A MLD Model Development of a Boost Converter

Mihaela Sbarciog
EeSA - Department of Electrical Energy, Systems and Automation
Ghent University
Technologiepark 913, Ghent 9052, Belgium
Email: mihaela@autoctrl.UGent.be

Robin De Keyser
EeSA - Department of Electrical Energy, Systems and Automation
Ghent University
Technologiepark 913, Ghent 9052, Belgium
Email: rdk@autoctrl.UGent.be

Abstract—This work presents the development of a MLD (Mixed Logical Dynamical) model for a boost converter. The boost converter, as well as any other switched device in power electronics, is a system suited to be characterized as a MLD system due to the presence of the switch, introducing a binary input to the system, and the dynamical evolution depending on if-then-else rules. The classical approach to describe switched-mode power electronic devices is via "averaging" techniques. A comparison of the simulation results obtained with two models, hybrid and real, is also presented.

Index Terms—DC-DC converters, hybrid systems, modelling

I. INTRODUCTION

DC-DC converters have been the object of sustained interest in the last decades. They are extensively used in power supplies for electronic equipment in order to control the energy flow between two DC systems. Although a typical DC-DC converter circuit requires few components and, from a theoretical point of view, is simplistic to operate, all DC-DC converters require control circuitry in order to account for load variations, component tolerances, system aging and input source voltage variations.

Due to their time-varying characteristic and to their inherent nonlinear feature, DC-DC converters are a traditional benchmark for testing advanced controllers. Lately, the emerging control strategies compute the control signal, in order to improve the overall quality of control, based on a model of the process.

The modelling and control of DC-DC converters can be formulated in two different ways, depending on whether the switching signal \( q(t) \) (Fig. 1) is directly manipulated - hybrid system approach - or an auxiliary pulse width modulation (PWM) circuit is necessary in order to determine the switch position. While many control applications are available in literature for the latter case, based on the availability of the averaged model, few results are available in the case of hybrid systems to which the MLD (Mixed Logical Dynamical) formulation belongs. Among the PWM-based control strategies reported in literature to provide improved closed loop behavior and disturbance rejection, the following methods can be listed: sliding mode control strategies [1], nonlinear PI controllers based on the method of extended linearization [2], nonlinear \( H_{\infty} \) controllers [3], passivity-based controller [4] and model-based predictive controller [5], [6].

On the other hand, MLD formalism is a versatile approach of describing hybrid systems, allowing to state and solve control problems in a systematic way, using for example a model-based predictive control strategy. Real-life applications that can be naturally modelled within the MLD framework are presented in [7], [8], [9] and [10].

In this work a MLD model for a boost converter is developed, the control - in the predictive control framework - being approached in a subsequent publication. The paper is structured as follows. The next section describes the DC-DC system dynamics. Section 3 deals with the MLD formalism, while Section 4 provides a step-by-step development of the boost converter MLD model. A comparison between the simulation results obtained by using the MLD model and the real model is given in Section 5. Finally, some conclusions are drawn.

II. PROCESS DESCRIPTION

The boost converter, represented in Fig. 1, is a "step-up" converter, which produces a higher DC output voltage \( V_2 \) than the one supplied by the external voltage source \( V_0 \). The differential equations describing the circuit are given by:

\[
\dot{x}_1(t) = - (1 - q(t)) \frac{1}{L} x_2(t) + \frac{1}{L} V_0 g(x_1(t), q(t))
\]

\[
\dot{x}_2(t) = (1 - q(t)) \frac{1}{C} x_1(t) - \frac{1}{RC} x_2(t)
\]

(1)

where

\[
g(x_1(t), q(t)) = \begin{cases} 
1, & \text{if } q(t) = 1 \\
1, & \text{if } q(t) = 0 \text{ and } x_1(t) \neq 0 \\
0, & \text{if } q(t) = 0 \text{ and } x_1(t) = 0 
\end{cases}
\]

(2)
The variables appearing in model (1) are defined as follows:

- \( x_1(t) \) - input inductor current \( i_L(t) \);
- \( x_2(t) \) - output capacitor voltage \( V_d(t) \);
- \( V_0 \) - nominal value of the external voltage source;
- \( R \) - nominal value of the output resistance;
- \( q(t) \) - switch position function, which takes values in the discrete set \{0, 1\}.

As model (1) shows, the dynamic behavior of the boost converter is driven by the switch position with a binary value and evolves according to if-then-else rules summarized in (2), the boost converter being classified in this way in the hybrid systems category.

Before proceeding further with the development of the MLD model, a short analysis of the boost converter behavior is considered. When the switch is ON \((q(t) = 1)\), the input inductor current \( i_L \) increases and the capacitor \( C \) discharges on the resistor \( R \) according to the relations:

\[
L \frac{di_L}{dt} = V_g \\
C \frac{dV_d}{dt} = -\frac{V_d(t)}{R}
\]

In this situation the converter is said to function in state \( S_1 \). Alternatively, when the switch is OFF \((q(t) = 0)\), the variations of the input inductor current \( i_L \) and of the output capacitor voltage \( V_d \) are described respectively by:

\[
L \frac{di_L}{dt} = V_g - V_d(t) \\
C \frac{dV_d}{dt} = i_L - \frac{V_d(t)}{R}
\]

and the operational state is denoted by \( S_2 \). While working in state \( S_2 \), the current \( i_L \) will constantly decrease and, possibly, will become negative. However the presence of the diode in the circuit represented in Fig. 1 prevents that the current flows in the opposite direction, thus \( i_L \) is limited to zero and a transition to state \( S_3 \) takes place.

Consequently, the operational states of the boost converter can be rigourously defined, in terms of state variables and switching function, as:

\[
S_1 : \quad q(t) = 1 \\
S_2 : \quad q(t) = 0 \quad \& \quad x_1(t) \neq 0 \\
S_3 : \quad q(t) = 0 \quad \& \quad x_1(t) = 0
\]

and the functioning of the DC-DC system is described by the finite state automaton illustrated in Fig. 2.

The transitions between the states can occur only in the order \( S_1 - S_2 - S_3 - S_1 \) or \( S_1 - S_2 - S_1 \), a direct transition from \( S_1 \) to \( S_3 \) being impossible due to the fact that in \( S_1 \) the input inductor current is continuously increasing, at the end of this state reaching its highest value, and will never become instantaneously zero when the switch is open to allow for a transition to \( S_2 \). Similarly, a transition from \( S_3 \) to \( S_2 \) is not possible, the inductor current can not decrease more (phenomenon characteristic to state \( S_2 \)) starting from its zero value in state \( S_3 \). However, it can happen that the system does not enter state \( S_3 \), if the switch is not kept open a long enough time.
"x(t) > 0" or "Temperature is hot", referred to as literals that can have either a true or a false value. Logic propositions are obtained by combining literals in statements, by means of connectives: "∧" - and, "∨" - or, "¬" - not, "⇒" - implies, "⇔" - equivalent, "⊕" - exclusive or. Connectives satisfy several properties that allow transformation of logic statements into equivalent statements involving different connectives and simplify complex statements. Thus the propositional logic governing the dynamic evolution of a system can be formulated as a collection of statements of the form:

\[
X_1 \lor X_2 \Rightarrow X_3
\]  
(8)

Basically, the derivation of the MLD form of a hybrid system involves three steps [7], [9]. The first step is to associate with each literal \(X_i\) a logical variable \(\delta_i \in \{0, 1\}\) (\(\delta_i = 1\) if \(X_i\) is true or \(\delta_i = 0\) otherwise). Consequently, complex statements resulted from combinations of elementary statements via the boolean operators introduced above can be represented as linear inequalities over the corresponding binary variables \(\delta_i\). Some basic equivalent translations of logic propositions into linear constraints are given in Table I. On one hand these relations can be extended to involve an arbitrary number of literals/logical variables, on the other hand new entries can be appended to Table I to cover the equivalency between more complex logic propositions and their corresponding integer inequalities.

The second step consists in representing the product between linear functions and logical variables by introducing an auxiliary variable \(z = \delta a^T x\). The mixed integer inequalities corresponding to the auxiliary real variables involve the lower and upper bounds on continuous quantities:

\[
M = \max_{x \in R(a^T x)}, m = \min_{x \in R(a^T x)} \quad (9)
\]

and a small tolerance \(\epsilon\), typically the machine precision, introduced to replace the strict inequalities by non-strict ones. Auxiliary real variables are uniquely specified by the mixed integer linear inequalities listed on the last row of Table I.

The third step is to include auxiliary binary and continuous variables defined above in a LTI discrete-time dynamic system in order to define, in a unified model, the evolution of the continuous and logic components of the system. The general MLD form of a hybrid system is

\[
x(k + 1) = Ax(k) + B_1 u(k) + B_2 \delta(k) + B_3 z(k)
\]

\[
y(k) = Cx(k) + D_1 u(k) + D_2 \delta(k) + D_3 z(k) \quad (10)
\]

subjected to the inequality constraint

\[
E_2 \delta(k) + E_3 z(k) \leq E_1 u(k) + E_4 x(k) + E_5 \quad (11)
\]
where \( x \in \mathbb{R}^{n_c} \times \{0, 1\}^{n_l} \) are the continuous and binary states, \( u \in \mathbb{R}^{m_c} \times \{0, 1\}^{m_l} \) are the inputs, \( y \in \mathbb{R}^{n_r} \times \{0, 1\}^{n_l} \) are the outputs and \( \delta \in \{0, 1\}^{n_\delta} \), \( z \in \mathbb{R}^{n_z} \) represent respectively auxiliary binary and continuous variables. The matrix inequality (11) comprises all the constraints imposed on the states, inputs and \( z \) and \( \delta \) variables. Although the description (10)-(11) seems to be linear (in the state variables \( x \) and auxiliary binary and continuous variables \( \delta \) and \( z \)), system nonlinearity is hidden in the inequality constraints over the binary variables.

IV. BOOST CONVERTER MLD MODEL DEVELOPMENT

Once the continuous and discrete variables have been identified and the logic governing the process functioning has been stated, one can proceed developing the MLD model of the system. The boost converter described by (1) and ruled by (2) has two continuous variables - the states of the system - namely the input inductor current and the output capacitor voltage and one binary variable - the system input - that is the switch position.

In order to apply the three-steps procedure for MLD model development presented in the previous section, literals are associated first with statements derived from (2) as follows:

\[
X_1 : \ g(t) = 1 \quad (12)
\]

\[
X_2 : \text{"System in state } S_2\text{"} \quad (13)
\]

\[
X_3 : \text{"System in state } S_3\text{"} \quad (14)
\]

\[
X_4 : \ x_1 \neq 0 \quad (15)
\]

While (12) and (15) introduce basic statements that can not be represented by combination of any other meaningful statements, (13) and (14) can be decomposed in terms of the literals \( X_1 \) and \( X_4 \), according to the system functioning principles detailed in Section II and summarized by (7), as:

\[
X_2 \Leftrightarrow \sim X_1 \land X_4 \quad (16)
\]

\[
X_3 \Leftrightarrow X_1 \land \sim X_4 \quad (17)
\]

The propositional logic expressions involving the literals \( X_i \) can be translated into a mathematical representation by associating to each literal a binary variable \( \delta_i \in \{0, 1\} \) (the first step of the procedure). If \( X_i \) is true, then \( \delta_i = 1 \), else \( \delta_i = 0 \). With this association, the logic expressions (16)-(17) become in terms of integer variables:

\[
[\delta_2 = 1] \iff \ [\delta_1 = 0] \land [\delta_4 = 1] \quad (18)
\]

\[
[\delta_3 = 1] \iff \ [\delta_1 = 0] \land [\delta_4 = 0] \quad (19)
\]

Note that literals where replaced by their associated binary variables in the logic propositions in order to illustrate the smooth transition from logic to integer variables.

Further (18) and (19) are translated into two sets of mixed integer inequalities (Table I) as follows:

\[
\begin{align*}
\delta_2 & \leq 1 - \delta_1 \\
\delta_2 & \leq \delta_4 \\
\delta_2 & \geq (1 - \delta_1) + \delta_4 - 1
\end{align*}
\]

(20)

\[
\begin{align*}
\delta_3 & \leq 1 - \delta_1 \\
\delta_3 & \leq 1 - \delta_4 \\
\delta_3 & \geq (1 - \delta_1) + (1 - \delta_4) - 1
\end{align*}
\]

(21)

In light of the newly introduced concepts and transformations, the function \( g \) in (2), selecting the operational state of the converter, can be rewritten as:

\[ g(x_1(t), q(t)) = \delta_1 + (\delta_2 + \delta_3)\delta_4 \quad (22) \]

Combination (22) has been determined according to the following two observations:

1) At a certain time instant \( t \), the boost converter can be in only one of the possible three states, this condition translating - in terms of integer variables - into:

\[ \delta_1 + \delta_2 + \delta_3 = 1 \quad (23) \]

2) Out of the two variables (input inductor current and switch position) used for describing states \( S_2 \) and \( S_3 \), only the value of the inductor current (consequently the logical variable \( \delta_1 \)) is useful for discriminating between the two states.

Assuming the discrete-time representation of the boost converter, obtained by sampling the continuous model (1) with a small sampling period \( T \), making use of the knowledge stated by relationship (22) and keeping in mind that \( \delta_1 \) is the logical variable associated with the switch position, the model of the boost converter becomes:

\[
x_1(k+1) = x_1(k) - \frac{T}{L} x_2(k) + \frac{T}{L} x_2(k) \delta_1(k) + \frac{TV_g}{L} \left[ \delta_1(k) + [\delta_2(k) + \delta_3(k)] \delta_4(k) \right]
\]

\[
x_2(k+1) = \frac{T}{C} x_1(k) - \frac{T}{C} x_1(k) \delta_1(k) + \left( 1 - \frac{T}{RC} \right) x_2(k)
\]

(24)

where \( k \) is an integer number of sampling periods.

Expanding model equations (24) one can immediately notice the occurrence of product terms of logical variables and of continuous and logical variables. Procedures to transform these product terms exist, however they introduce auxiliary variables [7], [12].
Four new logical variables are introduced, with the following definitions:
\[
\delta_5 = \delta_2 \cdot \delta_4 \tag{25} \\
\delta_6 = \delta_3 \cdot \delta_4 \tag{26} \\
\delta_7 = \ldots = \tag{27} \\
\delta_8 = \delta_1 \cdot \delta_6 \tag{28}
\]
and the associated propositional logic:
\[
[\delta_5 = 1] \iff [\delta_2 = 1] \land [\delta_4 = 1] \tag{29} \\
[\delta_6 = 1] \iff [\delta_3 = 1] \land [\delta_4 = 1] \tag{30} \\
[\delta_7 = 1] \iff [\delta_1 = 1] \land [\delta_5 = 1] \tag{31} \\
[\delta_8 = 1] \iff [\delta_1 = 1] \land [\delta_6 = 1] \tag{32}
\]
which transforms to linear mixed integer inequalities:
\[
\begin{aligned}
&-\delta_2 + \delta_5 \leq 0 \\
&-\delta_4 + \delta_5 \leq 0 \\
&\delta_2 + \delta_1 - \delta_5 \leq 1 \\
&-\delta_3 + \delta_6 \leq 0 \\
&-\delta_4 + \delta_6 \leq 0 \\
&\delta_3 + \delta_1 - \delta_6 \leq 1 \\
&-\delta_1 + \delta_7 \leq 0 \\
&-\delta_7 + \delta_1 \leq 0 \\
&\delta_1 + \delta_5 - \delta_7 \leq 1 \\
&-\delta_1 + \delta_6 \leq 0 \\
&-\delta_6 + \delta_8 \leq 0 \\
&\delta_1 + \delta_6 - \delta_8 \leq 1
\end{aligned} \tag{33}
\]
Aside the four new logical auxiliary variables, seven auxiliary continuous variables are defined (second step of the procedure):
\[
\begin{aligned}
z_1(k) &= x_1(k)\delta_1(k) \\
z_2(k) &= x_1(k)\delta_2(k) \\
z_3(k) &= x_1(k)\delta_3(k) \\
z_4(k) &= x_2(k)\delta_4(k) \\
z_5(k) &= x_1(k)\delta_5(k) \\
z_6(k) &= x_2(k)\delta_6(k) \\
z_7(k) &= x_2(k)\delta_8(k)
\end{aligned} \tag{37}
\]
which satisfy:
\[
[\delta_1 = 0] \implies [z_1 = 0] , \quad [\delta_1 = 1] \implies [z_1 = x_1] \\
[\delta_5 = 0] \implies [z_2 = 0] , \quad [\delta_5 = 1] \implies [z_2 = x_1] \\
\vdots \\
[\delta_8 = 0] \implies [z_7 = 0] , \quad [\delta_8 = 1] \implies [z_7 = x_2]
\]
To transform the propositional logic involving the continuous auxiliary variables \(z_i\) into linear inequalities (see last row of Table I), lower and upper limits are imposed on the state variables \(x_i\), i.e. the input inductor current and the output capacitor voltage. Assume that the input inductor current is allowed to vary between 0 and \(I_{max}\), while the output capacitor voltage varies in the range \([V_{min}, V_{max}]\). Under these assumptions, the logic constraints in relationships (38) corresponding to \(z_1\) and \(z_2\) are equivalent to:
\[
\begin{aligned}
z_1 &\leq I_{max}\delta_1 \\
z_1 &\geq 0\delta_1 \\
z_1 &\leq x_1 - 0(1 - \delta_1) \\
z_1 &\geq x_1 - I_{max}(1 - \delta_1)
\end{aligned} \tag{39}
\]
\[
\begin{aligned}
z_5 &\leq V_{max}\delta_5 \\
z_5 &\geq V_{min}\delta_5 \\
z_5 &\leq x_2 - V_{min}(1 - \delta_5) \\
z_5 &\geq x_2 - V_{max}(1 - \delta_5)
\end{aligned} \tag{40}
\]
Similar sets of constraints are obtained in the same way also for the other variables.

To represent the hybrid model in form (10) (third step of the procedure), variables - binary and continuous - are grouped in matrices. Defining
\[
x(k) = [x_1(k) \ x_2(k)]^T, \\
\delta(k) = \begin{bmatrix} \delta_1(k) \\ \delta_2(k) \\ \delta_3(k) \\ \delta_4(k) \\ \delta_5(k) \\ \delta_6(k) \\ \delta_7(k) \end{bmatrix}, \ z(k) = \begin{bmatrix} z_1(k) \\ z_2(k) \\ z_3(k) \\ z_4(k) \\ z_5(k) \\ z_6(k) \\ z_7(k) \end{bmatrix}
\]
the system matrices become:
\[
A = \begin{bmatrix} 0 & 0 \\ \frac{T}{\tau} & 1 - \frac{T}{\tau RC} \end{bmatrix}, \ B_1 = 0, \\
B_2 = \begin{bmatrix} \frac{TV_a}{T} & 0 & 0 & 0 \\ 0 & \frac{TV_b}{T} & 0 & 0 \end{bmatrix}, \\
B_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \ D_1 = 0, \ D_2 = 0, \ D_3 = 0.
\]

For matrices appearing in the inequality constraint (11), the following partition has been chosen, according to the origin of the conditions:
\[
E_2 = \begin{bmatrix} E_{2\delta} \\ E_{2z} \end{bmatrix}, \ E_3 = \begin{bmatrix} E_{3\delta} \\ E_{3z} \end{bmatrix}, \\
E_4 = \begin{bmatrix} E_{4\delta} \\ E_{4z} \end{bmatrix}, \ E_5 = \begin{bmatrix} E_{5\delta} \\ E_{5z} \end{bmatrix}
\]
where
\[
E_{2\delta} \in \mathbb{R}^{12 \times 8}, \ E_{3\delta} \in \mathbb{R}^{12 \times 7}, \\
E_{4\delta} \in \mathbb{R}^{12 \times 2}, \ E_{5\delta} \in \mathbb{R}^{12 \times 1}.
\]
while

\[ E_{2z} \in R^{28 \times 8}, \quad E_{3z} \in R^{28 \times 7}, \]
\[ E_{4z} \in R^{28 \times 2}, \quad E_{5z} \in R^{28 \times 1}. \]

\[ E_{2z} = \begin{bmatrix} E_{2x_1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & E_{2x_1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & E_{2x_1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & E_{2x_1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & E_{2x_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & E_{2x_1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & E_{2x_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & E_{2x_1} \end{bmatrix}, \]

\[ E_{2\delta} = \begin{bmatrix} 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \end{bmatrix}, \]

\[ E_{3\delta} = \text{zeros}(12, 7), \quad E_{4\delta} = \text{zeros}(12, 2), \]
\[ E_{3z} = \text{diag}(E_{3z}). \]

\[ E_{4z} = \begin{bmatrix} E_{4z} & O_{4 \times 1} \\ E_{4z} & O_{4 \times 1} \\ O_{4 \times 1} & E_{4z} \\ O_{4 \times 1} & E_{4z} \end{bmatrix}, \quad E_{5z} = \begin{bmatrix} E_{5z} \\ E_{5z} \\ E_{5z} \\ E_{5z} \end{bmatrix}. \]

\[ E_{5z} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}^T, \]
\[ E_{2x_1} = \begin{bmatrix} -I_{\text{max}} \\ 0 \\ 0 \\ I_{\text{max}} \end{bmatrix}, \quad E_{2x_2} = \begin{bmatrix} -V_{\text{max}} \\ V_{\text{min}} \\ -V_{\text{min}} \\ V_{\text{max}} \end{bmatrix}. \]

\[ E_{3z} = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix}, \quad E_{4z} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad E_{5z_1} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \]

\[ E_{5z_2} = \begin{bmatrix} 0 \\ -V_{\text{min}} \\ V_{\text{max}} \end{bmatrix}. \]

V. SIMULATION OF THE BOOST CONVERTER

At the implementation stage an important issue is that the resulting MLD model has to be completely well-posed, that is at each time instant the pair \((x(k), u(k))\) should generate single-valued \(z(k)\) and \(\delta(k)\), otherwise the simulation is not feasible. Usually this property of the hybrid system is guaranteed by the way the inequalities (11) are generated [8]. For a formal demonstration of this property, the values of \(z(k)\) and \(\delta(k)\) can be determined by solving a Mixed-Integer Feasibility Test with respect to the constraints (11). This test can be done efficiently by using branch and bound algorithms [13], [14].

The MLD boost converter model was implemented and simulated in Matlab. In order to verify the representation accuracy, the output of the MLD model is compared with the output of the real model (1).

The real model of the boost converter was implemented in Simulink. The parameters of the boost converter used in simulations are:

\[ L = 1 \text{mH}, \quad C = 100 \mu\text{F}, \]
\[ R = 200 \Omega, \quad V_g = 230V. \]

Figs. 3 and 4 present respectively the variations of the output capacitor voltage \(V_d\) and of the inductor current \(i_L\) when the position of the switch was modified every 10\(µ\text{sec}\) (this corresponds to a duty cycle of 0.5). The two figures are obtained via the MLD model. However, the real model of the boost converter provides approximatively the same result, illustration of both curves on the same graphic being meaningless for the chosen time span.

A detail of the output capacitor voltage is provided in Fig. 5. The output of the MLD model is represented by the continuous line, the output of the real model is depicted by the dashed line, the maximum difference between the two representations being of half a volt.
Fig. 4. Input inductor current

Fig. 5. Detail of the output capacitor voltage - MLD model (continuous), real model (dashed)

Fig. 6. Output capacitor voltage for various switching patterns - MLD model (continuous), real model (dashed)

Fig. 7. State transitions

VI. CONCLUSION

In this paper a MLD model was developed for a boost converter. The MLD representation accuracy was verified against the response obtained from the real model of the boost converter. The simulation results show a very good correspondence between the two models.

From the complexity point of view, the real model consists in two differential equations involving two state variables and one integer input variable for each of the three possible working situations, while the MLD model described in this paper has 8 integer variables, 7 continuous variables and 40 inequality constraints. In this light, building minimal MLD models is of great importance. However, in spite of the
high complexity that MLD development could lead sometimes to, it provides a unified framework for modelling and control of hybrid systems which explicitly takes into account all possible functioning modes of the system rather than controlling them separately and interpolating between them.

REFERENCES