HYBRID MODELING OF OPEN LOOP DC-DC CONVERTERS

Ekaitz Zulueta[1] Teodoro Rico[1] José María González de Durana [1]

Received september 02, 2003, accepted december 31, 2003

ABSTRACT

Power electronic converters always have been circuits of difficult modelling because differential equations that describe them have discontinuities. Although this situation has been improved since the appearance of the Hybrid Systems theory, able to jointly describe both continuous and discrete behaviors exhibited by some physical systems, nowadays it is possible to obtain very precise models which help us in the study and design of such circuits. An excellent option for the discrete part model (reactive system) is to use statecharts, since this powerful language has recently been implemented and named Stateflow as a part of the Simulink toolbox of Matlab. So, today, the complete modeling of some hybrid systems within Matlab environment is possible. In this work the open loop hybrid modeling and simulation of the well-known dc-dc converters named buck and boost, using Matlab-Simulink-Stateflow, is presented.

Keywords: Hybrid-systems, hybrifold, dc-dc converters, reactivate-systems, switching-systems, statecharts.

INTRODUCTION

Hybrid Systems constitute a multi-disciplinary area which arises during the last decade and extends between the limits of computer science, applied control engineering and mathematics. A hybrid system is a mathematical model able to represent some complex physical systems with hierarchic structure and made up of discrete and continuous subsystems which communicate and interact with each other. This system's operation can be explained, intuitively, by means of the next diagram

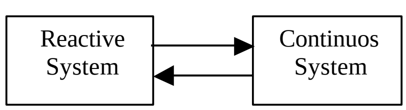

Fig. 1.- Buck converter

in which the *reactive system* (also called *discrete event system*) includes all the discrete components, whereas *continuous time system* includes the rest of them, all with continuous dynamics. The reactive system processes events that receive and emit orders; the continuous time system evolves continuously in time, in agreement with the physical laws and based on orders received, and emits events when such evolution fulfil determined conditions.

There are also other more formal descriptions, mathematical definitions, which are the base on which present theory of Hybrid Systems settle down. Hybrid Systems can serve to represent a large variety of physical systems: realtime systems, structured software, robotics, mechatronics, aeronautics, electronics systems, process automation, economic systems and even biological systems [5].

The commutation dc-dc converters are electronic systems which turn into a level of continuous voltage to another, higher or lower, level through a commutation mechanism. They are used a lot to build power sources of electronic equipment, due to their great efficiency and reduced size, and also for electrical motor control. A dc-dc converter admits clearly being modeled as hybrid system since it is formed basically by an electrical circuit (system of continuous time) and by other devices, such as diodes and transistors, that work as electronic switches (systems of discrete time) on such way that switches act over electrical circuit and vice versa. The more used converters are the down converter (*buck*), the up converter (*boost*), the up–down converter (*buck – boost*), the *Cúk* converter and the *full-bridge* converter. The main goal of this work is the obtainment of hybrid models for the *buck* and *boost* converters and then simulating the behavior for each one.

We will denote by $\boldsymbol{Z}_+$ the set of positive or equal whole numbers to zero, by $\boldsymbol{R}$ the set of real numbers, by $\boldsymbol{R}_+$ the set of positive or equal to zero real numbers and by $\boldsymbol{R}^n$ the vectorial space of n-tuples.

[1] Universidad del País Vasco, EUITI, Vitoria-Gasteiz, España, iepzugue@vc.ehu.es, iszripat@vc.ehu.es, jtpgogaj@vc.ehu.es.

HYBRID SYSTEMS

The next definition of Hybrid System has been taken from [5].

Definition: *An n-dimensional hybrid system with k states is a 6-tuple*

H=(Q,E,D,F,G,R)

whose elements are:

- *Q = {1, ..., k} the finite set of states,*
- *E ÌQ ´Q the collection of edges,*
- *D={D_i:i ÎQ} the collection of domains where D_i Ì{i}´$\boldsymbol{R}^n$ for all i ÎQ,*
- *F={F_i:i ÎQ} the collection of vector fields such that F_i is Lipschitz on D_i for each i ÎQ.*
- *G={G(e):e ÎE} collection of guards where for each e=(i,j) ÎE, G(e) ÌD_i ,*
- *R={R_e:e ÎE} the collection of resets where for each e=(i,j) ÎE, R_e is a relation between elements of G(e) and elements of D_j, i.e. R_e ÌG(e) ´D_j .*

We would like to remark the change in notation which comes up in Hybrid Systems with respect to the used in Dynamic Systems theory. Thus, in dynamic continuous time systems, the term "state of the system" usually denotes, for each $t\in\boldsymbol{R}$, a point

$$x(t)\in\boldsymbol{R}^n,$$

in the trajectory or solution of a differential equation. Nevertheless, in Hybrid Systems, a state is the situation in which the system stays while certain conditions are fulfilled.

ANALYSIS

The analysis that usually is made for the dc-dc converters is based on assuming a priori some operation hypothesis, in order to be able to obtain formulas that therefore will be valid only if such hypotheses are fulfilled. Nevertheless, we will not made previous hypothesis but we will associate the operation modes of the system to different states of an hybrid system.

In the circuits we will suppose switch *sw* is controlled by a binary periodic signal *clock*(t), with period T_s, being t_{on} and t_{off} the times during which the function value is 1 and 0 respectively.

We will consider three states, S_{on}, S_{off} and S_{nc}. The states S_{on} and S_{off} will be associate to operation modes with switch *sw* in states *on* and *off*, respectively, whereas the state S_{nc} will be associate to the mode in which the diode does not conduct (null intensity). In this way we are going to analyze the both *buck* and *boost* converters. We will denote by $i_l(t)$ the intensity through the coil and $v_c(t)$ the voltage across the capacitor.

Buck converter

This converter gives an output voltage v_o smaller than the input voltage v_s. It is based on the circuit of figure 2.

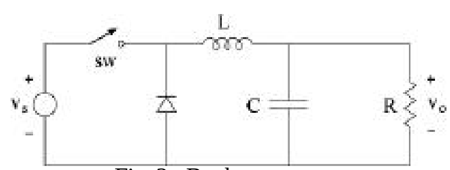

Fig. 2.- Buck converter.

If *sw* is closed, the diode is on inverse polarization and can be eliminated for analysis. The resulting electrical system, without diode and with *sw* closed, with two meshes, is described by the pair of differential equations

$$\begin{aligned} \frac{di_l(t)}{dt} &= -\frac{1}{L}v_c(t)+\frac{1}{L}v_s(t) \\ \frac{dv_c(t)}{dt} &= \frac{1}{C}i_l(t)-\frac{1}{RC}v_c(t) \end{aligned}$$

When opening the switch *sw*, whenever the intensity i_l is positive, it will also flow through the diode, now directly polarized. For analysis we can replace the diode by a conductor and delete the switch *sw* and the source of voltage v_s. The resulting circuit, with two meshes, is described by the equations,

$$\begin{aligned} \frac{di_l(t)}{dt} &= -\frac{1}{L}v_c(t) \\ \frac{dv_c(t)}{dt} &= \frac{1}{C}i_l(t)-\frac{1}{RC}v_c(t), \end{aligned}$$

which are the same ones that have been described for the previous mode by doing v_s=0.

In switch-off mode, with i_l>0, the voltage v_c in the capacitor will be increasing and the intensity i_l will be diminishing; if time t_{off} is long enough, it will be a moment when i_l is annulled, later trying the capacitor to discharge through the diode, which is not possible, so there is to be i_l=0. In this case, with i_l=0, we can consider the circuit reduced to a single mesh, the one that contains R and C. This circuit is described by

$$\frac{di_l(t)}{dt} = 0$$
$$\frac{dv_c(t)}{dt} = -\frac{1}{RC} v_c(t).$$

Boost converter

This converter is able to give an output voltage v_o greater than the input one v_s. It is based on the circuit of Fig. 3.

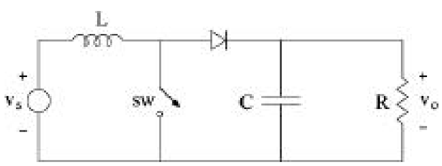

Fig. 3.- Boost Converter

When switch *sw* is closed, the diode is on inverse polarization (it can be eliminated for analysis), the mesh on the left is isolated and the equations are:

$$\frac{di_l(t)}{dt} = \frac{1}{L} v_s(t)$$
$$\frac{dv_c(t)}{dt} = -\frac{1}{RC} v_c(t).$$

If switch *sw* is open, whenever i_l is positive, the diode is directly polarized. For analysis we can replace the diode by a conductor and eliminate switch *sw*. The resulting circuit with two meshes is described by

$$\frac{di_l(t)}{dt} = -\frac{1}{L} v_c(t) + \frac{1}{L} v_s(t)$$
$$\frac{dv_c(t)}{dt} = \frac{1}{C} i_l(t) - \frac{1}{RC} v_c(t)$$

In switch-off mode, with i_l>0, the voltage v_c in the capacitor will be increasing and the intensity i_l will be diminishing; if the time t_{off} is long enough, it will be a moment when i_l is annulled, later trying the capacitor to discharge through the diode, which is not possible, so there is to be i_l=0. In this case, with i_l=0, we can consider the circuit reduced to a single mesh, the one that contains R and C. This circuit is described by

$$\frac{di_l(t)}{dt} = 0$$
$$\frac{dv_c(t)}{dt} = -\frac{1}{RC} v_c(t).$$

State models

With usual notation of linear systems state model, by means of the changes i_l=x_1,v_c=x_2,v_s=u, each mode of previous converters is described by the state model

$$x'(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) \qquad (1)$$

with matrices A and B given in the table 1

Table 1.- Matrices

	A	B	A	B
S_{on}	A_1	B_1	A_2	B_1
S_{off}	A_1	B_2	A_1	B_1
S_{nc}	A_2	B_2	A_2	B_2
	buck		Boost	

where

$$A_1 = \begin{bmatrix} 0 & \frac{-1}{L} \\ \frac{1}{C} & \frac{-1}{RC} \end{bmatrix}, B_1 = \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix},$$
$$A_2 = \begin{bmatrix} 0 & 0 \\ 0 & \frac{-1}{RC} \end{bmatrix}, B_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \qquad (2)$$

The matrix C will be the adapted to the variables considered as outputs in each application.

HYBRID MODELS

Buck hybrid model

We are liked to specify the elements of the hybrid system

$$H = (Q, E, D, F, G, R),$$

before defined, for the case of the *buck* converter. As we already said, we considered three states: state 1 =s_{on} state 2 =S_{off} and state 3 =S_{nc}, reason why k=3, and

$$Q = \{1, 2, 3\}$$

are the numbers that indicate the states. The collection of edges is going to be

$$E = \{(1,2), (2,1), (2,3), (3,1)\}.$$

Originally we have two state variables, $x_1 = i_l$ and $x_2 = v_c$. However, in order to use the methodology of Simić et al. [5], we are going to add to the system the differential equation

$$x'_3(t) = 1,$$

whose solution, with $x_3(0) = 0$, is $x_3(t) = t$, $t \geq 0$, so $x_3(t)$ will exactly represent the time. So we have n=3.

Both the intensity i_L and the voltage in the capacitor are positive or zero, therefore the collection of domains is

$$D = \{D_i := \{i\} \times \boldsymbol{R}_+^{\;3},\; i \in Q\},$$

so the system will evolve in three copies of $\boldsymbol{R}_+^{\;3}$.

The vectorial fields are easily obtained from the before shown table. Thus, we have

$$F(t) = \begin{bmatrix} F_1(t) \\ F_2(t) \\ F_3(t) \end{bmatrix} \tag{3}$$

being

$$F_i(t) = \begin{bmatrix} A_{ij} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} B_{ij} \\ 0 \end{bmatrix} u(t), \tag{4}$$

where $i \in Q$ indicates the row of the table 1 and j=1 indicates the column 1 of the same that corresponds to the *buck* converter.

The guards are

$$G(1,2) = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in \boldsymbol{R}^3 : x_3 = t_{on} + T_s \boldsymbol{Z}_+ \right\}$$

$$G(2,1) = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in \boldsymbol{R}^3 : x_3 = T_s \boldsymbol{Z}_+ \right\}$$

$$G(2,3) = \left\{ [x_1; x_2; x_3] \in \mathbf{R}^3 : x_1 = 0 \right\}$$

$$G(3,1) = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in \boldsymbol{R}^3 : x_3 = T_s \boldsymbol{Z}_+ \right\}.$$

Finally, the collection of resets is

$$R = \{R_e(x) = i_d(x) : x \in G(e),\; e \in \mathrm{E}\},$$

where $i_d(x)$ is the identity function.

Boost hybrid model

It is practically equal to the previous one. The only difference is the vectorial fields that, in this case, are obtained from the table like before but now using j=2.

SIMULATION

A suitable election, we thought, to implement the methodology of Simić et al. [5] is by using the powerful language *Statecharts* to capture the reactive system. For the continuous dynamic part (differential equations) there are many available options. But it is also necessary to have some programming environment that allows us to combine both procedures at run time. For this, some very powerful systems, that allow the complete modeling of complex hybrid systems , recently have been developed.

For practical reasons we used Stateflow to model the reactive system, and Simulink to model the continuous time system. The scheme we have built in Simulink-Stateflow to simulate the hybrid model

$$H = (Q, E, D, F, G, R)$$

appears in Fig. 4. We can see there is a Stateflow block named *Chart* and the others are Simulink blocks.

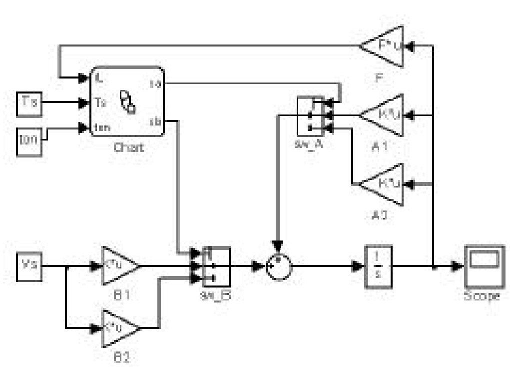

Fig. 4.- Diagram of H in Simulink

The *Chart* block implements a statechart with 3 states, S_{on},S_{off} and S_{nc} (see Figs. 5 and 8), representing the states Q={1,2,3} of H.

Edges E={(1,2),(2,1),(2,3),(3,1)} of H are represented by statechart transitions (arrows).

Domains $D=\{D_i:=\{i\}\times \boldsymbol{R}_+^3, i\in Q\}$ of H, where the flow of the hybrid system evolves (*hibrifold*) [5], cannot have a complete representation because they are infinite. In their place, the algorithm will calculate a finite succession of values of t (time) and their corresponding values of $x\in \boldsymbol{R}^3$, being superposed the three copies of $\boldsymbol{R}^3$.

Vectorial fields are the indicated in (4). In order to select the appropriate matrices by Simulink, we used a commutation mechanism made up by switches sw_A and sw_B which are driven by statechart outputs *sa* and *sb*.

Guards are represented by firing conditions of statechart transitions and they are denoted by writing between brackets these conditions, as it can be seen in figures 5 and 8.

As far as the collection of resets, in this case they have not any representation, because all of them are equal to the identity and the three copies of $\boldsymbol{R}^3$ are superposed.

Buck converter

The statechart we have made for this converter is in figure 5. The state machine implements the *clock*(*t*) signal, essential for the system operation, based on event *t* received from the system and which is given by the simulation algorithm step: whenever the algorithm increases the value of *t* in a step, an event *t* is activated.

When simulation begins, the system goes to state 1=S_{on}, because it is the state pointed by the beginning transition (•⟶). When entering this state, the system stores the value of t in the variable t_1, puts the outputs to Simulink s_a=1, s_b=1 and remains in that state until the condition [t>=t_1+t_{on}] of transition to state 2 is activated. This means that during a period of time of length t_{on}, the algorithm makes the simulation of the system given by the differential equation (1), where the matrices A and B are assigned to the values A_1 and B_1, that must be previously in the memory of Matlab (*workspace*).

Once the time t_{on}, passes the transition will be activated and the system will go to state S_{off}. When the system enters this state, immediately it does s_a=1 and s_b=2, then the values of matrices change and the algorithm will continue simulating the system (1) with A=A_1 and B=B_2. During this state, due to the presence of the diode, the current i_l must be always positive.

From this state, the system can jump to the other two states: to the state S_{on} and to the state S_{nc} (non-conduction state of the diode). In order to jump from the state S_{off} to the state S_{nc}, it is necessary for the time passed in this state to be greater than the time t_{off} . For the jump from the state S_{off} to the state S_{nc} the current i_l must be negative or zero and, in addition, the time passed in this state must not surpass the time t_{off}.

In the state S_{nc} the variables s_a and s_b are both put equal to 2, so the matrices A_2 and B_2 are selected. In this state the current i_l is zero, and the load current is provided by the accumulated charge in the capacitor C. From this state the system can only go to the conduction state. The transition occurs when a complete period of the *clock*(*t*) signal has passed.

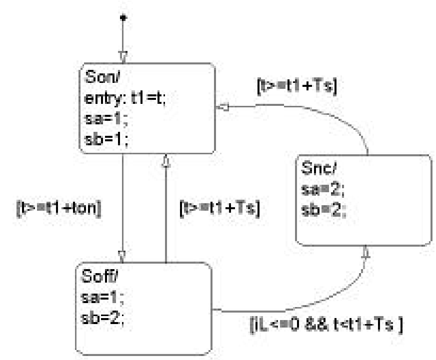

Fig. 5.- Statechart of buck

Execution and graphical results

Once we have constructed the model by means of Simulink-Stateflow, we can make the simulation of the system. For that, we select the algorithm of Dortman-Prince with fixed step, although other algorithms also have given good results. Then, in the Matlab command window, we introduce the data

```
>>  Ts=1e-3; ton=Ts/2; Vs=20;
>>  R=5; L=5e-3; C=47e-6;
>>  A1=[0 -1/L; 1/C -1/(R*C)];
```

```
>>  A2=[0 0; 0 -1/(R*C)];
>>  B1=[1/L; 0]; B2=[0; 0];
```

and we start the simulation. Thus the graphs of variables $i_l(t)$ $v_c(t)$ appears in the window of the block *scope*.

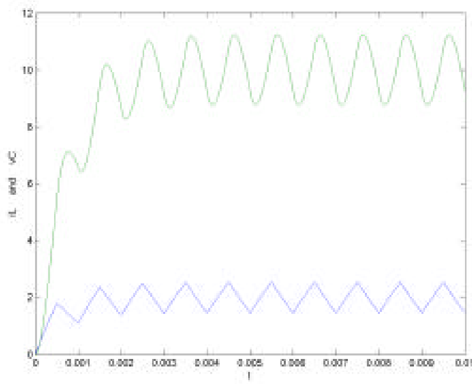

Fig. 6.- $i_l(t)$ and $v_c(t)$ of *buck* in CCM

Fig. 6 shows the converter has a stationary state of type CCM, that is, continuous conduction mode.

A new execution, changing only the value of the resistance *R*,

```
>>  R=25;
```

gives as result the graphs in Fig. 7, indicating an operation of type DCM, that is, discontinuous conduction mode.

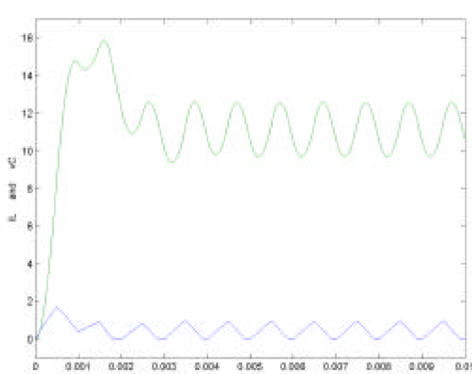

Fig. 7.- $i_l(t)$ and $v_c(t)$ of *buck* in DCM

Boost converter

The statechart of this converter is similar to the buck's one. The only differences are the values assigned in each state to the outputs s_a and s_b to Simulink, which are used to change the values of the matrices *A* and *B* in Fig. 8. Although, with the new values of the matrices, the operation of the system is similar to the buck's one.

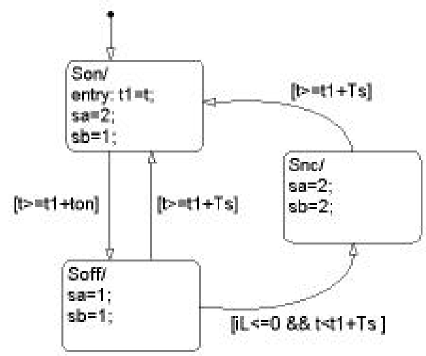

Fig. 8.- Statechart of *boost*

Execution and graphical results

With the same parameters as we used before for the algorithm, and now putting the data

```
>> Ts=5e-4;ton=1.2e-4;V_s=20;
>> R=4; L=250e-6; C=100e-6;
>> A1=[0 -1/L; 1/C -1/(R*C)];
>> A2=[0 0; 0 -1/(R*C)];
>> B1=[1/L; 0]; B2=[0; 0];
```

we obtain the graphs of the Fig. 9, in which a CCM operation type is shown.

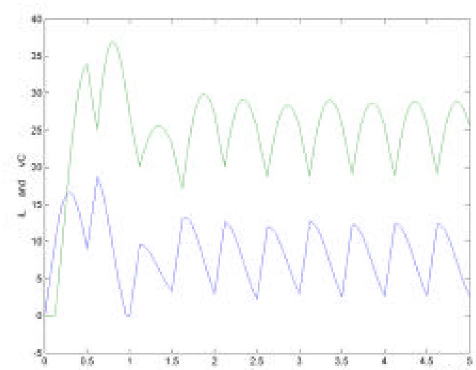

Fig. 9.- $i_l(t)$ and $v_c(t)$ of *boost* in CCM

Finally, changing the value of resistance *R*,

```
>> R=10;
```

we obtain the graphs of the Fig. 10 in which a clear DCM, discontinuous operation mode, is shown.

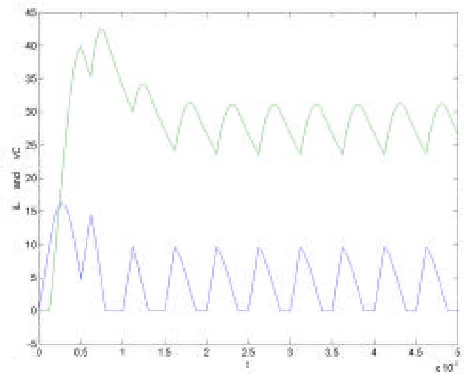

Fig 9.- $i_l(t)$ and $v_c(t)$ of *boost* in DCM

FINAL REMARKS

Actually, the switch which appears in the circuits of both converters is implemented by a transistor and that is the reason for the current, flowing through it, can circulate in an only direction. This fact could be considered easily only adding a new guard $e_5 = (1,3)$ to the model of the converter *boost*. This guard would be represented in the statechart by an associated transition with the condition $[i_l\text{<=}0 \;\&\&\; t<t_1+T_s]$.

It is necessary to have well-taken care when fixing the transition conditions so that they exclude themselves to each other, not giving rise to blocking states during simulation. For that reason, condition $[t<t_1+t_s]$ has been added to $[i_l\text{<=}0]$ one, that has been put at first for the guard $G(1,2)$.

We have observed in simulations that the *buck* and *boost* converters are stable in open loop and that for them do not take place the Zeno phenomenon. We think it could be particularly interesting the study of stability of these circuits in similar way as the authors of [3] made.

ACKNOWLEDGEMENTS

We would like to express our thankfulness to Ingeborg Mahla, by her kindles providing us the manuscript of their work [3] and by her interesting comments and ideas about Hybrid Systems.

Also to T. John Koo by his gentleness providing us the manuscript of their work [1]. As far as we know, they have been the first in obtaining hybrid models of power electronics devices.

Finally, to professor D. Manuel Fernandez Rodriguez, from the University of Oviedo, area I.S.A., for having notified to us, some several years ago, about the importance and utility of Matlab's Stateflow *toolbox.*

REFERENCES

[1] Matthew Senesky, Gabriel Eritrea, and T. John Koo: *Hybrid Modelling and Control of Power Electronics*. Hybrid Systems: Computation and Control, Lecture Notes in Computer Science, Vol 2623, p450-465, Springer-Verlag, 2003.

[2] J.H. Su, J.J. Chen and D.S.Wu: *Learning feedback controller design of switching converters via Matlab/Simulink*. IEEE Transactions on Education, Vol. 45, NO.4, pp. 307–315, Nov. 2002.

[3] Ricardo F. Ovalle, Ingeborg Mahla: *Introducción al modelamiento y análisis de estabilidad de sistemas híbridos mediante la transformación de Poincaré*, Anales del XV Congreso Chileno de Control Automático, Santiago, 28-31 de octubre de 2002.

[4] D. Harel, *Statecharts: a visual formalism for complex systems,* Science of Compueter Programing 8, pp. 231-274, 1987.

[5] S.N. Simić, K.H. Johanson, J. Lygeros & S. Sastry: *Towards a Geometric Theory of Hybrid Systems* Hybrid Systems: Computation and Control, vol. 1790 of Lecture Notes in Computer Science, pp.421–436, Springer Verlag, Pittsburg, PA, 2000.