Correct-by-Design Control of 5-level and 7-level Power Converters

Gilles Feld, Laurent Fribourg, Denis Labrousse,
Bertrand Revol, Romain Soulat

December 2012

Research report LSV-12-25

Laboratoire Spécification & Vérification

École Normale Supérieure de Cachan
61, avenue du Président Wilson
94235 Cachan Cedex France

Correct-by-Design Control of 5-level and 7-level Power Converters

G. Feld
SATIE, ENS Cachan & CNRS

L. Fribourg
LSV, ENS Cachan & CNRS

D. Labrousse
SATIE, ENS Cachan & CNRS

B. Revol
SATIE, ENS Cachan & CNRS

R. Soulat
LSV, ENS Cachan & CNRS

ABSTRACT

High-power converters based on elementary switching cells are more and more used in the industry of power electronics owing to various advantages such as lower voltage stress and reduced power loss. However, the complexity of controlling such converters is a major challenge that the power manufacturing industry has to face with. The synthesis of industrial switching controllers relies today on heuristic rules and empiric simulation. The state of the system is not guaranteed to stay within the limits that are admissible for its correct electrical behavior. We show here how to apply a formal method in order to synthesise a correct-by-design control that guarantees that the power converter will always stay within a predefined safe zone of variations for its input parameters. Our method finds local invariants by decomposing the safety space into smaller zones. The method is applied in order to synthesize correct-by-design controls for a 5-level and 7-level power converters. We check the validity of our approach by numerical simulations and physical experimentations done with a prototype built by SATIE laboratory.

1. INTRODUCTION

The general function of a multilevel power converter is to synthesize a desired voltage from several levels of direct current (DC) voltage. For this reason, multilevel power converters can easily provide the high power required of a large electric drive (see [12]). There are different possible topologies for multilevel power converters: neutral-point clamped, cascaded H-bridge, flying capacitor... (see [11]). We focus here on the flying capacitor topology [8]. Schematically, a multilevel converter is made of n capacitors $C_1, \ldots, C_n$ and n switching cells $S_1, \ldots, S_n$ (as well as n opposite switching cells which are in complementary positions). This is schematized on Fig. 1. According to the positions of S_i (the high-side switch conducting position is indicated by 1 and the lowside switch conducting position by 0), one is able to fraction the load voltage v_0. By controlling the global position of the switches during a simple fixed time-stepping procedure, it is then possible to generate a staircase voltage with n levels that approximate a sinusoidal waveform (see Fig. 2 for $n = 5$).

The problem which arises is to select the appropriate switching control strategy among a number of combinations of switch positions which increase exponentially with the number n of levels (and pairs of switches). A crucial additional difficulty comes from the fact that, in order to be admissible, the control of the switching cells must guarantee that the voltages across the cell-capacitors are constrained within a certain range defined by the device blocking voltage rating. The control must then guarantee a *safety property*, called "capacitor voltage balancing": the voltage of each individual capacitor should stay inside a limited predefined interval.

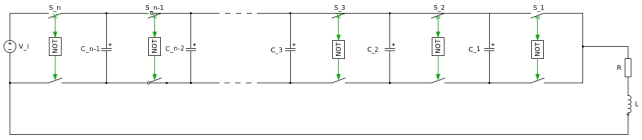


Figure 1: Electrical scheme of an n-level converter

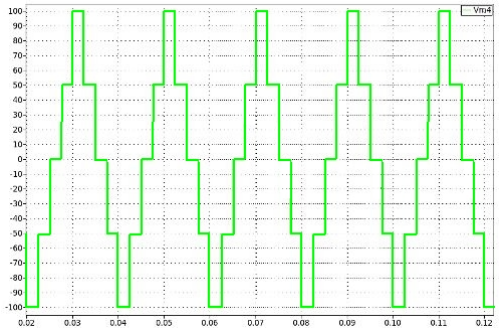


Figure 2: Ideal output for a 5-level converter

In power electronics manufacturing, switching controllers that observe such safety requirements, are synthesized today using heuristic rules and empiric criteria which are tested only for specific initial values of the voltage parameters (see, e.g., [14]). There is no formal guarantee that, during time, the system voltages will not exceed the acceptable rates. It

is therefore interesting to apply formal methods in order to produce correct-by-design controllers.

These latter years, a number of authors have designed correct-by-design control synthesis formal methods. Basically, they extend Ramadge-Wonham's method [10] for computing "maximal invariant subsets" in the context of finite transition systems. These extensions are either *direct* or *indirect* ([2]), depending on whether they work on the infinite state space (as in [13]) or on a finite abstraction (as in [6]). Direct methods are "backward-oriented": they start from a given safety zone, and apply iteratively the inverse of the transition relation associated to the modeled system, until an invariant subspace is generated. This may pose problems of numerical stability (see [9]) in the context of contractive hybrid systems (which is the case of multilevel converters). Indirect methods suffer from an explosion of the size of finite abstraction, when one wants to obtain a good precision (however, see [3] for overcoming this problem).

We apply instead here a *direct forward-oriented* method that works by iteratively decomposing the given safety space until local invariants are found (see [5] for a general description of our method). The method is successfully applied here to a 5-level and 7-level power converters. The interest of our approach is emphasized by conclusive numerical simulations as well as physical experimentations done with a prototype built by SATIE laboratory.

Plan

In Section 2, we give the model of sampled switched systems and the principle of our control synthesis method. In Section 3, we apply our method for 5-level and 7-level converters, and give numerical simulations. In Section 4, we present physical experimentations done with a prototype built by the SATIE laboratory. We conclude in Section 5.

2. MODEL OF SAMPLED SWITCHED LINEAR SYSTEMS

2.1 Model

We model power electronic as "switched systems" (as defined, e.g., in [6]), which are a subclass of hybrid systems [7].

DEFINITION 1. *A* switched linear system Σ *is a couple* $(\mathbb{R}^n, P)$ *where:*

- $\mathbb{R}^n$ *is the state space*
- $P = \{1, \ldots, m\}$ *is a finite set of* modes
- *For all* $p \in P$*, a differential equation* $\dot{x} = A_p x + b_p$ *where* A_p *is a* $(n \times n)$*-matrix of constant elements* $(a_{i,j})_p$*,* b_p *is a* n*-vector of constant elements* $(b_k)_p$ *and* x *the state variable.*

We will use $\mathbf{x}(t, x_0, p)$ to denote the point reached at time $t \in \mathbb{R}_{\geq 0}$ from the initial condition x_0 under the mode p.

In the following, as in [6], we will work with trajectories of duration τ for some chosen $\tau \in \mathbb{R}_{\geq 0}$, called "time sampling parameter". In particular, we suppose that switching instants can only occur at times of the form $i\tau$ with $i \in \mathbb{N}$. Such switched systems are said to be *sampled*. The associated transition relation $\rightarrow_\tau^i$ is given by:

$$x \rightarrow_\tau^i x' \text{ iff } \mathbf{x}(\tau, x, i) = x' \text{ for } x \text{ and } x' \text{ in } \mathbb{R}^n$$

We define: $Post_i(X) = \{x' \mid x \rightarrow_\tau^i x' \text{ for some } x \in X\}$.
We say that a subspace X of R is R-invariant via mode i if

$$Post_i(X) \subseteq R.$$

We will generally group together sequences of modes as "patterns". Given a pattern Pat of the form $(i_1 \ldots i_m)$, we define:

$Post_{Pat}(X) = \{x' \mid x \rightarrow_\tau^{i_1} \cdots \rightarrow_\tau^{i_m} x' \text{ for some } x \in X\}$.
We say that X is R-invariant via a pattern Pat if $Post_{Pat}(X) \subseteq R$. Note that, when X is convex, in order to show that X is R-invariant via Pat, it suffices to show that every vertex of X is mapped by Pat to a point of R.

2.2 Principle of the control synthesis method

As explained below, the set of admissible patterns that allow to produce a stair-case form for the output signal corresponds to a path in a pre-defined graph. There are 576 admissible patterns for $n = 5$ and $518,400$ patterns for $n = 7$. In order to find a state-dependent control that outputs a stair-case signal while maintaining the capacitor voltages in a safe pre-defined zone R, we adopt the following state-decomposition method:

- we first decompose R into a covering set of sub-zones $\{V_i\}$ (with $R = \cup_i V_i$).
- for each zone V_i, we identify a pattern Pat_i that lets V_i R-invariant.

This guarantees that the system can be controlled in order to stay always in R at each beginning of cycle: if the starting state x_0 is in R, this means that $X \in V_i$ for some i; one thus applies Pat_i to x_0, which gives a new state x_1 which is guaranteed to be still in S (since V_i is R-invariant via Pat_i), and the control process can be guaranteed. Note that, with this method, the system can temporarily exit from R during the execution of a cycle. In practice, as the simulations will show, the system stays in R even during the cycle. Here R is rectangular box of dimension $n-2$, and the decomposition of R is simply obtained by dichotomously decomposing R into 2^{n-2} sub-boxes of identical size (2^3 boxes for $n = 5$ and 2^5 boxes for $n = 7$). The 2^{n-2} patterns that let the sub-boxes invariant are found by a random generate-and-test procedure.

3. APPLICATION TO MULTILEVEL CONVERTERS

3.1 5-level Converter

The electrical scheme of a 5-level converter is given in Figure 3. The system can output 5 different levels of voltage $-1, -\frac{1}{2}, 0, \frac{1}{2}, 1$. The goal of the system is to output a stair-case signal of the form presented in Figure 2.

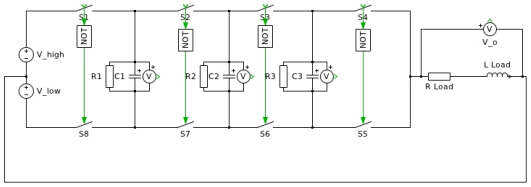


Figure 3: Electrical scheme of the 5-level converter

Let v_1 (resp. v_2, v_3) be the voltage across the capacitor C_1 (resp. C_2, C_3), R_{Load} the resistor of the load, L_{Load} (abbreviated hereafter as L) the inductor of the load, v_{high} and v_{low} the input voltage. The mode of the system is characterized by the position (0 or 1) of the switching cells S_1, S_2, S_3, S_4, i.e., by the value of vector $S = \begin{pmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{pmatrix}$.[1] There are thus $2^3 = 8$ modes. We will use the numerical values:

- $v_{high} = v_{low} = 100\text{V}$
- $R_{Load} = 50\Omega$
- $C_1 = C_2 = C_3 = 0.0012\text{F}$
- $L = 0.20\text{H}$
- $R_1 = R_2 = R_3 = R_4 = 20,000\Omega$

The state of the system is $x(t) = [v_1(t), v_2(t), v_3(t), i(t)]^T$ where $v_1(t)$ (resp. $v_2(t)$, $v_3(t)$) is the voltage across the capacitor C_1 (resp. C_2, C_3) and $i(t)$ is the current flowing in the circuit. This model of 5-level converter can be seen as a switched system. Given a mode, i.e. a certain value of vector S, the associated dynamics is indeed of the form $\dot{x}(t) = A_S x(t) + b_S$, since we have:

$\dot{X} = A_S.X + b_S$ with

$$A_S = \begin{pmatrix} -\frac{1}{R_1.C_1} & 0 & 0 & \frac{S_1-S_2}{C_1} \\ 0 & -\frac{1}{R_2.C_2} & 0 & \frac{S_2-S_3}{C_2} \\ 0 & 0 & -\frac{1}{R_3.C_3} & \frac{S_3-S_4}{C_3} \\ \frac{S_2-S_1}{L} & \frac{S_3-S_2}{L} & \frac{S_4-S_3}{L} & -\frac{R_{Load}}{L} \end{pmatrix}$$

$$\text{and } b_S = \begin{pmatrix} 0 \\ 0 \\ 0 \\ S_1 \frac{v_{high}+v_{low}}{L} \end{pmatrix}$$

By controlling the modes at each sampling time, one can synthesize a 5-level staircase function. This is schematized by the graph depicted on Fig. 4. The nodes of the graph are labelled by the configurations of the switching cells S_1, S_2,S_3 and S_4 (mode 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111). Each path represents a possible sequence of control for 1 cycle, leading from voltage $-v_{low}$ (state 0000) to voltage $+v_{high}$ (state 1111) through voltages $-\frac{1}{2}.v_{low}$, 0, $\frac{1}{2}.v_{high}$ then back to voltage $-v_{low}$ (state 0000) through voltages $\frac{1}{2}.v_{high}$, 0, $\frac{1}{2}.v_{low}$. There are thus 576 possible sequences of control for generating a 5-level staircase signal on 1 cycle. These sequences of control are called *patterns*, denoted by $\text{Pat}_1, \ldots, \text{Pat}_{576}$.

[1] Besides, we have: $S_5 = \neg S_1$, $S_6 = \neg S_2$, $S_7 = \neg S_3$ and $S_8 = \neg S_4$.

Here, it is convenient to consider the 5-level converter, not as a τ-sampled transition system with $2^4 = 16$ modes, but, more globally, as a T-sampled transition systems where T is the duration of one cycle of the staircase signal (i.e. 8τ), and modes correspond to the 576 patterns. The control problem is now to find a strategy for deciding, at each beginning of cycle, which Pat_i ($1 \leq i \leq 576$) to apply in order to maintain all the capacitor voltages within a predefined limited zone.

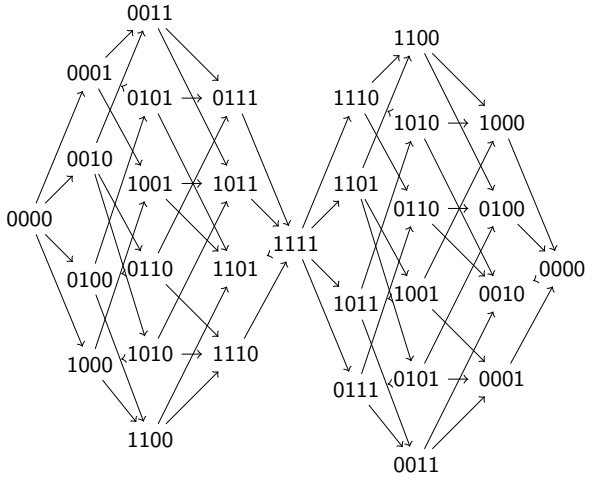


Figure 4: Transition graph corresponding to a cycle of 5-level staircase signal

We consider a 5-level inverter that will output ideally a staircase waveform with an amplitude of 200V, centered around 0V (i.e., $v_1 = 150V$, $v_2 = 100V$, $v_3 = 50V$). We consider that a variation of $\pm 5V$ is admissible as it represents a variation of 10% on the least charged capacitor C_3. The admissible area R is hence defined as $R = [145, 155] \times [95, 105] \times [45, 55] \times [i_{min}, i_{max}]$, where i_{min} and i_{max} are the physical limits for i. It is interesting to notice that at each beginning of a cycle the value of i is the same. This suggests to take a state-dependent control, that takes into account only the capacitor voltages v_1, v_2, v_3 and not the value of the current intensity i. This is interesting because as noted in [4], "For practical applications, a current sensor is not desired". Thus, we will focus on the voltage dimensions of R and disregard the intensity dimension ($R = [145, 155] \times [95, 105] \times [45, 55]$).

We decompose dichotomously R into 8 subset V_i of equal size such that $\bigcup_{i=1}^{8} V_i = R$:

- $V_1 = [145, 150] \times [95, 100] \times [45, 50]$
- $V_2 = [145, 150] \times [95, 100] \times [50, 55]$
- $V_3 = [145, 150] \times [100, 105] \times [45, 50]$
- $V_4 = [145, 150] \times [100, 105] \times [50, 55]$

- $V_5 = [150, 155] \times [95, 100] \times [45, 50]$
- $V_6 = [150, 155] \times [95, 100] \times [50, 55]$
- $V_7 = [150, 155] \times [100, 105] \times [45, 50]$
- $V_8 = [150, 155] \times [100, 105] \times [50, 55]$

Using a standard random generate-and-test program, we find that the following patterns Pat_j (defined as paths of the transition graph) let V_j R-invariant:

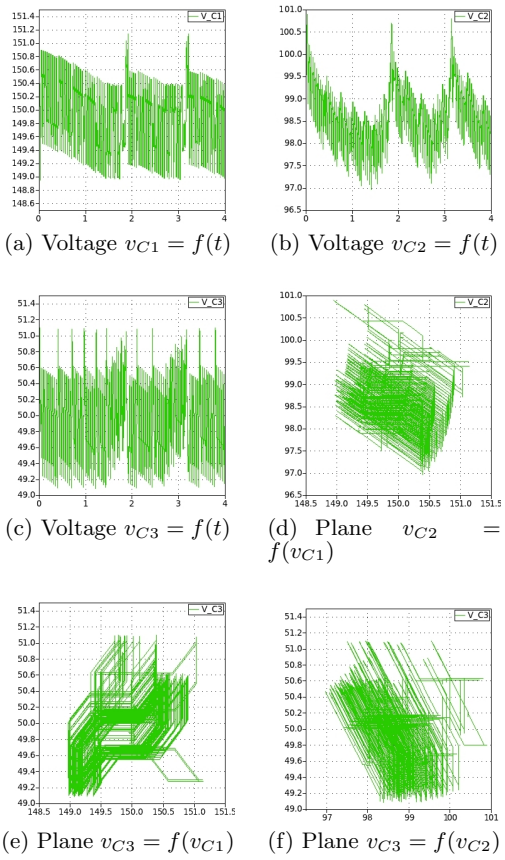
- Pat_1: (0000 → 0001 → 0101 → 1101 → 1111 → 1101 → 0101 → 0001 → 0000)
- Pat_2: (0000 → 0100 → 0101 → 1101 → 1111 → 1101 → 0101 → 0100 → 0000)
- Pat_3: (0000 → 0001 → 0011 → 1011 → 1111 → 1011 → 0011 → 0001 → 0000)
- Pat_4: (0000 → 0010 → 0011 → 1011 → 1111 → 1011 → 0011 → 0010 → 0000)
- Pat_5: (0000 → 1000 → 1010 → 1110 → 1111 → 1110 → 1010 → 1000 → 0000)
- Pat_6: (0000 → 1000 → 1100 → 1101 → 1111 → 1101 → 1100 → 1000 → 0000)
- Pat_7: (0000 → 0100 → 0110 → 0111 → 1111 → 0111 → 0110 → 0100 → 0000)
- Pat_8: (0000 → 1000 → 1010 → 1011 → 1111 → 1011 → 1010 → 1000 → 0000)

We present in Figures 5 and 6 a simulation of this controller on the system starting from the point $v_{C1}(0) = 150V, v_{C2}(0) = 100V, v_{C3}(0) = 50V$ and $i(0) = -3A$. Formally, when starting from R, the state of the controlled system is guaranteed to return in R at each beginning of cycle, but it might temporarily exit from R during the execution of the electrical cycle. However in practice, the simulations show that the system always stays within R.

3.2 7-level Converter

In this section, we are interested in a more complex model for the converter. We will consider a 7-level converter that can generate waveform going from $-v_{low}$ up to $+v_{high}$ with steps at $-v_{low} + kv_{high}/3$ for $k = 0, \ldots, 6$. The electrical scheme is presented in Figure 7. We used the following values for the system constants:

- Output at 50Hz
- All the capacitance equal to $0.1F$
- The resistor values 50Ω
- The inductor values $0.137H$
- $v_{low} = v_{high} = 300V$ (giving an output between $-300V$ and $+300V$)



(a) Voltage $v_{C1} = f(t)$ (b) Voltage $v_{C2} = f(t)$

(c) Voltage $v_{C3} = f(t)$ (d) Plane $v_{C2} = f(v_{C1})$

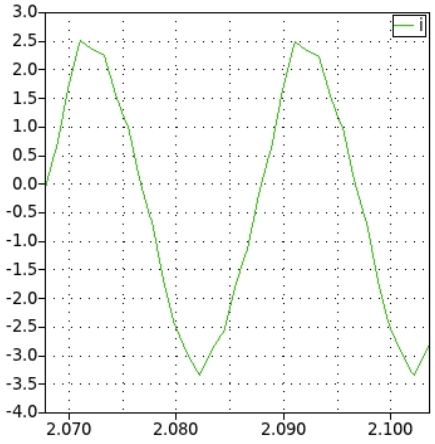
(e) Plane $v_{C3} = f(v_{C1})$ (f) Plane $v_{C3} = f(v_{C2})$

Figure 5: Capacitor voltages

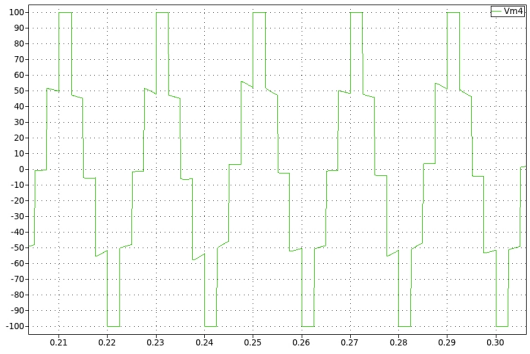
Forn $n = 7$, there are $518,400$ possible sequences of control (patterns) for generating a n-level staircase signal on 1 cycle. We consider a 7-level inverter that will output ideally a staircase waveform with an amplitude of 600V, centered around 0V (i.e., $v_1 = 500V$, $v_2 = 400V$, $v_3 = 300V$, $v_4 = 200V$, $v_5 = 100V$). We consider that a variation of $\pm 5V$ is admissible as it represents a variation of 5% on the least charged capacitor C_5. The admissible area R is hence defined as $R = [495, 505] \times [395, 405] \times [295, 305] \times [195, 205] \times [95, 105] \times [i_{min}, i_{max}]$, where i_{min} and i_{max} are the physical limits for i. As stated before, we do not need to take into account the intensity in our analysis since at every beginning of a new pattern the intensity has the same value. Thus, we will focus on the voltage dimensions of R and disregard the intensity dimension ($R = [495, 505] \times [395, 405] \times [295, 305] \times [195, 205] \times [95, 105]$).

The decomposition of R into sub-zones $\{V_j\}_j$ and the corresponding set of patterns $\{Pat_j\}_j$ are given in Appendix.

We present in Figures 8 and 9 a simulation of this controller on the system starting from the point $v_{C1}(0) = 500V$, $v_{C2}(0) = 400V$, $v_{C3}(0) = 300V$, $v_{C4}(0) = 200V$, $v_{C5}(0) = 100V$ and $i(0) = -2.5A$. Formally, when starting from R, the state of the controlled system is guaranteed to return in



(a) Current i



(b) Output voltage v_o

Figure 6: Current and output voltage

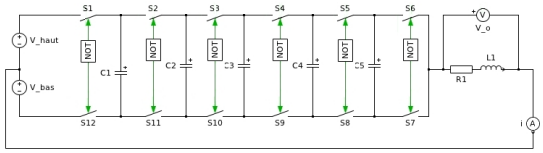


Figure 7: Electrical scheme of the 7-level converter

R at each beginning of cycle. However, the simulations show that the system never leaves R (not even temporarily).

4. PHYSICAL EXPERIMENTATIONS

A prototype of the 5-level flying capacitor has been realized by the SATIE Laboratory in order to test our control strategy on an actual system. See Figure 10 for a picture of the system. Our control strategy was applied to the system via Simulink and a dSpace® interface. The results are presented in Figure 11 for the output voltage and the capacitor charges. In Figure 12, we present the same results but with a larger scale on the capacitor voltage to see the fluctuations around the reference values. As we can see, this is very close to what was obtained by the simulation with PLECS [1]. In Figure 13, we represent the output voltage in green together with the current in yellow (after appropriate resizing)

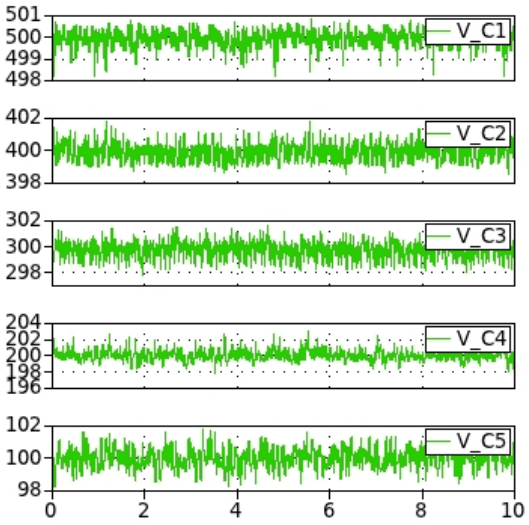
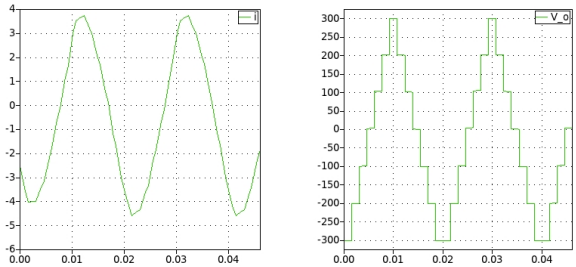


Figure 8: Capacitor voltages



(a) Current i (b) Current v_o

Figure 9: Current and output voltage

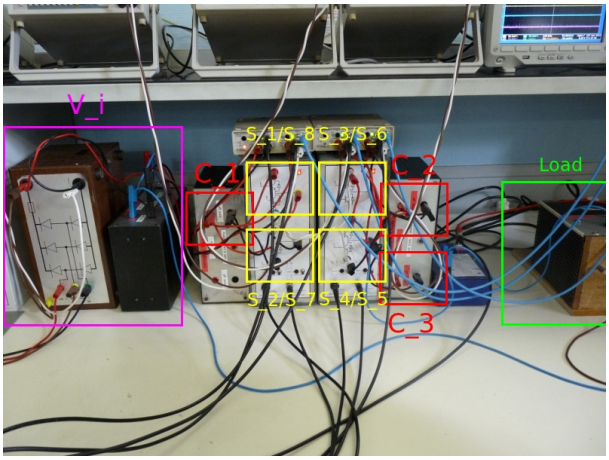


Figure 10: Actual system build by SATIE

flowing the load.

During the experimentations, we have tested the robustness

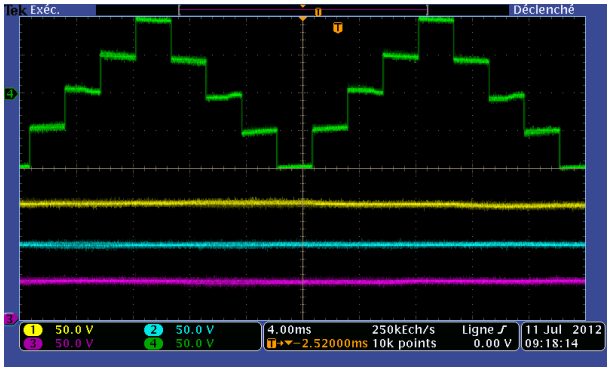


Figure 11: Output voltage and capacitor voltages

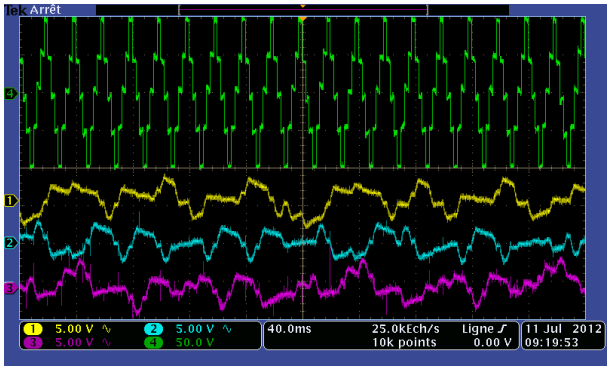


Figure 12: Zoom of output and capacitors voltages

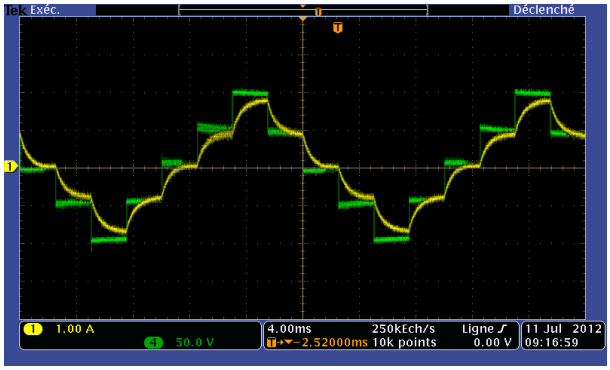


Figure 13: Output voltage and current (after appropriate resizing) in the circuit

of our solution to the following modifications of the system:

- The ideal voltage source as input is no longer ideal but its values fluctuate around the reference value.
- The system does not start from the reference valuations for the capacitor voltages and the input voltage, but the input voltage increases gradually until reaching the desired value, and the capacitor are discharged at $t = 0$.
- The switching strategy is robust to slight changes of the resistor load, the inductor load or both.
- We tested what happens if we apply the same pattern two times in a row instead of picking a new one at each beginning of a cycle: the system presents more fluctuations around the desired values but the capacitor voltages balancing is maintained.
- Recall that the standard strategy is to select at each T a new pattern according to the current state of the system (if the system is in V_j, one apply Pat_j). Surprisingly, if we apply this strategy not periodically every T seconds (at each beginning of cycle), but following an arbitrary sequence of durations $T_1, T_2, T_3, \ldots$, the capacitor voltages remain balanced (although at some slightly different mean value). This is depicted in Figure 14. (Each vertical line is an artifact corresponding to a pattern switch.)

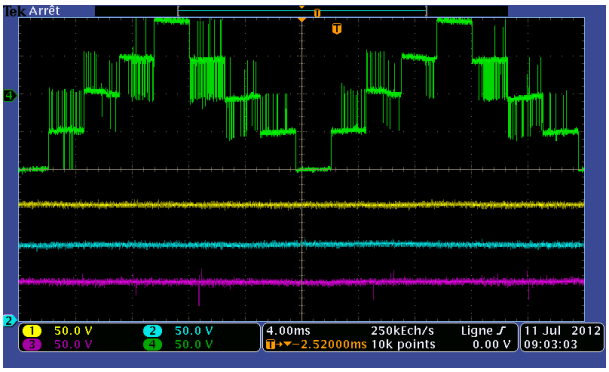


Figure 14: Output voltage and capacitor voltages

5. CONCLUSION

We have synthesized a control strategy for n-level ($n = 5,\ 7$) power converters using an original forward-oriented formal method. This control is state-dependent and is interesting because:

- at each control time, the controller indicates the $2(n-1)$ subsequent switching modes (instead of just the next switching mode),
- the controller takes into account only the capacitor voltages state and not the intensity state.

We have checked by numerical simulations and physical experimentations that the control satisfies the capacitor voltage balancing and the staircase shape of the output voltage. We have also checked the robustness of the method with respect to several sources of perturbation. The method is general and can be applied in principle to any n-level converter, with n bigger than 7.

In future work, we plan to test further the robustness of our control strategy under variations of the resistive and inductive load, in order to model the time-varying load of electrical networks.

Acknowledgement. We are grateful to Stéphane Lefebvre for numerous helpful discussions. This work has been done within the framework of projects BOOST and BOOST2 supported by Institut Farman.

6. REFERENCES

[1] PLECS Web page. http://www.plexim.com.

[2] E. Asarin, O. Bournez, T. Dang, O. Maler, and A. Pnueli. Effective Synthesis of Switching Controllers for Linear Systems. *Proceedings of the IEEE, Special Issue on Hybrid Systems*, 88:1011–1025, 2000.

[3] J. Cámara, A. Girard, and G. Gößler. Safety controller synthesis for switched systems using multi-scale symbolic models. In *Proc. 50th IEEE Conference on Decision and Control (CDC-ECC '11)*, pages 520–525, 2011.

[4] Z. Du, L. Tolbert, B. Ozpineci, and J. Chiasson. Fundamental frequency switching strategies of a seven-level hybrid cascaded h-bridge multilevel inverter. *IEEE Transactions on Power Electronics*, 24(1):25 –33, jan. 2009.

[5] L. Fribourg and R. Soulat. Controlled recurrent subspaces for sampled switched linear systems. Research Report LSV-12-24, Dec. 2012. 10 pages.

[6] A. Girard, G. Pola, and P. Tabuada. Approximately bisimilar symbolic models for incrementally stable switched systems. *IEEE Trans. on Automatic Control*, 55:116–126, 2010.

[7] T. A. Henzinger. The theory of hybrid automata. In *Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science*, LICS '96, pages 278–292, Washington, DC, USA, 1996. IEEE Computer Society.

[8] T. Meynard and H. Foch. Multi-level conversion: high voltage choppers and voltage-source inverters. In *23rd Annual IEEE Power Electronics Specialists Conference (PESC '92)*, pages 397 –403 vol.1, jun-3 jul 1992.

[9] I. M. Mitchell. Comparing forward and backward reachability as tools for safety analysis. In *Proceedings of the 10th international conference on Hybrid systems: computation and control*, HSCC'07, pages 428–443, Berlin, Heidelberg, 2007. Springer-Verlag.

[10] P. J. G. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of The IEEE*, 77:81–98, 1989.

[11] J. Rodriguez, S. Bernet, B. Wu, J. Pontt, and S. Kouro. Multilevel voltage-source-converter topologies for industrial medium-voltage drives. *IEEE Transactions on Industrial Electronics*, 54(6):2930 –2945, dec. 2007.

[12] L. M. Tolbert, F. Z. Peng, and T. G. Habetler. Multilevel converters for large electric drives. *IEEE Trans. on Industry Applications*, 35(1):36–44, 1999.

[13] C. Tomlin, J. Lygeros, and S. Shankar Sastry. A game theoretic approach to controller design for hybrid systems. *Proceedings of the IEEE*, 88(7):949 –970, july 2000.

[14] L. Zhang and S. J. Watkins. Capacitor voltage balancing in multilevel flying capacitor inverters by rule-based switching pattern selection. *IET Electr. Power Appl.*, 1(3):116–126, 2007.

APPENDIX

A. DECOMPOSITION OF THE 7-LEVEL CONVERTER

The dichotomous decomposition of R gives a decomposition into $2^5 = 32$ sub-zones V_j. The V_js and Pat_js are defined as follows:

- V_1: $([495, 500] \times [395, 400] \times [295, 300] \times [195, 200] \times [95, 100])$
- V_2: $([495, 500] \times [395, 400] \times [295, 300] \times [195, 200] \times [100, 105])$
- V_3: $([495, 500] \times [395, 400] \times [295, 300] \times [200, 205] \times [95, 100])$
- V_4: $([495, 500] \times [395, 400] \times [295, 300] \times [200, 205] \times [100, 105])$
- V_5: $([495, 500] \times [395, 400] \times [300, 305] \times [195, 200] \times [95, 100])$
- V_6: $([495, 500] \times [395, 400] \times [300, 305] \times [195, 200] \times [100, 105])$
- V_7: $([495, 500] \times [395, 400] \times [300, 305] \times [200, 205] \times [95, 100])$
- V_8: $([495, 500] \times [395, 400] \times [300, 305] \times [200, 205] \times [100, 105])$
- V_9: $([495, 500] \times [400, 405] \times [295, 300] \times [195, 200] \times [95, 100])$
- V_{10}: $([495, 500] \times [400, 405] \times [295, 300] \times [195, 200] \times [100, 105])$
- V_{11}: $([495, 500] \times [400, 405] \times [295, 300] \times [200, 205] \times [95, 100])$
- V_{12}: $([495, 500] \times [400, 405] \times [295, 300] \times [200, 205] \times [100, 105])$
- V_{13}: $([495, 500] \times [400, 405] \times [300, 305] \times [195, 200] \times [95, 100])$
- V_{14}: $([495, 500] \times [400, 405] \times [300, 305] \times [195, 200] \times [100, 105])$
- V_{15}: $([495, 500] \times [400, 405] \times [300, 305] \times [200, 205] \times [95, 100])$
- V_{16}: $([495, 500] \times [400, 405] \times [300, 305] \times [200, 205] \times [100, 105])$
- V_{17}: $([500, 505] \times [395, 400] \times [295, 300] \times [195, 200] \times [95, 100])$
- V_{18}: $([500, 505] \times [395, 400] \times [295, 300] \times [195, 200] \times [100, 105])$
- V_{19}: $([500, 505] \times [395, 400] \times [295, 300] \times [200, 205] \times [95, 100])$
- V_{20}: $([500, 505] \times [395, 400] \times [295, 300] \times [200, 205] \times [100, 105])$
- V_{21}: $([500, 505] \times [395, 400] \times [300, 305] \times [195, 200] \times [95, 100])$
- V_{22}: $([500, 505] \times [395, 400] \times [300, 305] \times [195, 200] \times [100, 105])$
- V_{23}: $([500, 505] \times [395, 400] \times [300, 305] \times [200, 205] \times [95, 100])$
- V_{24}: $([500, 505] \times [395, 400] \times [300, 305] \times [200, 205] \times [100, 105])$

- V_{25}: $([500, 505] \times [400, 405] \times [295, 300] \times [195, 200] \times [95, 100])$
- V_{26}: $([500, 505] \times [400, 405] \times [295, 300] \times [195, 200] \times [100, 105])$
- V_{27}: $([500, 505] \times [400, 405] \times [295, 300] \times [200, 205] \times [95, 100])$
- V_{28}: $([500, 505] \times [400, 405] \times [295, 300] \times [200, 205] \times [100, 105])$
- V_{29}: $([500, 505] \times [400, 405] \times [300, 305] \times [195, 200] \times [95, 100])$
- V_{30}: $([500, 505] \times [400, 405] \times [300, 305] \times [195, 200] \times [100, 105])$
- V_{31}: $([500, 505] \times [400, 405] \times [300, 305] \times [200, 205] \times [95, 100])$
- V_{32}: $([500, 505] \times [400, 405] \times [300, 305] \times [200, 205] \times [100, 105])$.

- Pat_1: (000000 → 000001 → 000101 → 010101 → 110101 → 111101 → 111111 → 111101 → 110101 → 010100 → 000101 → 000001 → 000000)
- Pat_2: (000000 → 000010 → 000110 → 010110 → 110110 → 111110 → 111111 → 111110 → 110110 → 010110 → 000110 → 000010 → 000000)
- Pat_3: (000000 → 000001 → 001001 → 011001 → 111001 → 111011 → 111111 → 111011 → 111001 → 011001 → 001001 → 000001 → 000000)
- Pat_4: (000000 → 000010 → 001010 → 011010 → 111010 → 111011 → 111111 → 111011 → 111010 → 011010 → 001010 → 000010 → 000000)
- Pat_5: (000000 → 001000 → 011000 → 111000 → 111100 → 111110 → 111111 → 111110 → 111100 → 111000 → 011000 → 001000 → 000000)
- Pat_6: (000000 → 001000 → 011000 → 111000 → 111001 → 111101 → 111111 → 111101 → 111001 → 111000 → 011000 → 001000 → 000000)
- Pat_7: (000000 → 001000 → 011000 → 111000 → 111010 → 111011 → 111111 → 111011 → 111010 → 111000 → 011000 → 001000 → 000000)
- Pat_8: (000000 → 001000 → 011000 → 111000 → 111001 → 111011 → 111111 → 111011 → 111001 → 111000 → 011000 → 001000 → 000000)
- Pat_9: (000000 → 000001 → 000011 → 000111 → 001111 → 101111 → 111111 → 101111 → 001111 → 000111 → 000011 → 000001 → 000000)
- Pat_{10}: (000000 → 000010 → 000011 → 000111 → 001111 → 101111 → 111111 → 101111 → 001111 → 000111 → 000011 → 000010 → 000000)
- Pat_{11}: (000000 → 000001 → 000101 → 000111 → 001111 → 101111 → 111111 → 101111 → 001111 → 000111 → 000101 → 000001 → 000000)
- Pat_{12}: (000000 → 000100 → 000110 → 000111 → 001111 → 101111 → 111111 → 101111 → 001111 → 000111 → 000110 → 000100 → 000000)
- Pat_{13}: (000000 → 000001 → 000011 → 001011 → 001111 → 101111 → 111111 → 101111 → 001111 → 001011 → 000011 → 000001 → 000000)
- Pat_{14}: (000000 → 000010 → 000011 → 001011 → 001111 → 101111 → 111111 → 101111 → 001111 → 001011 → 000011 → 000010 → 000000)
- Pat_{15}: (000000 → 100000 → 110000 → 111000 → 111100 → 111110 → 111111 → 101111 → 100111 → 100011 → 100001 → 100000 → 000000)
- Pat_{16}: (000000 → 100000 → 110000 → 111000 → 111100 → 111110 → 111111 → 101111 → 100111 → 100011 → 100001 → 100000 → 000000)
- Pat_{17}: (000000 → 000001 → 000011 → 000111 → 001111 → 011111 → 111111 → 111110 → 111100 → 111000 → 110000 → 100000 → 000000)
- Pat_{18}: (000000 → 000010 → 000110 → 001110 → 011110 → 011111 → 111111 → 111110 → 011110 → 001110 → 000110 → 000010 → 000000)
- Pat_{19}: (000000 → 000100 → 001100 → 001101 → 001111 → 011111 → 111111 → 011111 → 001111 → 001101 → 001100 → 000100 → 000000)
- Pat_{20}: (000000 → 000100 → 001100 → 001110 → 001111 → 011111 → 111111 → 011111 → 001111 → 001110 → 001100 → 000100 → 000000)
- Pat_{21}: (000000 → 000001 → 001001 → 001011 → 001111 → 011111 → 111111 → 011111 → 001111 → 001011 → 001001 → 000001 → 000000)
- Pat_{22}: (000000 → 000010 → 001010 → 001011 → 001111 → 011111 → 111111 → 011111 → 001111 → 001011 → 001010 → 000010 → 000000)
- Pat_{23}: (000000 → 000001 → 001001 → 001101 → 001111 → 011111 → 111111 → 011111 → 001111 → 001101 → 001001 → 000001 → 000000)
- Pat_{24}: (000000 → 001000 → 001100 → 001110 → 001111 → 011111 → 111111 → 011111 → 001111 → 001110 → 001100 → 001000 → 000000)
- Pat_{25}: (000000 → 000001 → 000011 → 000111 → 001111 → 011111 → 111111 → 011111 → 001111 → 000111 → 000011 → 000001 → 000000)
- Pat_{26}: (000000 → 000010 → 000011 → 000111 → 001111 → 011111 → 111111 → 011111 → 001111 → 000111 → 000011 → 000010 → 000000)
- Pat_{27}: (000000 → 000001 → 000101 → 000111 → 001111 → 011111 → 111111 → 011111 → 001111 → 000111 → 000101 → 000001 → 000000)
- Pat_{28}: (000000 → 000100 → 000110 → 000111 → 001111 → 011111 → 111111 → 011111 → 001111 → 000111 → 000110 → 000100 → 000000)
- Pat_{29}: (000000 → 000001 → 000011 → 001011 → 001111 → 011111 → 111111 → 011111 → 001111 → 001011 → 000011 → 000001 → 000000)
- Pat_{30}: (000000 → 000010 → 000011 → 001011 → 001111 → 011111 → 111111 → 011111 → 001111 → 001011 → 000011 → 000010 → 000000)
- Pat_{31}: (000000 → 100000 → 101000 → 101001 → 101011 → 101111 → 111111 → 101111 → 101011 → 101001 → 101000 → 100000 → 000000)
- Pat_{32}: (000000 → 100000 → 101000 → 101010 → 101011 → 101111 → 111111 → 101111 → 101011 → 101010 → 101000 → 100000 → 000000)