

Control of Binary Input Systems

William HOLDERBAUM

School of Systems Engineering, the University of Reading, Reading, RG66AY, UK

Abstract:—Nowadays many systems are controlled by items of commutation, mainly in powers electronics and hydraulic. This Boolean input systems has increased in the electric industry. Power supply include such system and there are represented by power converter. The goal of this paper is to propose a methodology to design a control law to apply on systems with Boolean inputs. Firstly, we study the problem of regulation around a desired state trajectory. Afterwards, the algorithm is applied to a predefined trajectory from initial state. Also this method is used for the tracking of a real time trajectory. Finally, these results are implemented to an electronic circuit.

Keyword:—Boolean control, commutation, state space.

I. INTRODUCTION

Several methods dealing with Boolean control have been developed during the last years. These strategies of Boolean control are generally applied in order to improve the behavior of switching systems. These switching systems are frequently used in industrial applications because they give a high power in output as for example power converters. The power converter is a part of Boolean input systems. The behavior of such systems is controlled by the switching ON (value 1) and OFF (value 0) of components as thyristors or transistors. These Boolean control methods are separated in three classes. The first class of methods consists to control the process using mean values of inputs, like in the well known P.W.M (Pulse Width Modulation) technique (1)(2). The regulation is often realized by a P.I.D controller of which performance is good. But this controller doesn't straight take effect on the commutation. The second class of control design consists in keeping the binary values of the inputs, and in using different approaches like Sliding Mode Control(SMC) (3)(4),input-output decoupling (5). This technique is characterized by discontinuous control action on Variable Structure System(VSS) which changes structure upon reaching a set of switching surfaces. The switching instants are determined by appropriate sliding surface (switching surface). Sliding surface are chosen to achieve a desired dynamical response.

Sliding Mode Control for multi-input systems is used to control electronics converter. The main drawback of a Sliding Mode Control system have that it is very difficult to choose sliding surface in Multi Input Multi Ouput (MIMO).

The last class concern the direct self control (DSC) which is a recently introduced control methodology (6)(7). This method is generally applied to electrical system composed of an induction motor and its power converter (8). The basic principle of DSC is in fact to control torque and stator flux amplitude by selecting suitable switching states on the basis of the stator flux position. But this procedure is too exclusive, because it is used for motor control.

A new Boolean control system is introduced in this paper. This technique uses directly the Boolean values to control the system in order to take account the whole model, and to act on commutations. The paper is outlined as follows. The section II describes the model used. In section III we define a new control criterion for systems with binary inputs. Then we will apply this criterion for the regulation around a predefined trajectory, and afterward to follow a real time trajectory. We illustrate the proposed method by application to a particular example.

II. PRESENTATION

Consider the system modeled by the state equation :

$$\dot{x} = f(x, u) \quad (1)$$

where $x = (x_1 \dots x_n)^T \in \mathfrak{R}^n$ is the state vector and $u = (u_1 \dots u_m)^T \in \{0,1\}^m$ is the input vector composed of Boolean variables.

The input vector u can take any configuration (9) among 2^m different vectors $Config_i(u)$ containing Boolean values as :

$$\{Config_i(u) / i = 1 \dots 2^m\} = \left\{ \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}_1 \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}_2 \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}_3 \dots \begin{pmatrix} 0 \\ 0 \\ 1 \\ \vdots \\ 1 \end{pmatrix}_{m+1} \begin{pmatrix} 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}_{m+2} \begin{pmatrix} 1 \\ 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}_{m+3} \dots \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix}_i \dots \begin{pmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}_{2^m} \right\}$$

Between two commutations of the system, the input u is a constant vector denoted $Config_i(u)$. For example if the number input $m = 2$ then the $Config_i(u)$ set becomes :

$$\{Config_i(u) / i = 1 \dots 4\} = \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}_1 \begin{pmatrix} 1 \\ 0 \end{pmatrix}_2 \begin{pmatrix} 0 \\ 1 \end{pmatrix}_3 \begin{pmatrix} 1 \\ 1 \end{pmatrix}_4 \right\}$$

III. BOOLEAN CONTROL

The aim of the study is to determine the sequence of $Config_i(u)$ for vector u for the state vector x to reach a desired fixed state (10), denoted by $(x)_d$.

Let us consider a vector ε which represents a position error vector associated with :

$$\varepsilon = (x)_d - (x)_p \quad (2)$$

where $(x)_p$ is the current position of the model in the hyperplane associated with the state space and $(x)_d$ is the target. We calculate the vector \vec{V}_i associated with $(\dot{x})_i$ such as $\vec{V}_i \stackrel{\text{def}}{=} (\dot{x})_i$, for each configuration $Config_i(u)$ associated with the current position $(x)_p$

$$\begin{cases} \vec{V}_i \stackrel{\text{def}}{=} (\dot{x})_i = f((x)_p, Config_i(u)) \\ i = \{1 \dots 2^m\} \end{cases} \quad (3)$$

Our aim is to determine the configuration $Config_i(u)$ minimizing the angle α between ε and \vec{V}_i . We can illustrate this procedure with the example shown figure 1, where $x \in \mathbb{R}^2$ and $u \in \{0,1\}^2$.

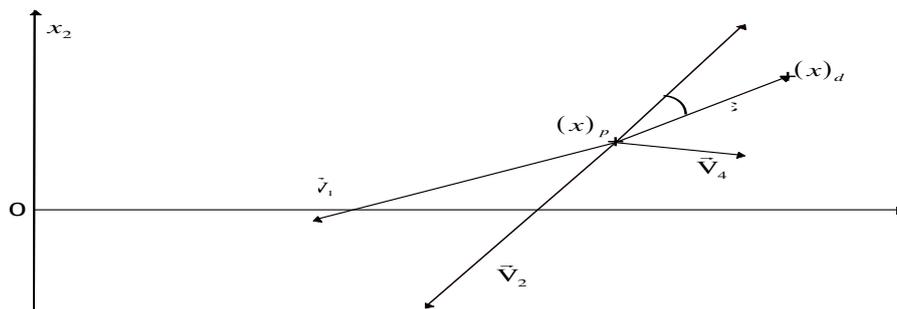


Fig.1 : Control in the state space

In that case, the vector \vec{V}_3 has the smallest angle with the vector ε . \vec{V}_3 corresponds to the control

$$Config_3(u) = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ applied to the model.}$$

Interpretation : If the target $(x)_d$ is fixed and constant, then $(\dot{x})_i$ can be defined by the equation (eq. 4) in relation to the derivative of ε (eq. 2) when the configuration u_i is applied.

$$(\dot{x})_i = -\left(\frac{d\varepsilon}{dt}\right)_i \quad (4)$$

We calculate the inner product between ε and \vec{V}_i ($\vec{V}_i \stackrel{\text{def}}{=} (\dot{x})_i$) by using :

$$\langle \varepsilon, \vec{V}_i \rangle = \|\varepsilon\| \cdot \|\vec{V}_i\| \cdot \cos(\alpha) \quad (5)$$

$$\text{then } \langle \varepsilon, \vec{V}_i \rangle = -\varepsilon^T \cdot \left(\frac{d\varepsilon}{dt}\right)_i \quad (6)$$

$$\text{or } \frac{d}{dt} \left\{ \frac{1}{2} (\varepsilon^T \cdot \varepsilon) \right\} = \frac{d(\varepsilon^T)}{dt} \cdot \varepsilon + \varepsilon^T \cdot \frac{d(\varepsilon)}{dt} \quad (7)$$

$$\text{consequently } -\varepsilon^T \cdot \left(\frac{d\varepsilon}{dt}\right)_i = -\frac{1}{2} \cdot \left(\frac{d(\|\varepsilon\|^2)}{dt}\right)_i \quad (8)$$

The cosine of the angle is given by :
$$\cos(\alpha) = \frac{\langle \varepsilon, \vec{V}_i \rangle}{\|\varepsilon\| \cdot \|\vec{V}_i\|} \quad (9)$$

The configuration $Config_i(u)$ is chosen by minimizing angle α . The minimization of angle α is deduced by the cosine maximum :

$$\begin{aligned} & \max(\cos(\alpha)) \\ & i = \{1, \dots, 2^m\} \end{aligned} \quad (10)$$

which can be written as :

$$\max_{i=1, \dots, 2^m} \left\{ \frac{-\frac{1}{2} \cdot \left(\frac{d(\|\varepsilon\|^2)}{dt}\right)_i}{\|\varepsilon\| \cdot \left\| \left(\frac{d\varepsilon}{dt}\right)_i \right\|} \right\} \quad (11)$$

So the control criterion is expressed by equation 11. By now we can show that the produce error due to this developed strategy control is bounded.

Proof: The operation principle is such that the deviation with respect to the next point, is less than smallest of two following terms : the initial deviation and the maximum allowable variations.

The proof is explained with the figure 2.

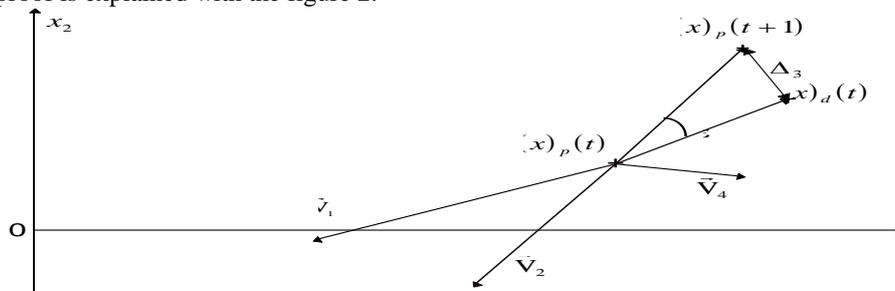


Fig.2 : Control in the state space

In fact the deviation with respect to the next point is stated by Δ_3 , consequently this deviation is less than smallest of these two terms : the initial deviation ε and the maximum allowable variations. This maximum

allowable variations is calculated for each configuration i (where $i = \{1, \dots, 2^m\}$) and represented by a set of $\|\vec{V}_i\|$ which obviously depend of the sampling time. That's why we can say that the error due to the strategy control is bounded.

IV. TRACKING PREDEFINED TRAJECTORY

Our main objective here is to allow the system to track a desired trajectory. The desired trajectory is predefined (10) in the state space (figure 3), and it is defined between a starting point M_1 to an end point M_E .

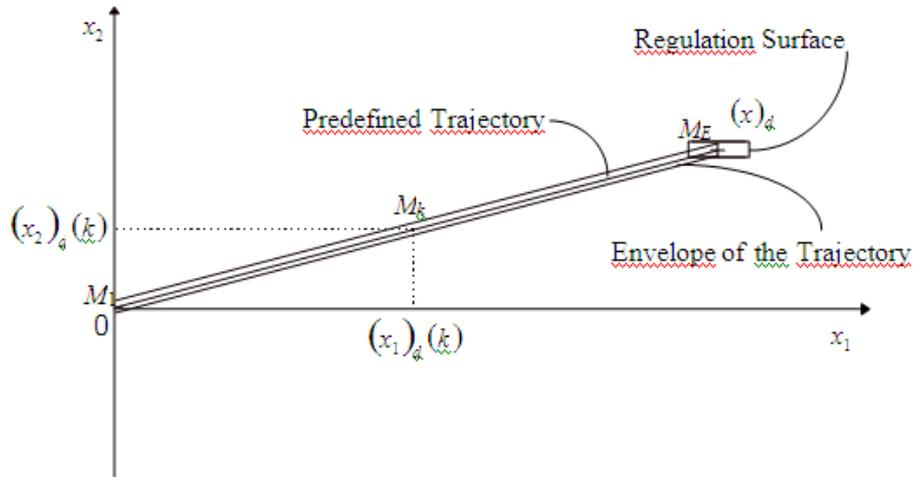


Fig.3 : Predefined trajectory in the state space

This trajectory is not supposed to be defined in real time, and consequently this definition form will be settled in order to respect specification sheets. In general case the equation form trajectory is unknown that's why it is necessary to discrete. So the defined continuous trajectory is changed into file data points. Hence we use the parametric equation to obtain the expected state trajectory, and which can be represented by a set of point $M_k = \left((x_1)_d(k), (x_2)_d(k) \right)$. This description is interpreted as follows :

$$\begin{cases} (x_1)_d = (x_1)_d(k) \\ \vdots \\ (x_n)_d = (x_n)_d(k) \end{cases} \quad (12)$$

where $(x_j)_d(k)$ ($j = 1, \dots, n$) represents the k^{th} discrete point define on the state variable trajectory by the parameter $k \in [1, E] \subset N$. The sampling period is chosen according to the dynamic system. The whole points number E depends from sampling period.

In order to minimize the number of switching, we define a domain of dimensional n all around this parameter curve whose boundary will be denoted the enveloping curve of the desired trajectory. This region contains the evolution trajectory of the system. We calculate the control u in closed-loop at each instant of intersection of the state trajectory with the enveloping curve of the desired trajectory (figure 3).

Methodology

The aim of this procedure is to minimize the commutation, and to control system in a bandwidth. For that we have to determine when the current position $(x)_p$ is inside or outside of the boundary domain previously defined by enveloping curve. The location of the current position $(x)_p$ is necessary to know when a new Boolean control has to be computed. A new configuration $Config_i(u)$ is calculated when the current position $(x)_p$ is outside so that the system converges inside the bandwidth. More precisely the idea is to select the velocity vectors which cross the envelope just when the current position $(x)_p$ goes outside it. So we choose between them the smallest angle with the vector ε .

Conversely if the current position $(x)_p$ is inside any control is computed, but the same control previously defined (when the current position $(x)_p$ goes outside the boundary domain) has to be applied. This last consideration has for objective to minimize the commutation frequency since the same control is kept in the bandwidth.

The method is summarized here, and presented in several steps in the next section. As to explain to the beginning of methodology we have to know at all moment the position $(x)_p$, in comparison of the discrete desired trajectory. To realize that, the minimal distance is computed with the discrete state trajectory with respect to whole points in the first step, but the major drawback is the compute time is too long. To overcome this problem, an observation window is defined around the predefined state trajectory, and by now the minimal distance have to be computed by this observation window. If the minimal distance corresponds to window 's boundary then the window is moved until the minimal distance does not belong to the limit of the window. So the compute time can be reduced. With the computed minimal distance, a vector orthogonal can be formed, this allows to discern which vectors cross the domain previously defined by using the inner product. Hence we can choose the configuration between them and the vector \mathcal{E} by the method previously exposed in section III.

1 In the first step : Initialization :

⇒ To initialize we search for the M_k on the discrete state trajectory with a minimal distance from $(x)_p$. The distance in \mathfrak{R}^n is calculated by using :

$$d_{min} = \min_{k=1 \text{ to } E} \sqrt{\left((x_1)_d(k) - (x_1)_p \right)^2 + \dots + \left((x_n)_d(k) - (x_n)_p \right)^2} \quad (13)$$

$k_{d_{min}}$: it is the parameter on this curve that corresponds to the minimal distance.

The consideration where $k_{d_{min}}$ is not unique has to be taken account here. That means there are several minimal distance d_{min} due to geometrical form of the desired trajectory. In that case, the choice is performed by taking the nearest value parameter $k_{d_{min}}$ with the whole number parameter E . This selection is realized in order to reach an end point M_E by the predefined trajectory as rapidly as possible, that's why this procedure of choice has been defined.

⇒ An observation window is defined (figure 4), called *win* around this discrete state trajectory. The size of *win* is defined by five points. The point $M_{k_{d_{min}}}$ which has been found corresponding to $k_{d_{min}}$ is used to form this window 's boundary as follows :

-center of window ⇒ $M_{k_{d_{min}}}$

-lower bound of the window

$$\Rightarrow \left\{ \begin{array}{l} M_{k_{min \text{ win}}} \\ \text{with } k_{min \text{ win}} = k_{d_{min}} - 2 \end{array} \right.$$

-upper bound of the window

$$\Rightarrow \left\{ \begin{array}{l} M_{k_{max \text{ win}}} \\ \text{with } k_{max \text{ win}} = k_{d_{min}} + 2 \end{array} \right.$$

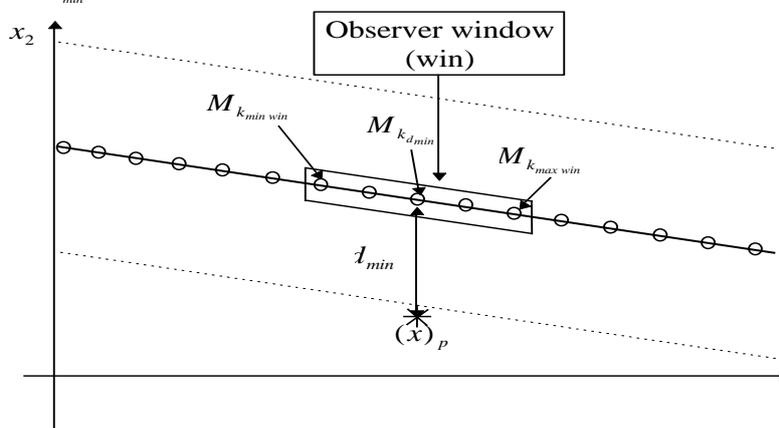


Fig.4 : Observation window around state trajectory in state space

For the following sampling time we go to the second step.

2 In the next iteration we repeat the calculation of the minimal distance between the current position and the observation window. This minimal distance calculated on the observation window allow to reduce the compute time, because the calculation is only performed on five points and not whole points on this discrete desired state trajectory.

$$d_{min} = \min_{k_{win}} \sqrt{\left((x_1)_d(k) - (x_1)_p \right)^2 + \dots + \left((x_n)_d(k) - (x_n)_p \right)^2} \quad (14)$$

with $k_{win} = k_{min\ win}$ to $k_{max\ win}$

$(x)_d$ represents the target such that $(x)_d(k)$ is the set of states which belongs to the window.

$$\{(x)_d(k) | (x)_d \in win\}$$

⇒ If the new computed point $M_{k_{d_{min}}}$ (resp. d_{min} with this minimal distance) corresponds to the window's boundary ($k_{d_{min}} = k_{min\ win}$ or $k_{max\ win}$); then the window on this curve is moved with respect to previous value of $k_{min\ win}$ and $k_{max\ win}$:

$$\begin{cases} k_{min\ win} = k_{min\ win} + 1 \\ k_{max\ win} = k_{max\ win} + 1 \end{cases}$$

This step by using distance minimal (equ. 14) and shifting window is repeated until the position $k_{d_{min}}$ (resp. d_{min}) on parametrized curve does not belong anymore to the window limits.

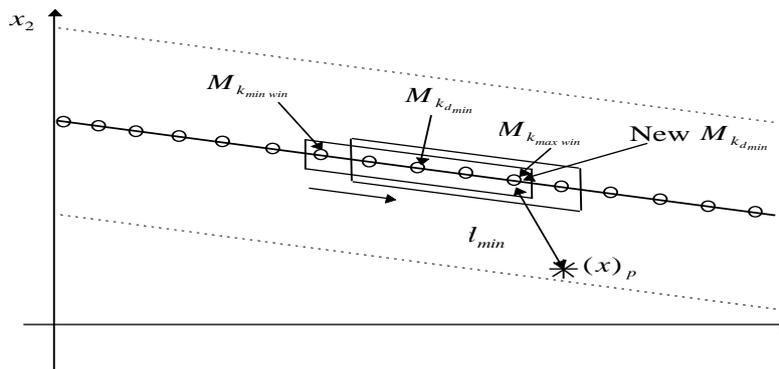


Fig.5 : Observation window moving around desired trajectory in state space

In this example (figure 5) d_{min} corresponds to the point $M_{k_{max\ win}}$ belonging to window's boundary, then the window is moved as previously defined.

⇒ If the new computed point $M_{k_{d_{min}}}$ (resp. d_{min} with this minimal distance) does not correspond to the window's boundary; then the new boundary limits ($k_{min\ win}$ and $k_{max\ win}$) of the observation window are defined and the algorithm can be go on to the third step.

The new definition of the observation window which is occurred in this stage, is expressed by taking $M_{k_{d_{min}}}$ as the center of the window as follows :

-center of window ⇒ $M_{k_{d_{min}}}$

-lower bound of the window

$$\Rightarrow \begin{cases} M_{k_{min\ win}} \\ \text{with } k_{min\ win} = k_{d_{min}} - 2 \end{cases}$$

-upper bound of the window

$$\Rightarrow \begin{cases} M_{k_{max\ win}} \\ \text{with } k_{max\ win} = k_{d_{min}} + 2 \end{cases}$$

3 The width between desired state trajectory and the envelope is defined by the distance d_c . In other words d_c represents the envelop around the desired state trajectory. The value of d_c is chosen by the user such that efficiently about the reduction of switching frequency and accuracy in order to follow desired state trajectory nearest possible have to be respected.

⇒ The calculated distance d_{min} is compared with this distance d_c .

-If $d_{min} < d_c$ thus we are inside the bandwidth and hence we apply the same control as for the previous sampling time. Then we return to the second step at the next sampling time.

-If $d_{min} > d_c$ then we go to the forth step.

4 The vector \vec{V}_{orth} is formed, defined below, using the minimal distance

$$\vec{V}_{orth} = (x)_p - M_{k_{d_{min}}} \quad (15)$$

⇒ By using the inner product between \vec{V}_{orth} and \vec{V}_i , the vectors which cross the envelope are determined.

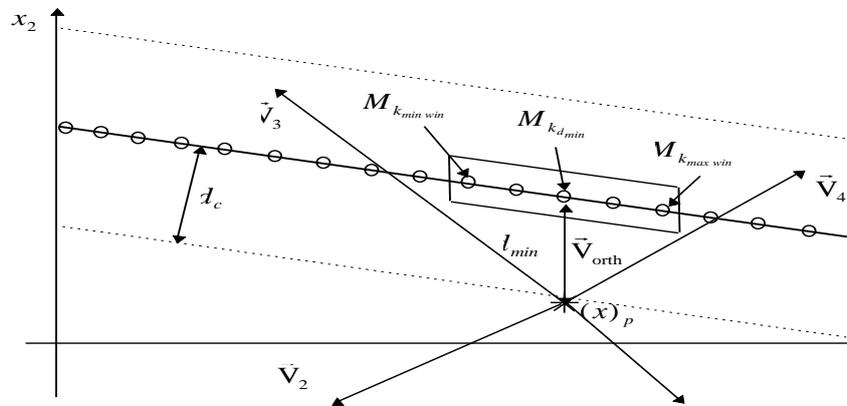


Fig.6 : Selection vectors which cross the envelope in state space

For this example (figure 6), the vector \vec{V}_3 and \vec{V}_4 which cross the tube will be selected to go in the fifth step.

5 The configuration u_i of input vector u is chosen among the remaining vectors which cross the envelope and we select the best vector by the method described in § 3 (**Boolean Control**).

The algorithm is restarted after the new sampling time in the second step. Repeating the above process we converge to the regulation domain.

This procedure is illustrated by the following scheme (figure 7):

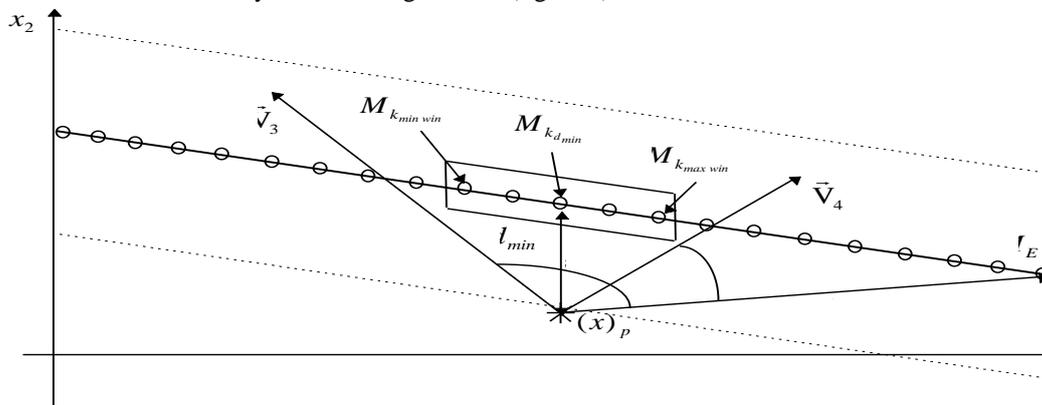


Fig.7 : Tracking in state space

Here vectors \vec{V}_3 and \vec{V}_4 cross the tube. The vector \vec{V}_4 has the smallest angle α_4 in comparison of α_3 with the vector ϵ . Therefore the vector \vec{V}_4 is chosen for the control in order to converge to the regulation region. The center of this regulation region is represented by the end point M_E as described in the next section.

Regulation problem

A domain D_r is defined around of the end point M_E (cf. figure 8). The goal of this domain is to minimize the commutation as previously described method.

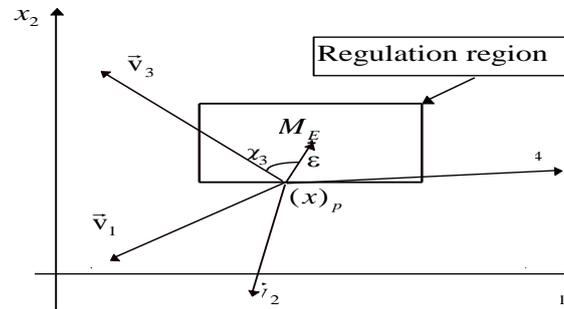


Fig.8 : Regulation in state space

If $(x)_p \notin D_r$, then the determination of the sequence u is made by using the method presented in § 3 (**Boolean Control**). For example in figure 8 the vector \vec{V}_3 will be applied for the control.

If $(x)_p \in D_r$, then the same previous computed control, corresponding to $(x)_p$ when it was outside of the domain D_r , is applied.

Application

This method is applied on a power electronic circuit illustrated in figure 9 modeled by a second order composed of inductor L1 and capacitor C1, resistances and switching components. This electronic circuit is supplied by two continuous voltages E1 and E2. The aim is to control the magnetic flux ϕ_{L1} in the inductor and the electrical charge q_{C1} in the capacitor in order to get a desired magnetic flux and electrical charge.

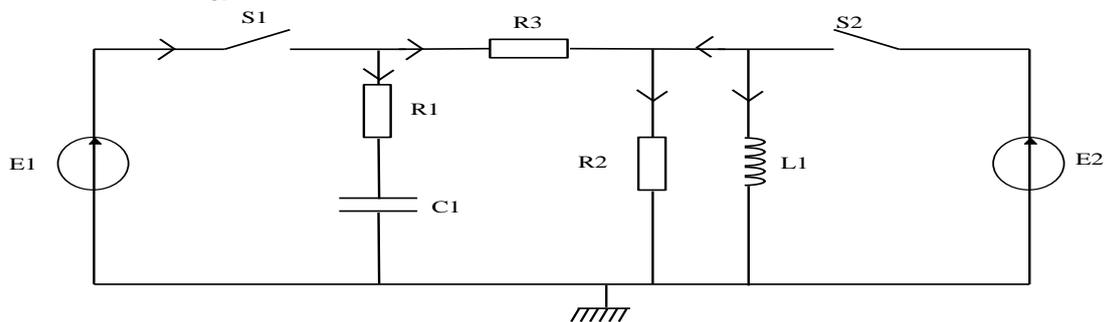


Fig.9 : scheme of electronics circuit

Modeling

The classical approach to study a switching electronic circuit is based on the determination of all possible topologies of the circuit with respect of each possible configurations. In fact there are 2^m topologies for the electronic circuit. This approach become very complicated when the number of switching elements increase. The proposed method, to obtain the model, is to use Bond-Graph methodology (11)(12). This approach allows to determine an unique state space equation of the electronic circuit under the form defined in (eq. 1). This unique model represent the association between switching elements and the circuit. The state equation so obtained includes Boolean terms. The control vector u is consequently straightforward since it contains the Boolean terms. The Bond-Graph is given figure 10 associated with the physical system :

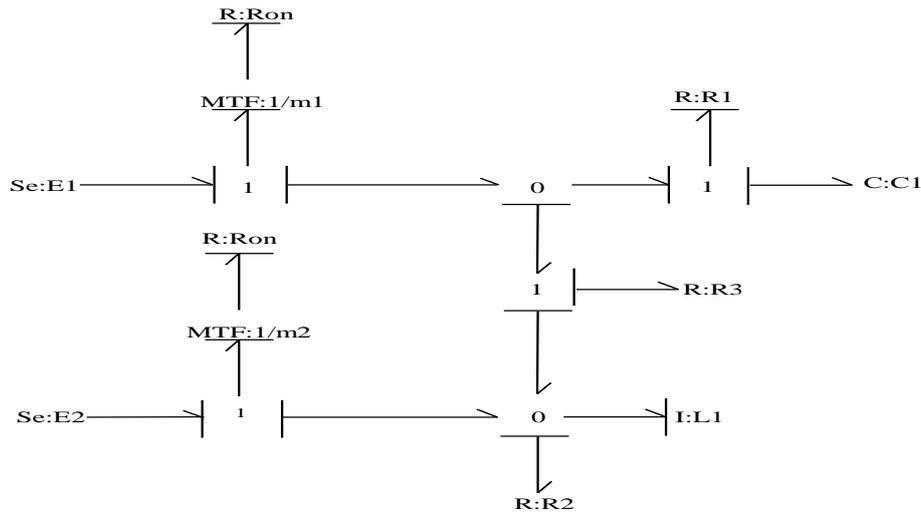


Fig.10 : Bond-Graph model

Ron is the internal resistance of the switches when they are ON, and m_i ($i = 1,2$) are Boolean variables representing the switching logical rules.

m_i for $i = 1,2$ are the Boolean control input which represents the state of two switches. $m_i = 0$ if the i^{th} transistor is in OFF state and $m_i = 1$ if it is in ON state.

The mathematical model of the circuit can be represented in this form :

$$\{\dot{x} = A(u)x + B(u)u \quad \text{with } x = \begin{pmatrix} \phi_{L1} \\ \phi_{C1} \end{pmatrix}$$

The input vector u is composed of the Boolean parameters as $u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \end{bmatrix}$ as the input vector . The

$$\text{matrices } A \text{ and } B \text{ are expressed as: } A = \frac{1}{\Delta_r} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}; B = \frac{1}{\Delta_r} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

with :

$$\Delta_r = 1 + \frac{R3}{Ron} u_1 + \frac{R3}{R1} + \frac{R2}{Ron} u_2 + \frac{R2}{Ron} u_1 + \frac{R2}{R1}$$

$$a_{11} = -\frac{R2}{L1} \left(1 + \frac{R3}{Ron} u_1 + \frac{R3}{R1} \right)$$

$$a_{12} = \frac{R2}{C1R1}$$

$$a_{21} = -\frac{R2}{L1R1}$$

$$a_{22} = -\frac{1}{C1R1} \left(1 + \frac{R3}{Ron} u_1 + \frac{R2}{Ron} u_2 + \frac{R2}{Ron} u_1 \right)$$

$$b_{11} = E1 \frac{R2}{Ron}$$

$$b_{12} = E2 \frac{R2}{Ron} \left(1 + \frac{R3}{Ron} u_1 + \frac{R3}{R1} \right)$$

$$b_{21} = E1 \frac{R3}{RonR1} \left(1 + \frac{R2}{Ron} u_2 \right) + E1 \frac{R2}{RonR1}$$

$$b_{22} = E_2 \frac{R2}{RonR1}$$

Consider the electrical parameters defined as :

$$\left\{ \begin{array}{l} E1 = 12 \text{ volts} \\ E2 = 4 \text{ volts} \\ Ron = 0.1 \Omega \\ R1 = 12 \Omega \\ R2 = 2 \Omega \\ R3 = 40 \Omega \\ L1 = 100\text{mH} \\ C1 = 4.7\text{mF} \end{array} \right.$$

Then the state equation becomes :

$$\dot{x} = \frac{1}{\Delta_r} \begin{bmatrix} -86.6 - 8000u_1 & 35.46 \\ -1.66 & -17.73(1 + 420u_1 + 20u_2) \end{bmatrix} \cdot x + \frac{1}{\Delta_r} \begin{bmatrix} 240 & 346.4 + 32000u_1 \\ 420 + 8000u_2 & 6.66 \end{bmatrix} \cdot u$$

with $\frac{1}{\Delta_r} = \frac{1}{4.5 + 420u_1 + 20u_2}$

The simulation software MATLAB-SIMULINK has been used for studying the response of the electrical system.

The sampling time period used for simulation is $t_{sim} = 0.001$ sec. This system in closed loop can be represented by :

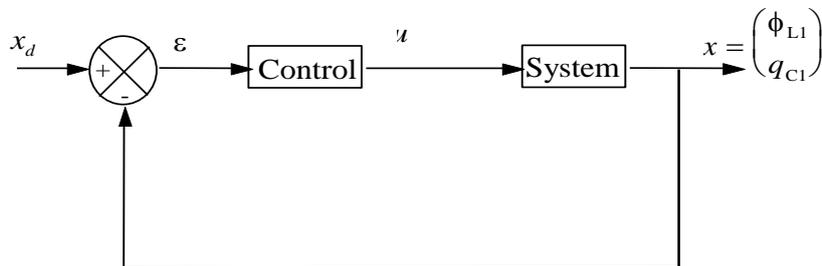


Fig.11 : Control scheme of electrical system

Simulation:

The target $(x)_d$ or M_E , and the predefined trajectory (as a linear trajectory) have been chosen inside the reachable domain, with :

$$(x)_d = \begin{pmatrix} \phi_{L1} = 3 \text{ Wb} \\ q_{C1} = 0.4 \text{ C} \end{pmatrix}_d$$

Figure 12 represents the trajectory in the state space.

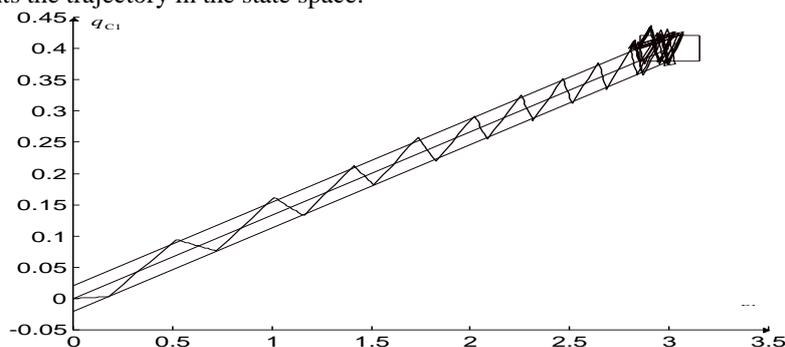


Fig .12 : Trajectory in the state space

The time responses of electrical charge q_{C1} in the capacitor and the magnetic linkage ϕ_{L1} in the inductor are represented figure 13 :

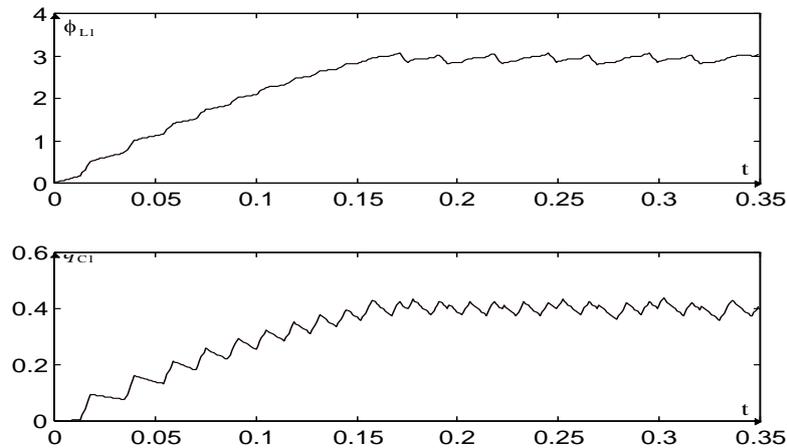


Fig.13 :Time variations of the magnetic linkage ϕ_{L1} and the electrical charge q_{C1} .

Figure 14 shows the variation of input variables :

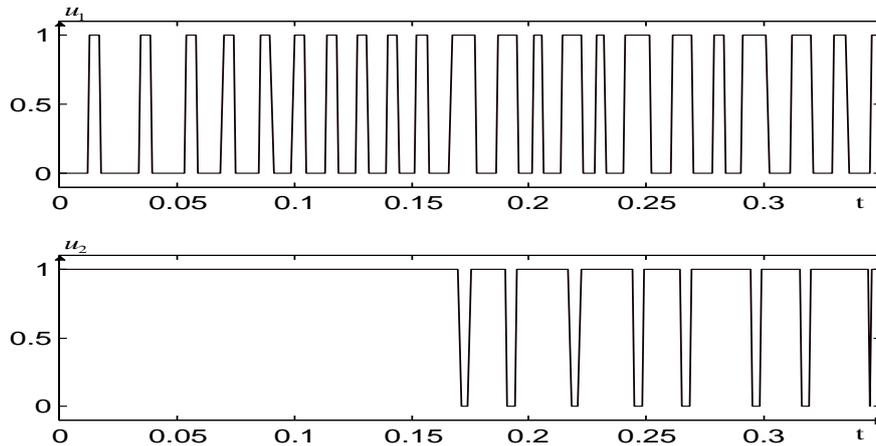


Fig.14 : Time variation of the input Boolean controls

The simulation results show that the predefined tracking trajectory, obtained with the Boolean control strategy.

V. A REAL TIME TRACKING TRAJECTORY

In this section we present a control law allowing the system to track a real time trajectory (13) in the state space. We use the same principle for the control law design as defined previously in III (*Boolean Control*).

We define a n -dimensional space all around the desired trajectory $(x)_d(t)$ so as to minimize the switching. The control strategy in closed-loop consists in determining the configuration u at each time of intersections with the domain according to the point $(x)_d(t)$. The minimization of the commutation frequency is achieved because the same control is kept in the domain. The details of this procedure is outlined below.

Procedure

The time of switching is determined by suitable error boundaries. When the current position $(x)_p(t)$ goes outside the domain, the next configuration $Config_i(u)$ of input vector u is determined by the methodology discussed above in III (*Boolean Control*). The algorithm is explained as follows :

- ◆ In the first case, a domain D_s is defined around the point $(x)_d(t)$.
- ◆ We study if the point $(x)_p(t)$ is inside the domain D_s :

- if $(x)_p \notin D_s$, then the determination of the sequence u is made by using the method presented in section III (**Boolean Control**).

- if $(x)_p \in D_s$, then the same previous step control corresponding to $(x)_p$ when it was outside of the domain D_r is applied.

The same algorithm is restarted at each sampling time. This procedure is illustrated by the following example (figure 15):

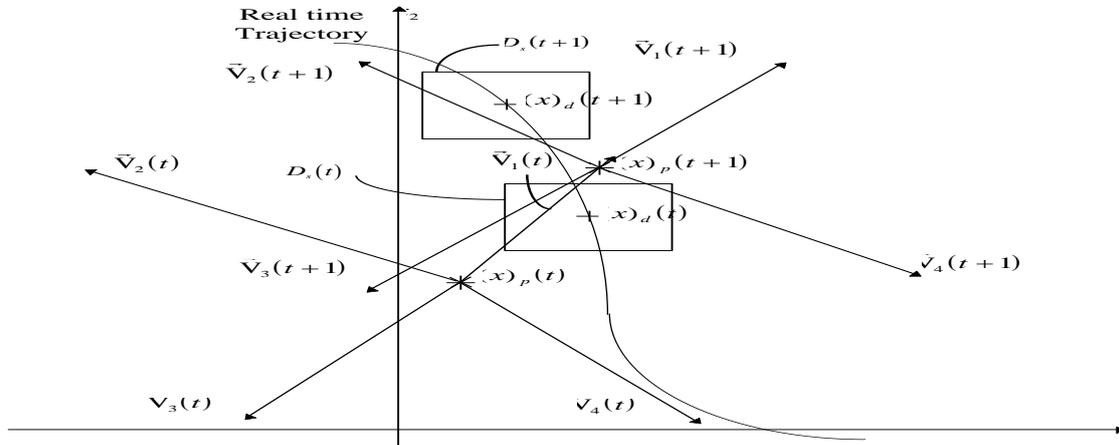


Fig.15: Tracking in state space

In this example at the instant t the vector $\vec{v}_1(t)$ is chosen and at $t+1$ the vector $\vec{v}_2(t+1)$ will be selected.

Application

The proposed control law is illustrated on the same electronic circuit as described previously (figure 9).

Simulation:

The desired vector $(x)_d$ has been simulated by two sinusoidal waveforms such as :

$$x_d = \begin{pmatrix} x_{d1} = 3 + \frac{1}{2} \sin(6.t) \\ x_{d2} = 0.4 + \frac{1}{5} \sin(3.t) \end{pmatrix}$$

These equations simulate the vector $(x)_d$ in real time. Note that frequency of the waveform must respect the bandwidth of the system.

Figure 16 shows the trajectory in the state space

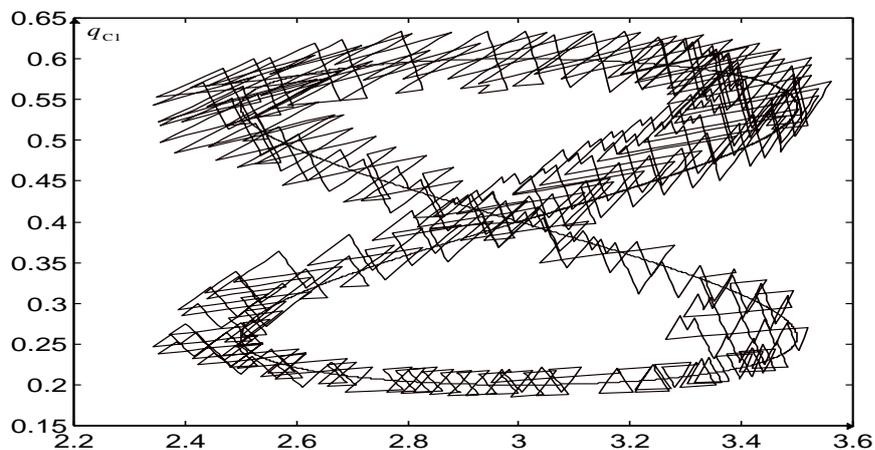


Fig.16 : Trajectory in the state space

The magnetic linkage ϕ_{L1} in the inductor and the electrical charge q_{C1} in the capacitor, with the corresponding desired trajectory are represented figure 17 and figure 18.

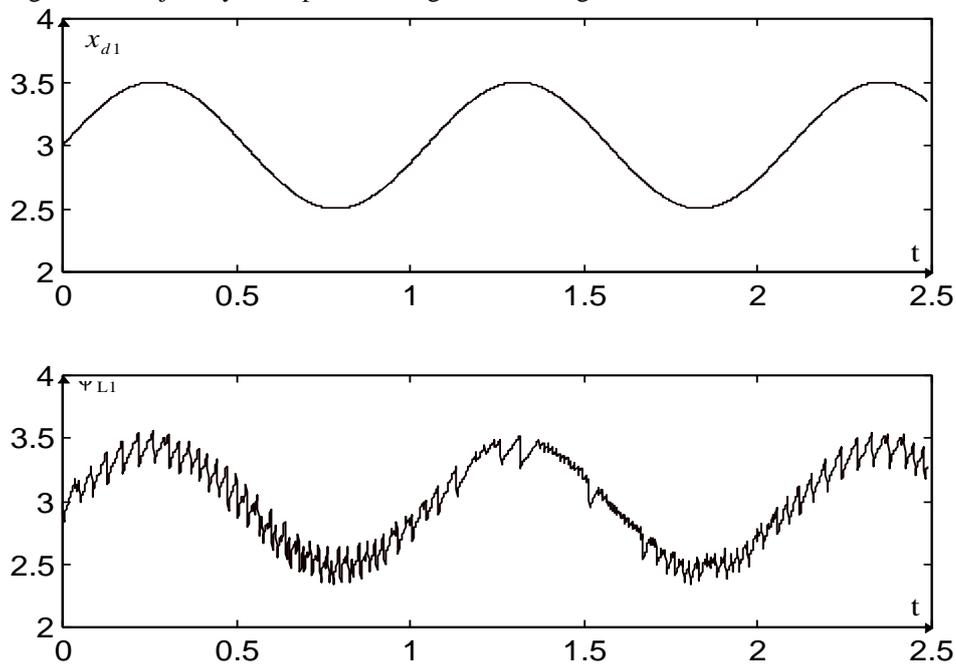


Fig .17 : Time variation of the magnetic linkage ϕ_{L1} compared to the desired trajectory

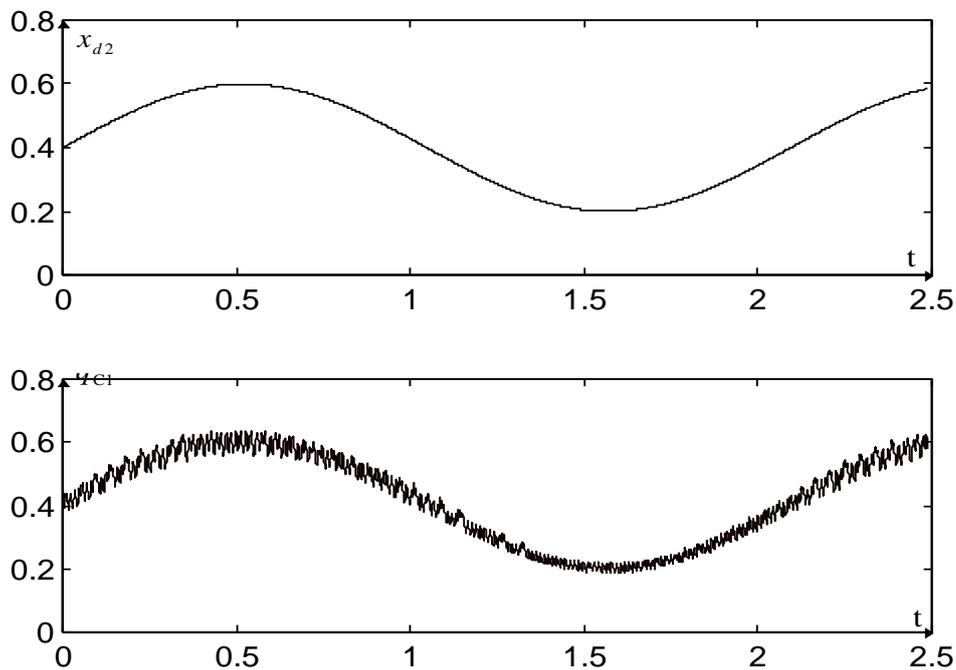


Fig .18 : Time variation of the electric charge q_{C1} compared to the desired trajectory

Figure 19 represents the Boolean control :

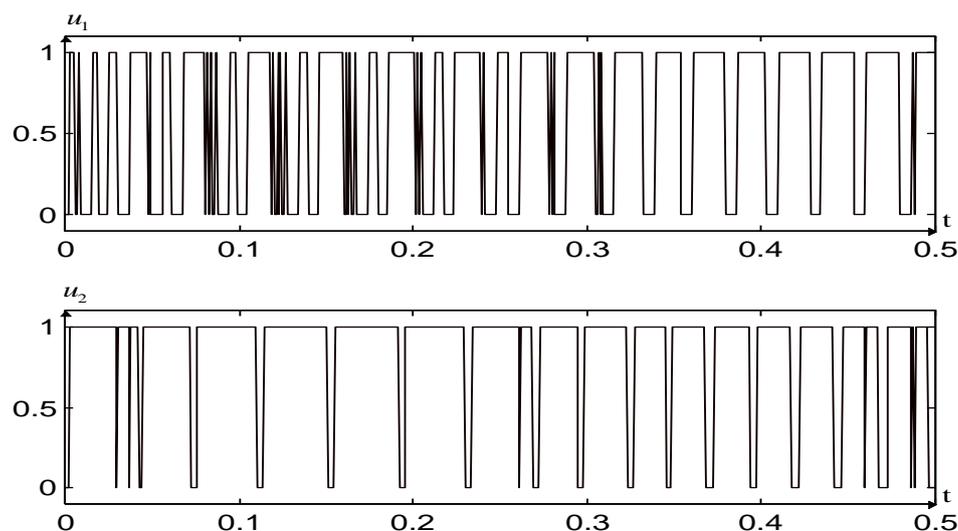


Fig .19 : Boolean controls

VI. CONCLUSION

In this paper a method is proposed for the control design of systems with Boolean inputs. The problem of regulation and tracking has been discussed. The proposed control law has been applied in the initially time to follow a predefined trajectory from the initial state and the second time to track a real time trajectory.

In both cases some remarks have been presented about the domains. If these domains are very large then we reduce the commutation numbers but we have low precision. Conversely if these domains are very small, the commutations are enhanced and we have high precision. The domains can be defined by the specification sheets.

The Bond-Graph approach has been used in this paper. One of this major advantage for Boolean control is to represent the model by a unique state equation including control vector. Indeed this control vector is characterized by the Boolean value. The other technique has drawback to obtain a multi-model representation, and it is less practicable when the number of switches items rises.

The improvements of this approach presented in this paper are :

- To consider whole model which give a direct link for Boolean control system.
- To reduce the switching frequency in order to have energy low in the commutation.

The same example has been used in the two cases to illustrate the control scheme. Simulation curves have shown that the vector $(x)_p$ reaches the desired value $(x)_d$.

The proposed strategy seems a few similar with the Direct Self Control (DSC). However the DSC technique is applied to power converter in order to control the electrical motor, while our method is considered as a whole method for Boolean input system as well as hydraulic system (valve control), chemical processes and so on. In addition a control criterion has been developed which minimize angle, and the news technique for tracking trajectory (predefined and real time trajectory) has been presented.

Currently new studies are performed in the laboratories. Their goal are to define a minimization frequency switching criterion and other part are to compare different strategies of Boolean control presented in introduction.

REFERENCES

- 1). Seguiet G., Labrique F. (1989).
"Les convertisseur de l'électronique de puissance", Editions technique et documentation Lavoisier.
- 2). S Nonaka et Y Neba (1987).
"Analysis of a P.W.M GTO current source inverter fed induction motor drive system", IEEE Transaction on Industry Applications, Vol. 23, No 2.
- 3). H Sira-Ramirez(1989).
"Sliding regimes in general non linear systems : a relative degree approach", Int. J. Control, Vol. 50, No 4, pp.1487-1506.
- 4). J E.Slotine(1984).
"Sliding controller for non-linear systems", Int. J. Control, Vol. 40, No 2, pp.421-434.
- 5). Abadie V (1994).

- "Commande des systèmes continus a entrées binaire. Application aux machines electriques", Thesis of UNIVERSITE DES SCIENCES ET TECHNIQUE DE LILLE No 1310. *In french.*
- 6). Habetler T.G., Profumo F., Pastorelli M., Tolbert L.M.(1992).
"Direct torque control of induction machines using space vector modulation", IEEE Transaction on Industry Applications, Vol. 28, No 5.
 - 7). W .Leonhard, (1985)
"Control of electrical drives", Springer, Berlin
 - 8). Depenbrock M. (1988).
"Direct self control of inverter fed induction machines", IEEE Transaction on Power Electronics, Vol. 3, No 4.
 - 9). Abadie V. Dauphin-Tanguy G (1993).
"Opened Loop control of switching linear system", Journal of the Franklin Institute, Vol. 330, No 5, pp.799-813.
 - 10). Holderbaum W., Dauphin-Tanguy G ., Borne P (1998).
"Boolean control for linear system", I.S.I.A.C International Symposium on Intelligent Automation and control, Wac'98 Anchorage , USA .
 - 11). Borne P., Dauphin-Tanguy G., Richard J.P., Rotella F., Zambettakis I.(1992).
"Modelisation et identification des processus" Tome 1. Collection : méthode et pratique de l'ingénieur. Edition Technip.
 - 12). Ducreux J.P., Castelain A., Dauphin-Tanguy G. and Rombaut C. (1992),
"Power electronics and electrical machines modelling using Bond-Graph ", IMACS Transactions on "Bond-Graph for Engineers "(eds. Dauphin-Tanguy G. and Breedveld P.) Elsevier, NewYork
 - 13). Holderbaum W., Dauphin-Tanguy G ., Borne P (1998).
"Tracking control problem for switching linear system", CESA'98 IEEE/SMC Conference Hammamet (Tunisia).