+1}) \end{array}\right. . \tag{29}$$

## 3.2 Software aspects

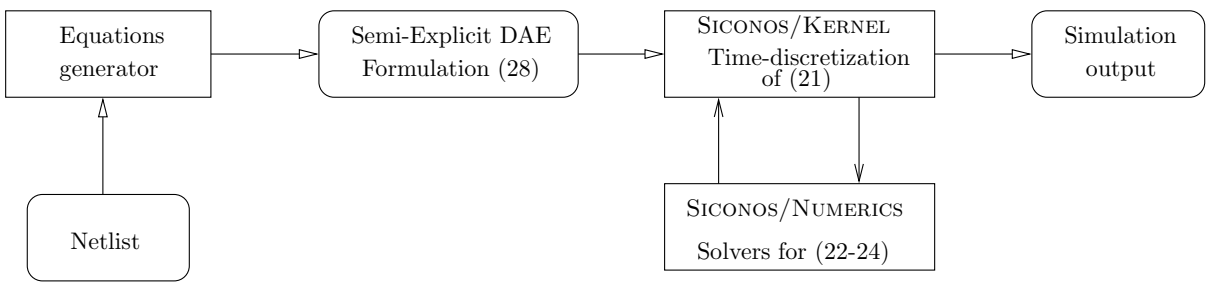



Figure 6: libraries and data.

Fig. 6 shows the libraries and the data involved during the simulation. A Netlist is a circuit textual description used by many simulators like Spice and Eldo. From a Netlist, the automatic generator builds all the components defined in (28). The opensource Siconos/Kernel library performs the time-discretization following the Moreau time–stepping scheme (21) and formulates

at each time–step one instance of the inclusion problem (22-24). The numerical algorithms for the latter problem are in the opensource SICONOS/NUMERICS library. The output of the simulation is a file containing the potential and current values in the SPICE format.

The implementation is object-oriented and mainly in C++. For each electrical component, group of equations and inclusions in (21), a corresponding instance of a class is built. The system is updated in memory at each iteration by the stamp method of each component. In the linear case, these methods are called only once, in the nonlinear case they may be called at any time to update the system. The open-source platform is under GPL license and can be freely used. The equation generator is under private license and can be obtained freely on demand for an academic use.

# 4 An elementary switching circuit

This section is devoted to the modelling and the simulation of the circuit in Fig. 7. In [40] it is shown that Newton-Raphson based methods fail to converge on such a circuit, with the switch model as in (12). The diode model is the equivalent resistor model of Fig. 2 (d). On the contrary the OSNSP solver correctly behaves on the same model.

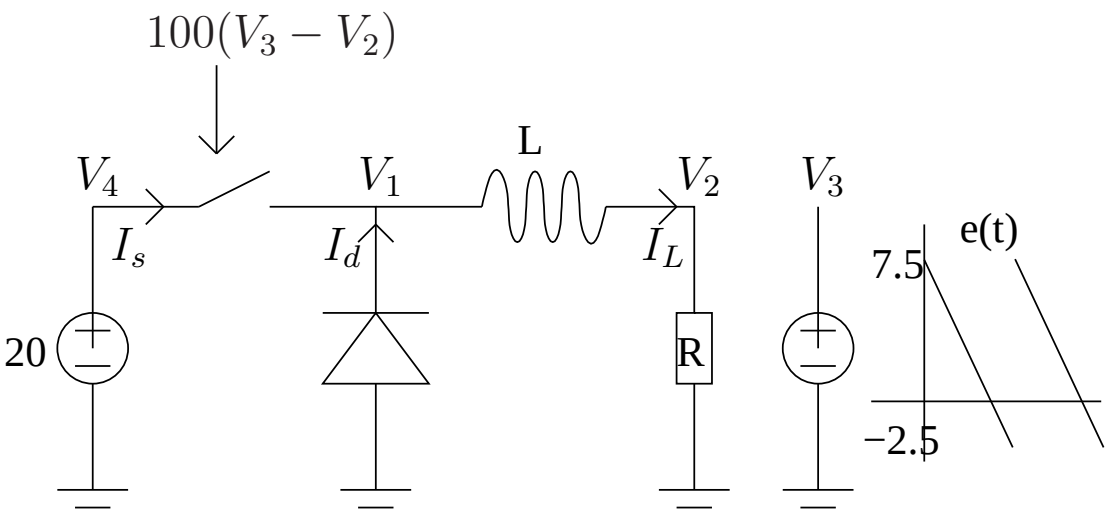



Figure 7: A simple switched circuit.

## 4.1 The dynamical system

The dynamics of the circuit in Fig. 7 is obtained using the algorithm of automatic circuit equation formulation. In a first step, the vector of unknowns is built, in a second step, the dynamical system is written, and, in a last step, the nonsmooth law is added. Applying the automatic equations generation algorithm leads to the following 9-dimensional state vector: $X = (V_1\ V_2\ V_3\ V_4\ I_L\ I_{03}\ I_{04}\ I_s\ I_d)^T$, where the potentials and the currents are depicted on Fig. 7. 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 (20): for the semi–explicit DAE, we obtain

$$\begin{cases} L\frac{dI_L}{dt}(t) = V_1(t) - V_2(t) \\ I_d(t) + I_s(t) - I_L(t) = 0 \\ I_L(t) - \frac{V_2(t)}{R} = 0 \\ I_{03}(t) = 0 \\ I_{04}(t) - I_s(t) = 0 \\ V_4(t) = 20 \\ V_3 = e(t) \end{cases} . \tag{30}$$

For the input/output relations on nonsmooth componenents, we get

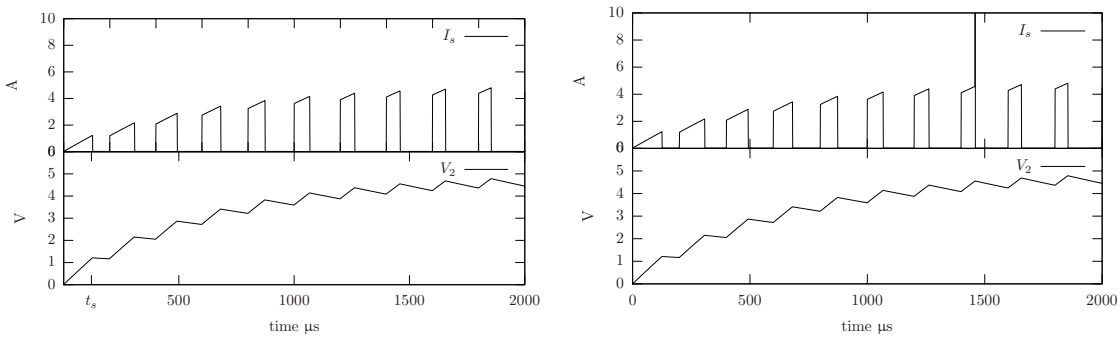
$$\begin{cases} V_1(t) = \frac{1}{2}(\tau_1(t) - 1)R_{\text{off}}I_d(t) - \frac{1}{2}(\tau_1(t) + 1)R_{\text{on}}I_d(t) \\ V_1(t) - V_4(t) = \dfrac{1}{2}(1 + \tau_2(t))R_{\text{off}}I_s(t) \\ \qquad\qquad + \dfrac{1}{2}(1 - \tau_2(t))R_{\text{on}}I_s(t) \end{cases} . \tag{31}$$

Finally, the inclusion rule is written as

$$\begin{cases} V_1(t) \in -N_{[-1,1]}(\tau_1(t)) \\ 100(V_3(t) - V_2(t)) \in -N_{[-1,1]}(\tau_2(t)) \end{cases} . \tag{32}$$

## 4.2 Numerical results with Siconos

The time step has been fixed to 0.1µs. Fig. 8(a) depicts the current evolution through the inductor $L$. In [40], it has been shown that the Newton-Raphson algorithm fails when the state of the diode and of the switch changes at $t = t_s$ in Fig. 8(a). Indeed, 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 a infinite loop without converging. 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 semi-smooth Newton method.




(a) Siconos simulation (b) Eldo simulation

Figure 8: Switched circuit simulations.

**Remark 3** *In [40] 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.*

### 4.3 Numerical results with ELDO

ELDO does not provide any non–smooth switch model. But it furnishes the 'VSWITCH' one described in (33), where $R_S$ is the controlled resistor value of the switch, and $V_C$ the voltage control. Setting $V_{\text{off}}$ to 0, and choosing a small value for $V_{\text{on}}$ lead to:

$$R_S(t) = \begin{cases} R_{\text{on}} & \text{if } V_C(t) \geqslant V_{\text{on}} \\ R_{\text{off}} & \text{if } V_C(t) \leqslant V_{off} \\ (V_C(t)(R_{\text{off}} - R_{\text{on}}) + R_{\text{on}} V_{\text{off}} - & \\ \quad R_{\text{off}} V_{\text{on}})/(V_{\text{off}} - V_{\text{on}}) & \text{otherwise} \end{cases} \tag{33}$$

which is close to (12) 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 Euler with the time step fixed to 0.1µs and $V_{on} = 1e-4$V, $V_{off} = 0$V, $R_{\text{off}} = 1000\Omega$, $R_{\text{on}} = 0.001\Omega$ cause troubles during the ELDO simulation, some messages like 'Newton no-convergence' appear. Fig. 8(b) 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 analog tools can fail to simulate a switched circuit.

## 5 Results on the buck converter

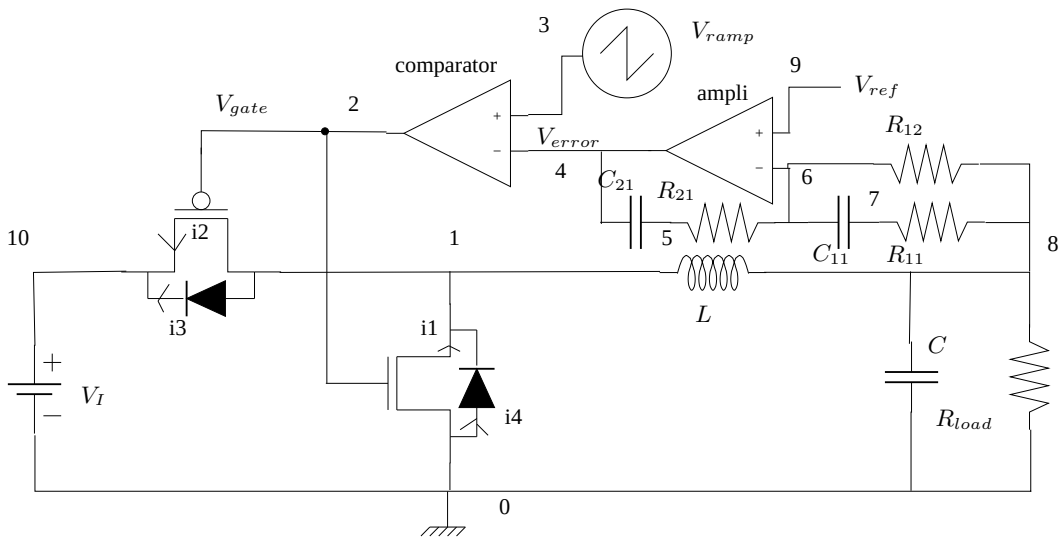



Figure 9: Buck converter

The components are modelled with either linear, or piecewise linear, or set-valued relations yielding a nonsmooth dynamical system of the linear time invariant complementarity systems class. The features of the models are given thereafter.

**Power MOSFETS pMOS/nMOS** they are described as an assembly of a piecewise-linear current source $I_{DS} = f(V_{GS}, V_{DS})$ and the intrinsic diode (DpMOS and DnMOS) with an ideal characteristic. The capacitances were not taken into account. The diodes residual voltage is 1V. The MOSFETs transconductance KP was set to 10AV$^{-2}$ and their threshold voltage to respectively $V_T = -2$V for the pMOS and $V_T = 2$V for the nMOS. One can notice that the sum of their absolute values largely exceeds the supply voltage $V_I = 3$V , thus providing non-overlapping

conduction times. The other physical parameters as chosen as follows : $\mu = 750\ cm^2.\mathrm{V}^{-1}.\mathrm{s}^{-1}$ for a nMOS and $\mu = 250\ cm^2.\mathrm{V}^{-1}.\mathrm{s}^{-1}$ for a pMOS, $\epsilon_{Ox} = \epsilon_{r\ SiO_2} \cdot \epsilon_0$ with $\epsilon_{r\ SiO_2} \approx 3.9$, $t_{OX} \approx 4$nm $W = 130$nm $L = 180$nm.

The piecewise linear model uses 6 segments given by the following data: $c_1 = 0.09, c_2 = 0.2238, c_3 = 0.4666, c_4 = 1.1605, c_5 = 2.8863, a_1 = 0, a_2 = 0.1, a_3 = 0.2487, a_4 = 0.6182, a_5 = 1.5383$. The relative error between $f(\cdot)$ and $f_{\mathsf{pwl}}(\cdot)$ is kept below 0.1 for $0.1 \leqslant x < 3.82$. The absolute error is less than $2 \cdot 10^{-3}$ for $0 \leqslant x < 0.1$ and 0 for negative $x$. In practice, the values of $V_G, V_S, V_D, V_T$ in logic integrated circuits allow a good approximation of $f(\cdot)$ by $f_{\mathsf{pwl}}(\cdot)$.

**Compensator amplifier** it is modelled as a $1.10^5$ gain and an output low-pass filter with a cutoff frequency of 30MHz.

**Comparator** it is modelled as a piecewise-linear function whose value is 0 if $x < -0.15$V and 3 if $x > 0.15$V.

**Ramp voltage** the frequency is 600kHz and the bounds are 0 and $0.75V_I = 2.25$V . The rise time is 1.655ns and the fall time is 10ns.

**Standard values for other components** $V_I = 3\mathrm{V}, L = 10\mu\mathrm{H}, C = 22\mu\mathrm{F}, R_{load} = 10\Omega, R_{11} = 15.58\mathrm{k}\Omega, R_{12} = 227.8\mathrm{k}\Omega, R_{21} = 5.613\mathrm{M}\Omega, C_{11} = 20\mathrm{pF}, C_{21} = 1.9\mathrm{pF}$.

**Values exhibiting a sliding mode** $L = 4\mu\mathrm{H}, C = 10\mu\mathrm{F}, R_{11} = 10\mathrm{k}\Omega, R_{21} = 8\mathrm{M}\Omega, C_{11} = 10\mathrm{pF}$.

The reference voltage Vref rises from 0 to 1.8 V in 0.1 ms at the beginning of the simulation. The output voltage Voutput is regulated to track the reference voltage $V_{ref}$ when $V_I$ or $V_{ref}$ or the load current vary. The error voltage Verror is a filtered value of the difference between Voutput and Vref . This voltage signal is converted into a time length thanks to a comparison with the periodic ramp signal. The comparator drives the pMOS transistor which in turn provides more or less charge to the output depending on the error level. The operation of a buck converter involves both a relatively slow dynamics when the switching elements (MOS and diodes) are keeping their conducting state, and a fast dynamics when the states change. The orders of magnitude are 50ps for some switching details, 1µs for a slow variation period and 100µs at least for a settling period of the whole circuit requiring a simulation.

## 5.1 The dynamical equations

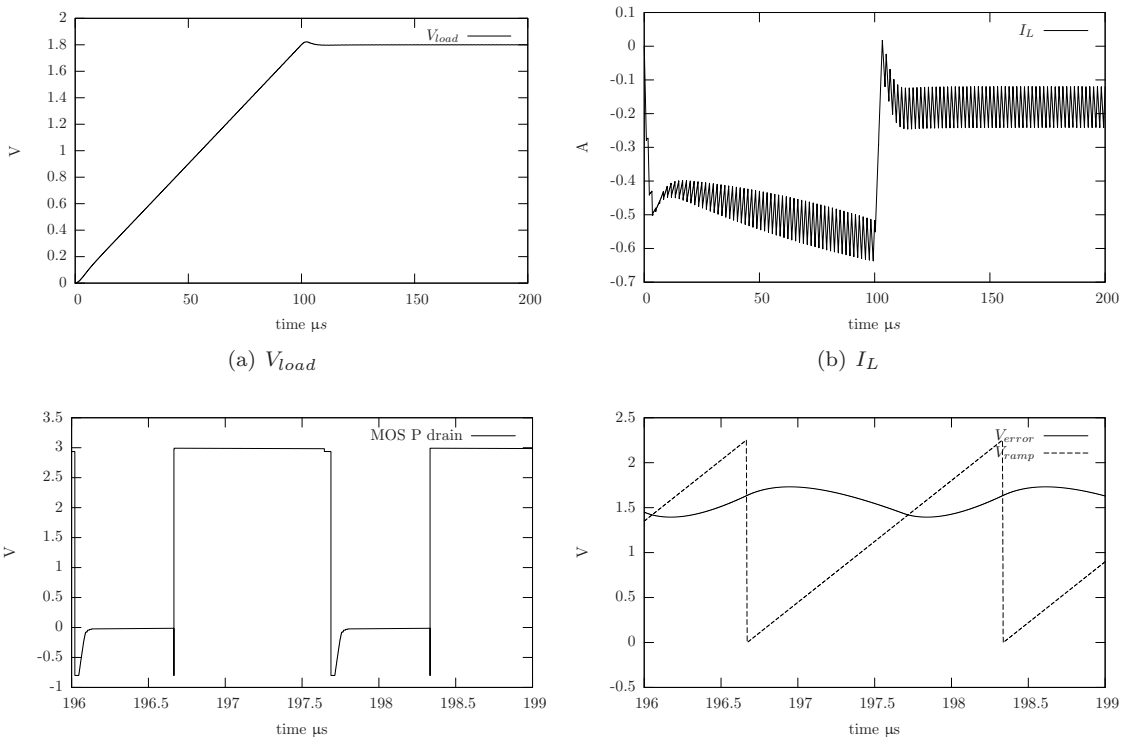
The nonsmooth DAE has been generated using the automatic circuit equation formulation described in Section 3.1. It leads to a dynamical system with 25 states coupled to an inclusion rule. The dimension of the inclusion rule is 24.

## 5.2 Numerical results with Siconos, and comparisons

### 5.2.1 Simulation with Siconos

The start-up of the converter was simulated thanks to Siconos. As initial conditions, all state variables are zeroed. The detailed analysis of the switching events requires to use a time step as small as 50ps. The simulations are carried with a fixed time step, $4.10^6$ steps are then computed for the 200µs long settling of the output voltage. The OSNSP solvers used are PATH with a convergence tolerance of $1e-7$, and a semi-smooth Newton method based on the Fischer-Bursmeister reformulation[22] that is our own implementation using a convergence tolerance of $1e-12$. The overall result is shown on the Fig. 10.

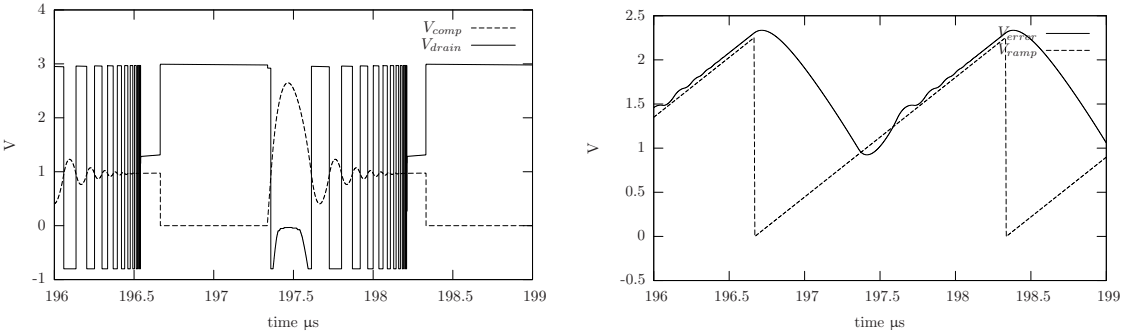
**Simulation time:** The CPU time required to achieve the simulation is 60s on a Pentium 4 processor clocked at 3 GHz. It includes 19s in the MLCP solvers, 40s in matrices products. The time to export the resulting data is not included.




(a) $V_{load}$ (b) $I_L$

(c) pMOS drain potential (d) $V_{ramp}$ and $V_{error}$

Figure 10: SICONOS buck simulation using standard parameters.




(a) $V_{comp}$ and $V_{drain}$ (b) $V_{ramp}$ and $V_{error}$

Figure 11: SICONOS buck simulation using sliding mode parameters.

- Fig. 10 (a) is the output potential, following the ramp $V_{ref}$.
- Fig. 10 (b) is the current through the inductor. Until $0.0001s$, $I_L$ is loading the capacitor C. After $0.0001s$, $I_L$ has to keep the capacitor charge constant.
- Fig. 10 (c) zooms on the pMOS drain potential with standard parameters.
- Fig. 10 (d) zooms on the $V_{error}$ and $V_{ramp}$ voltages.
- Fig. 11 (a) using sliding mode parameters, shows the stabilization of the comparator output to an unsaturated value. It also shows the stabilization of the current through the pMOS allowing the $V_{error}$ signal to follow the $V_{ramp}$ signal.

– Fig. 11 (b) using sliding mode parameters, shows the $V_{error}$ and $V_{ramp}$ voltages.

The simulation has been tested with many parameters values. The robustness of the nonsmooth modelling and solving algorithms enables one to perform with the same CPU time the simulation of such cases.

### 5.2.2 Simulation with Spice

**Simulation conditions: convergence issues related to the MOS model** The simulation of this circuit was done with several versions of Spice (the open source Ngspice from Berkeley and Eldo from Mentor Graphics) and two kinds of MOS models :

**The MOS level 3 model :** This model takes more physical effects into account than the piecewise-linear model used in Siconos simulations, in particular the voltage-dependent capacitances. It is an important issue since these varying capacitances cause some convergence problems when node 2 switches between $V_I$ and ground. Adding a small capacitor of a few picoFarad between this node and ground helps to solve the problem but may yield artifacts (spikes) on the current of the $V_I$ alim and the MOS transistors.

**An nMOS simplified model (Sah model)** with fixed capacitances and a quadratic static characteristic :

$$I_{DS} = \max(0, V_{GS} - Vt_N)^2 - \max(0, V_{GD} - Vt_N)^2$$

This model is very close to the piecewise-linear model used in Siconos simulations. The implementation in netlists was done thanks to voltage-dependent current sources that are very likely not compiled by the various Spice simulators tested. Thus the measured CPU time is increased with respect to a compiled version. An estimation of the CPU time with a compiled simplified model may be given by multiplying the MOS level 3 CPU time by the ratio of the Newton-Raphson iterations required respectively during the simulations with each model. An additional correction should be done to reflect that the computation of the jacobian matrix entries linked to a compiled simplified model would require less time than with a MOS level 3 model. Even if the Spice simulation includes other operations, jacobian matrix loading time is indeed known to be generally predominant.

- Power MOSFETS intrinsic diodes are modelled by the classical Shockley equation with an emission coefficient $N = 1$ :

$$I = I_S.(e^{\frac{q.V}{N.k.T}} - 1) \qquad \text{when} \quad V > -5.N.\frac{k.T}{q}$$

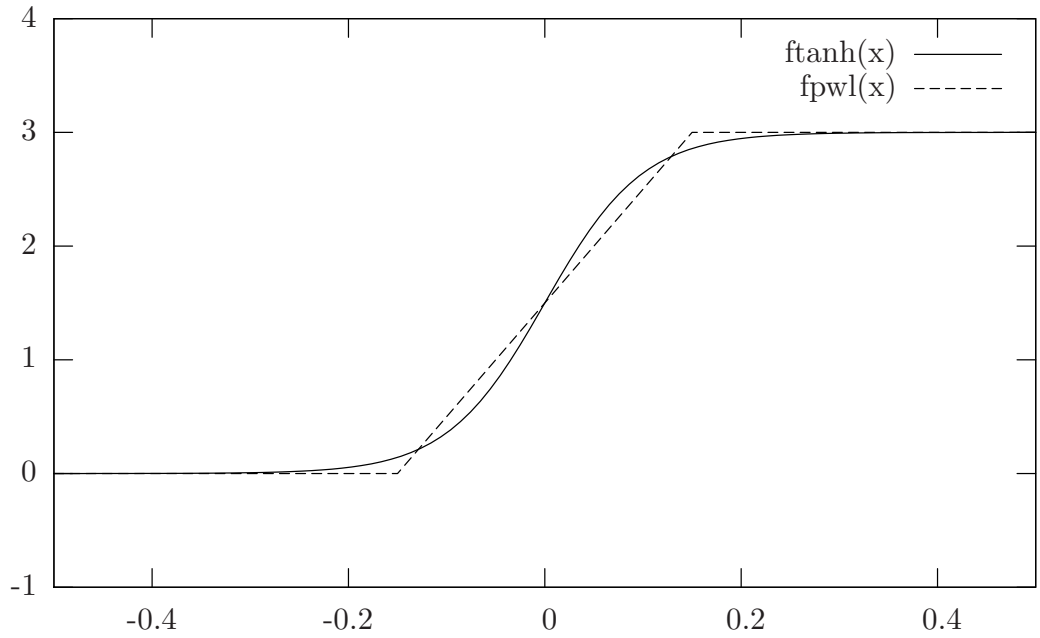
$$I = -I_S \qquad \text{when} \quad V < -5.N.\frac{k.T}{q}$$

with $V$ , $I$ voltage and current through the diode, $I_S$ saturation current, default value $10^{-14}$ A, $q$electron charge $1.6\,10^{-19}$ C, $k$ Boltzmann constant $1.38\,10^{-23}\,J.K^{-1}$, $T$ temperature in K and $N$ emission coefficient.

- The comparator is modelled as a non linear voltage controlled voltage source defined as $V_{out} = 1.5(\tanh(10V_{in}) + 1)$. Thus the 3-segment characteristic used as the nonsmooth model is regularized to help convergence of Spice (see a comparison of the piecewise-linear comparator as used in Siconos simulations with the Spice one on Fig. 5.2.2).

The power supply $V_I$ is raised from 0 in 50 ns at the beginning to help the convergence.[3] The Spice tolerance values used are 1nA for currents, 1µV for voltages and 0.00075 for relative differences. The maximum number of Newton-Raphson iterations is set to 100 (the default values are 10 for Ngspice and 13 for Eldo).

[3]This is not required with the Siconos algorithms that find a consistent initial solution from scratch.




Comparison of piecewise-linear and SPICE (tanh based) comparator models.

Usually, SPICE simulators integrate with a time step adjusted according to different strategies based on an estimation of the local truncation error (LTE) or the number of Newton-Raphson iterations required by previous steps. Since SICONOS simulations were carried with a fixed time step of 50 ps, simulators were forced to use this value as a maximum. Even when SPICE simulators use a fixed time step, they may compute LTE to assess a solution found by the Newton-Raphson algorithm. This computation of LTE was disabled because it could impair the performance of SPICE with respect to SICONOS. [4]

#### 5.2.3 Simulation comparisons

The table 1 displays the results with the standard and the sliding mode values of compensator components. An estimation of the CPU time with a compiled simplified model is added.

These results shall be compared to the 60 s CPU time achieved with the NSDS method. Depending on the model and the SPICE simulator, the (estimated) CPU time is from 2.8 to 6.1 larger than with SICONOS. Moreover, it was necessary to add a parasitic capacitor on the connection between the pMOS and nMOS transistors to allow the convergence of the NGSPICE simulator with the MOS level 3 model. All the SICONOS simulations presented in this paper have been obtained in one-shot from the dynamical equations automatically generated from the Netlist, without any further parameter tuning.

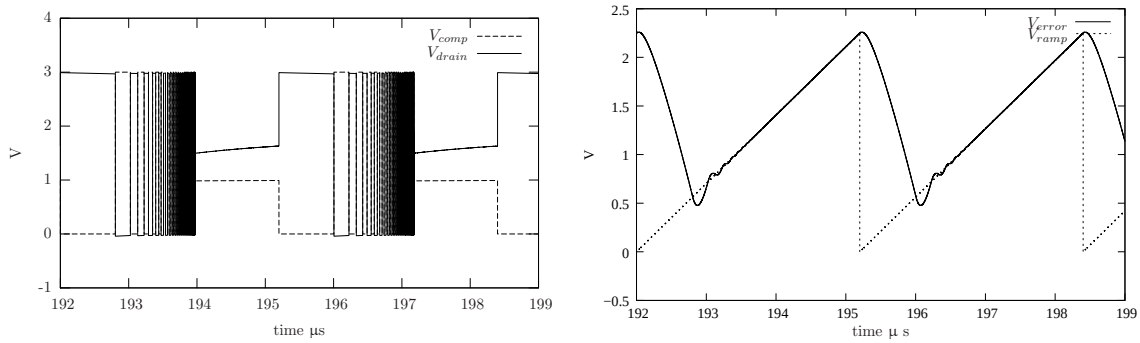
#### 5.2.4 Sliding mode using a multi-valued comparator

This paragraph focuses on the simulation with sliding parameters and using a multi-valued model for the comparator. The rise time of the ramp voltage has been increased to 3.2ns. The model used in SICONOS consists in setting the $\epsilon$ gap to 0 in the model depicted in 19. Fig. 12 shows the

[4] For NGSPICE, it implied a slight modification of the source code since no standard option is provided to do it.

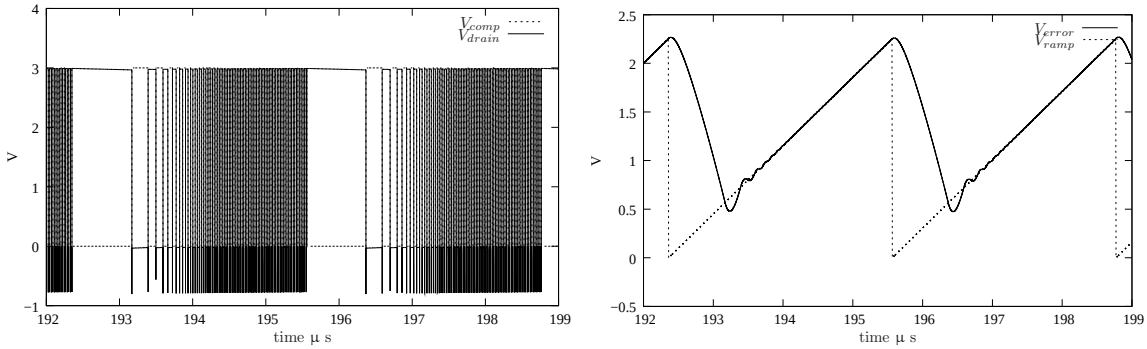
| simulator | model | # Newton iterations | CPU time (s) |
|---|---|---|---|
| **standard compensator values** | | | |
| NGSPICE | simple | 8024814 | 632 |
| NGSPICE | level 3 | 8304237 | 370 |
| ELDO | simple | 4547579 | 388 |
| ELDO | level 3 | 4554452 | 356 |
| SICONOS | LCP | – | 60 |
| **sliding mode compensator values** | | | |
| NGSPICE | simple | 8070324 | 638 |
| NGSPICE | level 3 | 8669053 | 385 |
| ELDO | simple | 5861226 | 438 |
| ELDO | level 3 | 5888994 | 367 |
| SICONOS | LCP | – | 60 |

Table 1: Numerical comparison on the Buck converter Example




(a) $V_{comp}$ and $V_{drain}$ (b) $V_{ramp}$ and $V_{error}$

Figure 12: SICONOS buck simulation using sliding mode parameters and multivalued comparator.




(a) $V_{comp}$ and $V_{drain}$ (b) $V_{ramp}$ and $V_{error}$

Figure 13: ELDO buck simulation using sliding mode parameters and $V_{out} = 1.5(\tanh(10000V_{in})+1)$ for the comparator.

SICONOS simulation using a fully implicit time-stepping. It could be noted that the comparator output is stabilized to an unsaturated value. Simulation using ELDO has been done using the model $V_{out} = 1.5(\tanh(10000V_{in}) + 1)$ for the comparator. The MOS level 3 leads to 'Newton no-convergence' messages, so the MOS SAh model has been used to run the simulation displayed

in Fig. 13. It is noteworthy that it does not handle the stabilization of the comparator output on the sliding surface.

#### 5.2.5 Simulation with Plecs

As we pointed out above, the hybrid approach that consists of an exhaustive enumeration of all the system's modes, soon become inefficient and unusable mainly because the simulation duration grows exponentially fast. Let us illustrate this fact with the buck converter, loaded with several devices: a resistance, and a chain of transistors. The simulator is Plecs, a hybrid simulator developed by Plexim [5].

The CPU time required to achieve the simulation of 200µs ranges between 596s to 4 hours, depending on the values of the resistors, capacitors and inductors and the existence of sliding modes. This should be compared to the 60s of the Siconos simulation, obtained independently of these components values. Moreover, the Plecs simulation performs only 168038 steps comparing the $4.10^6$ steps performed during the Siconos simulation. It can be concluded that the computation of one step of simulation using Siconos is 250 faster than using Plecs. This demonstrates the robustness and efficiency of the time-stepping scheme and the OSNSP algorithms of Siconos.

**Remark 4** *On both Fig.12(a) and Fig. 11(b) it is seen that the sliding surface is attained in finite time after an accumulation of switches. This is a classical phenomenon in nonsmooth systems, see Filippov's example in [3].*

## 6 Conclusions

In this paper we have presented numerical simulations of switched circuits obtained with a suitable time-stepping implicit method, named Moreau's time-stepping algorithm. This method is based on the nonsmooth dynamical systems modelling approach, and relies heavily on complementarity problems (equivalently, inclusions into normal cones) solvers. The advantages of such a method are that it allows one to:

- avoid computing the dynamics changes due to topology variations, since the circuits are treated as a global system with a fixed state dimension; modes transitions are taken care of by the complementarity problem solvers, which usually are polynomial in time;
- simulate circuits with very large number of events without slowing down too much the simulation;
- avoid regularization and consequently stiff systems of ODEs;
- accurately calculate the initial steady-state of the system;
- accurately simulate sliding mode trajectories without spurious oscillations around the switching surface;
- compute state jumps (initial jumps due to inconsistent states, or in the course of the integration).

The major drawback of the used method is its low order, so that its accuracy may be less good on smooth portions of the trajectoires. In this paper it is first shown that Moreau's time-stepping scheme allows one to integrate an academic example on which Newton-Raphson based methods fail. Then the buck converter system is simulated. Comparisons with other analog simulators are presented. The simulations have been led with the Siconos software package of the INRIA, an open source platform dedicated to nonsmooth multivalued dynamical systems.

[5] http://www.plexim.com/

## References

[1] V. Acary, O. Bonnefon, and B. Brogliato. Improved circuit simulator. Patent number 09/02605, May 2009.

[2] V. Acary and B. Brogliato. *Numerical Methods for Nonsmooth Dynamical Systems. Applications in Mechanics and Electronics*, volume 35 of *Lecture Notes in Applied and Computational Mechanics*. Springer Verlag, Heidelberg, 2008.

[3] V. Acary and B. Brogliato. Implicit Euler numerical simulations of sliding mode systems. Technical Report 6886, INRIA, `http://hal.inria.fr/inria-00374840/fr/`, 2009. submitted to Systems and Control Letters.

[4] V. Acary and F. Pérignon. An introduction to SICONOS. Technical Report TR-0340, INRIA, http://hal.inria.fr/inria-00162911/en/, 2007.

[5] V. Acary and F. Pérignon. SICONOS: a software platform for modeling, simulation, analysis and control of nonsmooth dynamical systems. *Simulation News Europe*, 17(3/4):19–26, December 2007.

[6] K. Addi, S. Adly, B. Brogliato, and D. Goeleven. A method using the approach of Moreau and Panagiotopoulos for the mathematical formulation of non-regular circuits in electronics. *Nonlinear Analysis: Hybrid Systems*, 1(1):30–43, March 2007.

[7] K. Addi, B. Brogliato, and D. Goeleven. A qualitative mathematical analysis of a class of linear variational inequalities via semi-complementarity problems: applications in electronics. *Mathematical Programming A*, 2008. in press, DOI 10.1007/s10107-009-0268-7.

[8] C. Batlle, E. Fossas, and A. Miralles. Generalized discontinuous conduction modes in the complementarity formalism. *IEEE Transactions on Circuits and Systems–II: Express Briefs*, 52(8):447–451, August 2005.

[9] D. Bedrosian and J. Vlach. Analysis of switched networks. *Int. Journal of Circuit Theory and Applications*, 20:309–325, 1992.

[10] S.C. Billups. *Algorithms for Complementarity Problems and Generalized Equations*. PhD thesis, University of Wisconsin, Madison, 1995. available as Technical Report MP-TR-1995-14.

[11] S.C. Billups, S.P. Dirkse, and M.C. Ferris. A comparison of large scale mixed complementarity problem solvers. *Computational Optimization and Applications*, 7:3–25, 1997.

[12] S.C. Billups and M.C. Ferris. QPCOMP: A quadratic programming based solver for mixed complementarity problems. Technical Report MP-TR-1995-09, University of Wisconsin, Madison, 1995.

[13] D. Biolek and J. Dobes. Computer simulation of continuous-time and switched circuits: limitations of SPICE-family programs and pending issues. In *Radioelektronika, 17th Int. Conference*, Brno, Czech Republic, 24-25 April 2007.

[14] M. K. Camlibel, W.P.M.H. Heemels, and J.M. Schumacher. Consistency of a time-stepping method for a class of piecewise-linear networks. *IEEE Trans. Circuits and systems I*, 49:349–357, 2002.

[15] M. Cao and M.C. Ferris. A pivotal method for affine variational inequalities. *Mathematics of Operations Research*, 21(1):44–64, 1996.

[16] C. Chen and O. L. Mangasarian. A class of smoothing functions for nonlinear and mixed complementarity problems. *Computational Optimization and Applications*, 5:97–138, 1996.

[17] L.O. Chua and A.C. Dang. Canonical piecewise-linear analysis: Part II – tracing driving-point and transfer characteristics. *IEEE Transactions on Circuits and Systems*, CAS-32(5):417–444, May 1985.

[18] L.O. Chua and R.L.P. Ying. Canonical piecewise-linear analysis. *IEEE Transactions on Circuits and Systems*, CAS-30(3):125–140, March 1983.

[19] H.S.H Chung and A. Ioinovici. Fast computer aided simulation of switching power regulators based on progressive analysis of the switches'state. *IEEE Transactions on Power Electronics*, 9(2):206–212, March 1994.

[20] R. W. Cottle, J. Pang, and R. E. Stone. *The Linear Complementarity Problem*. Academic Press, Inc., Boston, MA, 1992.

[21] T. De Luca, F. Facchinei, and C. Kanzow. A theoritical and numerical comparison of some semismooth algorithms for complementarity problems. *Computational Optimization and Applications*, 16:173–205, 2000.

[22] T. DeLuca, F. Facchinei, and C. Kanzow. A semismooth equation approach to the solution of nonlinear complementarity problems. *Mathematical Programming*, 75:407–439, 1996.

[23] P. Denoyelle and V. Acary. The non-smooth approach applied to simulating integrated circuits and power electronics. Evolution of electronic circuit simulators towards fast-SPICE performance. *INRIA Research Report 0321,* `http://hal.inria.fr/docs/00/08/09/20/PDF/RT-0321.pdf`, 2006.

[24] S. P. Dirkse and M. C. Ferris. The PATH solver: A non-monotone stabilization scheme for mixed complementarity problems. *Optimization Methods and Software*, 5:123–156, 1995.

[25] H. Elmqvist, S.E. Mattsson, and M. Otter. Object–oriented and hybrid modling in Modelica. *Joural Européen de systèmes automatisés*, 35(4):395–404, 2001.

[26] O. Enge and P. Maisser. Modelling electromechanical systems with electrical switching components using the linear complementarity problem. *Multibody System Dynamics*, 13:421–445, 2005.

[27] F. Facchinei and J. S. Pang. *Finite-dimensional Variational Inequalities and Complementarity Problems*, volume I & II of *Springer Series in Operations Research*. Springer Verlag NY. Inc., 2003.

[28] R. Frasca, M.K. Camlibel, I.C. Goknar, and F. Vasca. State jump rules in linear passive networks with ideal switches. In *IEEE Int. Symp. on Circuits and Systems (ISCAS)*, Seattle, Washington, May 2008.

[29] M. Fukushima. Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems. *Mathematical Programming*, 53:99–110, 1992.

[30] S.A Gabriel and J.S. Pang. An inexact NE/SQPmethod for solving the nonlinear complementarity problem. *Computational Optimization and Applications*, 1:67–91, 1992.

[31] Z. Galias and X. Yu. Complex discretization behaviors of a simple sliding-mode control system. *IEEE Transactions on Circuits and Systems – II: Express Briefs*, 53(8):652–656, August 2006.

[32] D. Giaouris, S. Banerjee, B. Zahawi, and V. Pickert. Stability analysis of the continuous-conduction-mode buck converter via Filippov's method. *IEEE Transactions on Circuits and Systems–I: Regular Papers*, 55(4):1084–1096, May 2008.

[33] C. Glocker. Models of non-smooth switches in electrical systems. *Int. J. of Circuit Theory and Applications*, 33:205–234, 2005.

[34] D. Goeleven. Existence and uniqueness for a linear mixed variational inequality arising in electrical circuits with transistors. *Journal of Optimization Theory and Applications*, 138(3):397–406, September 2008.

[35] J.B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms.* Springer Verlag, Heidelberg, 1993. (two volumes).

[36] M. Jean. The non-smooth contact dynamics method. *Comput. Methods Appl. Mech. Eng.*, 177(3-4):235–257, 1999.

[37] S.M. Kang and L.O. Chua. A global representation of multidimensional piecewise-linear functions with linear partitions. *IEEE Transactions on Circuits and Systems*, CAS-25(11):938–940, November 1978.

[38] D.M.W. Leenaerts and W.M.G. van Bokhoven. *Piecewise Linear Modeling and Analysis.* Kluwer, New-York, 1998.

[39] D.M. Leenarts. On linear dynamic complementarity systems. *IEEE Transactions on Circuits and Systems–I: Fundamental Theory and Applications*, 46(8):1022–1026, August 1999.

[40] P. Maffezzoni, L. Codecasa, and D. D'Amore. Event-driven time-domain simulation of closed-loop switched circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(11):2413–2426, November 2006.

[41] K. Mayaram, D.C. Lee, D.A. Moinian, and J. Roychowdhury. Computer-aided circuit analysis tools for RFIC simulation: algorithms, features, and limitations. *IEEE Transactions on Circuits and Systems–II: Analog and Digital Signal Processing*, 47(4):274–286, April 2000.

[42] M. Moeller and C. Glocker. Non-smooth modelling of electrical systems using the flux aproach. *Nonlinear Dynamics*, 50:273–295, 2007.

[43] J.J. Moreau. Evolution problem associated with a moving convex set in a Hilbert space. *Journal of Differential Equations*, 26:347–374, 1977.

[44] J.J. Moreau. *Unilateral contact and dry friction in finite freedom dynamics*, volume 302 of *CISM Courses and Lectures*, chapter 1, pages 1–82. Springer Verlag, International Center for Mechanical Sciences, 1988.

[45] J.J. Moreau. Numerical aspects of the sweeping process. *Comput. Methods Appl. Mech. Eng.*, 177(3-4):329–349, 1999.

[46] J.S. Pang and S.A Gabriel. NE/SQP: A robust algorithm for the nonlinear complementarity problem. *Mathematical Programming*, 60:295–338, 1993.

[47] R.T. Rockafellar. *Convex Analysis.* Princeton University Press, 1970.

[48] T. Rutherford. Miles: A mixed inequality and nonlinear equation solver, 1993.

[49] R.W.H. Sargent. An efficient implementation of the Lemke algorithm and its extension to deal with upper an lower bounds. *Mathematical Programming Study*, 7:36–54, 1978.

[50] S.N. Stevens and P.M. Lin. Analysis of piecewise-linear resistive networks using complementarity pivot theory. *IEEE Transactions on Circuits and Systems*, CAS-28(5):429–441, May 1981.

[51] W.M.G. van Bokhoven and J.A.G. Jess. Some new aspects of $P$ and $P_0$ matrices and their application to networks with ideal diodes. In *IEEE Int. Symp. Circuits and Systems*, pages 806–810, 1978.

[52] L. Vandenberghe, B.L. De Moor, and J. Vandewalle. The generalized linear complementarity problem applied to the complete analysis of resistive piecewise-linear circuits. *IEEE Transactions on Circuits and Systems*, 36(11):1382–1391, November 1989.

[53] F. Vasca, L. Iannelli, M.K. Camlibel, and R. Frasca. A new perspective for modelling power electronics converters: complementarity framework. *IEEE Transactions on Power Electronics*, 24(2):456–468, February 2009.

[54] J. Vlach and A. Opal. Modern CAD methods for analysis of switched networks. *IEEE Transactions on Circuits and Systems–I: Fundamental Theory and Applications*, 44(8):759–762, August 1997.

[55] J. Vlach, J.M. Wojciecchowski, and A. Opal. Analysis of nonlinear networks with inconsistent initial conditions. *IEEE Transactions on Circuits and Systems–I: Fundamental Theory and Applications*, 42(4):195–200, April 1995.

[56] F. Yuan and A. Opal. Computer methods for switched circuits. *IEEE Transactions on Circuits and Systems–I: Fundamental Theory and Applications*, 50(8):1013–1024, August 2003.

[57] D.L. Zhu and P. Marcotte. Modified descents methods for solving the monotone variational inequality problem. *Operations Research Letters*, 14:111–120, 1993.

[58] D.L. Zhu and P. Marcotte. An extended descent framework for monotne variational inequalities. *Journal of Optimization Theory and Applications*, 80:349–366, 1994.

[59] L. Zhu and J. Vlach. Analysis and steady state of nonlinear networks with ideal switches. *IEEE Transactions on Circuits and Systems–I: Fundamental Theory and Applications*, 42(4):212–214, April 1995.

Centre de recherche INRIA Grenoble – Rhône-Alpes
655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399