A graphical method for SVM duty cycles computation. Application on two-level and multi-level inverters.

Santiago de Pablo, Alexis B. Rey*, Luis C. Herrero and Fernando Martínez
UNIVERSITY OF VALLADOLID
E.T.S. Ingenieros Industriales, Paseo del Cauce, s/n.
47011 Valladolid, Spain.
(*) TECHNICAL UNIVERSITY OF CARTAGENA
Antiguo Cuartel de Antigones, Plaza del Hospital, 1.
30202 Cartagena, Murcia, Spain.
E-Mail: sanpab@eis.uva.es
URL: http://www.epYme.uva.es

Acknowledgements

Authors would like to acknowledge the financial support of the *Ministerio de Educación y Ciencia*, Spain, under grant CICYT ENE2007-67417/ALT, using FEDER funds.

Keywords

Pulse Width Modulation, Modulation strategy, Converter control, Three-phase system, VSI.

Abstract

A graphical method, easy to understand and easy to implement, has been developed for two-level and multi-level Space Vector Modulation. Switching sequences can be computed with no need of solving neither multi-variable inverse problems nor expensive trigonometric functions. Proposed method has been validated through simulations and physical implementations.

Introduction

Space Vector Modulation (SVM) [1] has been considered a reference for most analysis, comparisons and physical verifications [1]–[7]. Compared with the former three-phase sinusoidal pulsewidth modulation method (SPWM) it had (initially) the advantages of lower current harmonics and a possible higher modulation index, although these drawbacks have been easily avoided on SPWM techniques by means of distorted modulation [2]–[4]. At the end, it was demonstrated that both methods may lead to the same results [2]–[5].

Recently, an increasing interest has risen to develop multi-level inverters based on this modulation technique [8]–[14]. They are increasingly being used in high-power medium voltage applications due to their superior performance compared to two-level inverters [8]. The most used topologies are diode-clamped (neutral-point clamped or multi-point clamped), capacitor-clamped (flying capacitors) and cascaded multi-cell with separate DC sources [9], [10]. Advantages of this multi-level approach include lower switching losses, higher voltage capability, higher power quality, and better electromagnetic compatibility [10]. In addition, common mode voltage can be reduced or removed to avoid early motor failure just sacrificing voltage levels [11]. The main disadvantages are that a larger number of switching semiconductors are required for lower-voltage systems and the small voltage steps must be supplied on the DC side either by a capacitor bank or isolated voltage sources [10]. Finally, there exist redundant switching states that must be used to keep all DC sources balanced [9], [12], increasing the complexity of modulators.

There are two main approaches to obtain the switching times when using SVM. The first approach is to determine the triangle where the reference vector is located, and then solve two or three simultaneous equations for this triangle to obtain the *on*-times of each vector as in [10]. Whereas, a second approach normalizes the reference voltages using the main DC voltage and use a nonorthogonal coordinate transformation in such a way that the *on*-times are essentially the fractional part of the reference voltages on that basis [13]; equivalent equations using the usual α-β orthogonal basis can also be found [8]. The later approach solves the problem for the general *n*-level case, but gives no insight into the optimal sequence of space vector states [14] and do not justify why the *on*-times are related to the fractional part of the artificially transformed voltages; the former approach is more intuitive and gives more control to implement different strategies, for example, to balance DC voltages or reduce commutations, but do not scale well as the number of level increases.

This paper proposes a very simple set of equations to compute the SVM duty cycles on two-level inverters, and gives an intuitive graphical interpretation of such equations; then it generalizes the method for small multi-level inverters avoiding the necessity of solving simultaneous equations for each triangle, as stated by Corzine [10]. The resulting equations match those proposed by Celanovic *et al.* [13] for the general *n*-level case, so this development may help to better understand their method.

Classical two-level SVM

The structure of a typical three-phase two-level voltage source inverter (VSI) is shown at Fig. 1-a. The relationship between switching variables (S_a, S_b, S_c) and phase-to-neutral voltages (V_{an}, V_{bn}, V_{cn}) is given by (1), where V_{dc} is the bus voltage and S_k is '1' or '0' when the upper or lower transistor of phase k is *on*, respectively [10]. It satisfies that $V_{an} + V_{bn} + V_{cn} = 0$. Applying the Clarke transformation to these voltages, equation (2), it leads to a well known space vector $\boldsymbol{V}$ with the same instantaneous information in a stationary reference frame α-β.

$$\begin{bmatrix} V_{an} \\ V_{bn} \\ V_{cn} \end{bmatrix} = \frac{1}{3} V_{dc} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \cdot \begin{bmatrix} S_a \\ S_b \\ S_c \end{bmatrix} \qquad (1)$$

$$\boldsymbol{V} = \begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & \frac{-1}{3} & \frac{-1}{3} \\ 0 & \frac{\sqrt{3}}{3} & \frac{-\sqrt{3}}{3} \end{bmatrix} \cdot \begin{bmatrix} V_{an} \\ V_{bn} \\ V_{cn} \end{bmatrix} \qquad (2)$$

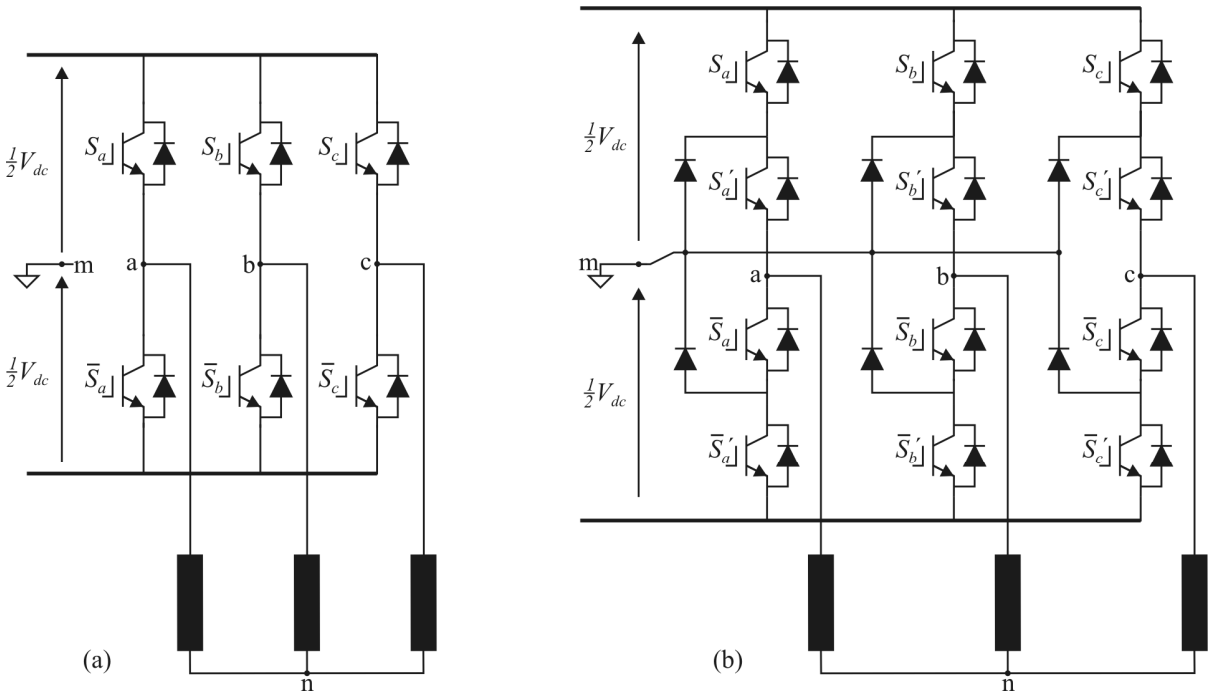

Fig. 1: Three-phase inverters: (a) a two-level inverter and (b) a diode-clamped three-level inverter.

There are eight different vectors for the two-level inverter output, named $\boldsymbol{V_0}$ to $\boldsymbol{V_7}$ as defined in Fig. 2, according to the eight possible switching states. To obtain the required output space vector $\boldsymbol{v_o}^*$, that usually is a rotating vector with constant module and pulsation, conduction times of the inverter switches are modulated according to the angle and magnitude of that reference. The null vectors $\boldsymbol{V_0}$ and $\boldsymbol{V_7}$, and two space vectors adjacent to $\boldsymbol{v_o}^*$, $\boldsymbol{V_i}$ and $\boldsymbol{V_{i+1}}$, are chosen and modulated during a short switching time T_S using the volt-second balance method:

$$d_0 \cdot \boldsymbol{V}_0 + d_i \cdot \boldsymbol{V}_i + d_{i+1} \cdot \boldsymbol{V}_{i+1} + d_7 \cdot \boldsymbol{V}_7 \cong \frac{1}{T_S} \int_t^{t+T_S} \boldsymbol{v}_o^* dt \cong \boldsymbol{v}_o^* \tag{3}$$

These duty cycles can be computed using polar coordinates (see [1] for further details) by means of equations (4) to (6) [1], [6] where θ_r is the relative angle between $\boldsymbol{v_o}^*$ and $\boldsymbol{V_i}$ ($0 \le \theta_r \le \pi/3$), and m is the modulation index ($0 \le m \le 1$) defined as $m = \sqrt{3} \cdot V_o / V_{dc}$, where V_o is the magnitude of $\boldsymbol{v_o}^*$.

$$d_i = m \cdot \sin(\tfrac{\pi}{3} - \theta_r) \tag{4}$$

$$d_{i+1} = m \cdot \sin(\theta_r) \tag{5}$$

$$d_0 + d_7 = 1 - d_i - d_j \tag{6}$$

This method calculates all information needed to generate suitable switching sequences (usually $d_0 = d_7$), but its straightforward implementation needs the computation of two 'sin(x)' functions [6], thus they require either big lookup tables or a time consuming process. In addition, the sector of $\boldsymbol{v_o}^*$ must be determined in order to compute θ_r and select $\boldsymbol{V_i}$ and $\boldsymbol{V_{i+1}}$.

A known improvement for the computation of SVM duty cycles, actually the usual method to compute them, is the use of rectangular coordinates instead of polar ones [3]. As the values of d_0 and d_7 have no effect in (3) because of the null magnitude of $\boldsymbol{V_0}$ and $\boldsymbol{V_7}$, solving this equation using rectangular coordinates it yields to (7):

$$\begin{bmatrix} d_i \\ d_j \end{bmatrix} = \begin{bmatrix} V_{i\alpha} & V_{j\alpha} \\ V_{i\beta} & V_{j\beta} \end{bmatrix}^{-1} \cdot \begin{bmatrix} v_{o\alpha}^* \\ v_{o\beta}^* \end{bmatrix} \tag{7}$$

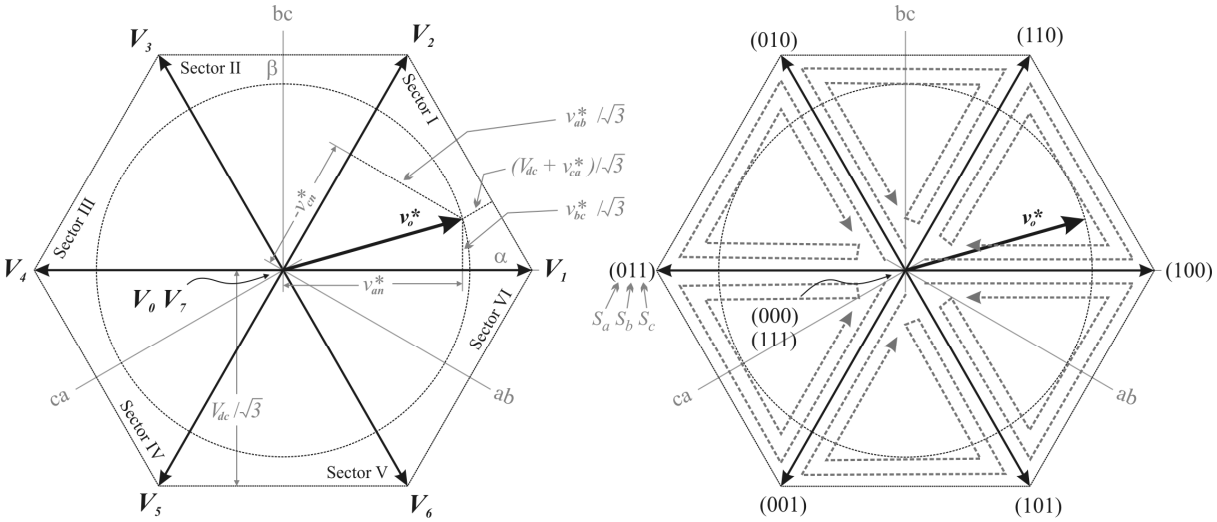

Fig. 2: At left, space vectors for the voltages of a two-level VSI and several values used to compute SVM duty cycles; at right, switching values and a set of switching sequences for all sectors.

 ISBN: 9789075815009

where the matrix arranges the α-β components of the adjacent inverter vectors $\boldsymbol{V_i}$ and $\boldsymbol{V_{i+1}}$. Equation (7) can be applied, without loss of generality, to a reference phase voltage between 0 and π/3, thus $\boldsymbol{V_1}$ and $\boldsymbol{V_2}$ will be used jointly with $\boldsymbol{V_0}$ and $\boldsymbol{V_7}$ to generate $\boldsymbol{v_o^*}$, as shown at (8). Thus, the amount of time for each voltage vector is computed by solving the inverse problem, leading to (9) and (10).

$$\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} \frac{2}{3}V_{dc} & \frac{1}{3}V_{dc} \\ 0 & \frac{\sqrt{3}}{3}V_{dc} \end{bmatrix}^{-1} \cdot \begin{bmatrix} v_{o\alpha}^* \\ v_{o\beta}^* \end{bmatrix} \qquad (8)$$

$$d_1 = T_1 / T_S = \frac{\frac{3}{2}\cdot v_{o\alpha}^* - \frac{\sqrt{3}}{2}\cdot v_{o\beta}^*}{V_{dc}} \qquad (9)$$

$$d_2 = T_2 / T_S = \frac{\sqrt{3}\cdot v_{o\beta}^*}{V_{dc}} \qquad (10)$$

Proposed analytical and graphical method for two-level inverters

Although equations (9), (10) and (6) would be the fastest method to compute SVM duty cycles on two-level inverters when using a voltage reference expressed in the stationary reference frame α-β, we may simplify these equations further using the Clarke transformation: as stated on [4] (see also [2] for a different approach), applying (2) on (9), (10) and (6) it yields to an astonishing simple result:

$$d_1 = \frac{\frac{3}{2}\cdot v_{o\alpha}^* - \frac{\sqrt{3}}{2}\cdot v_{o\beta}^*}{V_{dc}} = \frac{v_{an}^* - v_{bn}^*}{V_{dc}} = \frac{v_{ab}^*}{V_{dc}} \qquad (11)$$

$$d_2 = \frac{\sqrt{3}\cdot v_{o\beta}^*}{V_{dc}} = \frac{v_{bn}^* - v_{cn}^*}{V_{dc}} = \frac{v_{bc}^*}{V_{dc}} \qquad (12)$$

$$d_0 + d_7 = 1 - d_1 - d_2 = 1 + \frac{v_{cn}^* - v_{an}^*}{V_{dc}} = 1 + \frac{v_{ca}^*}{V_{dc}} \qquad (13)$$

where $v_{ab}^* = v_{an}^* - v_{bn}^*$, $v_{bc}^* = v_{bn}^* - v_{cn}^*$ and $v_{ca}^* = v_{cn}^* - v_{an}^*$ are the corresponding line-to-line reference voltages. Similar expressions have been found for all phase sectors [4], thus they are arranged on table I. These expressions show clearly that SVM duty cycles can be computed easily using the phase components of the voltage reference, and also that commutation sequences will be identical when phase-to-neutral voltages are used or line-to-line voltages are preferred instead.

Table I: Proposed expressions to compute SVM duty cycles on two-level inverters.

Sector	Duty cycles and vector sequences	
I 0–π/3	$d_A = v_{ab}^*/V_{dc}$	$\boldsymbol{V_A} = 100$
	$d_B = v_{bc}^*/V_{dc}$	$\boldsymbol{V_B} = 110$
	$d_Z = 1 + v_{ca}^*/V_{dc}$	$\boldsymbol{V_Z} = 000+111$
II π/3–2π/3	$d_A = -v_{ab}^*/V_{dc}$	$\boldsymbol{V_A} = 010$
	$d_B = -v_{ca}^*/V_{dc}$	$\boldsymbol{V_B} = 110$
	$d_Z = 1 - v_{bc}^*/V_{dc}$	$\boldsymbol{V_Z} = 000+111$
III 2π/3–π	$d_A = v_{bc}^*/V_{dc}$	$\boldsymbol{V_A} = 010$
	$d_B = v_{ca}^*/V_{dc}$	$\boldsymbol{V_B} = 011$
	$d_Z = 1 + v_{ab}^*/V_{dc}$	$\boldsymbol{V_Z} = 000+111$

Sector	Duty cycles and vector sequences	
IV π–4π/3	$d_A = -v_{bc}^*/V_{dc}$	$\boldsymbol{V_A} = 001$
	$d_B = -v_{ab}^*/V_{dc}$	$\boldsymbol{V_B} = 011$
	$d_Z = 1 - v_{ca}^*/V_{dc}$	$\boldsymbol{V_Z} = 000+111$
V 4π/3–5π/3	$d_A = v_{ca}^*/V_{dc}$	$\boldsymbol{V_A} = 001$
	$d_B = v_{ab}^*/V_{dc}$	$\boldsymbol{V_B} = 101$
	$d_Z = 1 + v_{bc}^*/V_{dc}$	$\boldsymbol{V_Z} = 000+111$
VI 5π/3–2π	$d_A = -v_{ca}^*/V_{dc}$	$\boldsymbol{V_A} = 100$
	$d_B = -v_{bc}^*/V_{dc}$	$\boldsymbol{V_B} = 101$
	$d_Z = 1 - v_{ab}^*/V_{dc}$	$\boldsymbol{V_Z} = 000+111$

 ISBN: 9789075815009

Compared to (9), (10) and (6), equations (11) to (13) are simpler, thus they allow a better understanding of the underlying processes (see also [4]). Indeed, a graphical method to compute SVM duty cycles has been developed using these results, as explained following.

The inverse Clarke transformation, equation (14), yields three phase-to-neutral voltages from their α-β representation of $\boldsymbol{v_o}^*$. Essentially, this computation is equivalent to three scalar products of such space vector $\boldsymbol{v_o}^*$ by three unitary vectors with phases 0º, 120º and 240º, respectively. The meaning of this operation is that phase-to-neutral voltages can be computed as the length from the origin to the perpendicular projection of $\boldsymbol{v_o}^*$ over three axes at 0º, 120º and 240º, respectively (see the left part of Fig. 2). When line-to-line voltages are used, equation (15) is derived from (14), projections are drawn over three axes at –30º, 90º and 210º for *ab*, *bc*, and *ca* components, respectively, and a scaling factor of $\sqrt{3}$ must be considered (see the left part of Fig. 2 again).

$$\begin{bmatrix} v_{an}^* \\ v_{bn}^* \\ v_{cn}^* \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{-1}{2} & \frac{\sqrt{3}}{2} \\ \frac{-1}{2} & \frac{-\sqrt{3}}{2} \end{bmatrix} \cdot \begin{bmatrix} v_{o\alpha}^* \\ v_{o\beta}^* \end{bmatrix} \qquad (14)$$

$$\begin{bmatrix} v_{ab}^* \\ v_{bc}^* \\ v_{ca}^* \end{bmatrix} = \sqrt{3} \begin{bmatrix} \frac{\sqrt{3}}{2} & \frac{-1}{2} \\ 0 & 1 \\ \frac{-\sqrt{3}}{2} & \frac{-1}{2} \end{bmatrix} \cdot \begin{bmatrix} v_{o\alpha}^* \\ v_{o\beta}^* \end{bmatrix} \qquad (15)$$

Therefore, a graphical interpretation of (11) to (13) can be done, as shown in Fig. 2: on sector I, the duty cycle d_1 can be computed using (11) as stated above, but it can also be computed as the ratio between the *height* of the point $\boldsymbol{v_o}^*$ over the segment $\boldsymbol{V_0}$–$\boldsymbol{V_2}$ (the opposite of $\boldsymbol{V_1}$) and the *height* of the triangle $\boldsymbol{V_0}$–$\boldsymbol{V_1}$–$\boldsymbol{V_2}$, because the former height is $v_{ab}^* / \sqrt{3}$ and the later one is $V_{dc} / \sqrt{3}$; likewise, the second duty cycle d_2 can be computed using the *height* of the point $\boldsymbol{v_o}^*$ over the segment $\boldsymbol{V_0}$–$\boldsymbol{V_1}$ (the opposite of $\boldsymbol{V_2}$), which is $v_{bc}^* / \sqrt{3}$, thus matching (12); and finally $d_0 + d_7$ can be computed applying (6), but also using the *height* of the point $\boldsymbol{v_o}^*$ over the segment $\boldsymbol{V_1}$–$\boldsymbol{V_2}$ (the opposite of $\boldsymbol{V_0}$) which is $(V_{dc} + v_{ca}^*) / \sqrt{3}$; it must be pointed out that in this case v_{ca}^* is a negative value, so the addition to V_{dc} decreases its value.

At last, when adjacent vectors have been selected (see [4]) and duty cycles have been computed, a suitable commutation sequence must be applied. The usual choice [7] has been drawn at the right part of Fig. 2: as an example, at Sector I the selected sequence would be '000'-'100'-'110'-'111' and then '111'-'110'-'100'-'000' back during every commutation period. This method avoids unwanted commutations, because all periods start on the same null state.

Proposed graphical method for small multi-level inverters

The above graphical method can be easily applied to small multi-level inverters, where the vector diagrams are universal regardless of the type of multilevel inverter [9]: diode-clamped, capacitor-clamped and cascaded. The analysis presented in Fig. 3 is developed for a three-level inverter for simplicity and ease of understanding, but it is also valid on higher level inverters.

Using as above the three-axes *ab-bc-ca* instead of α-β, the nearest three vectors are easily found: for example, any reference vector at sector "5" of Fig. 3 (the triangle V_Z-V_A-V_B) satisfies $v_{ab}^* \geq 0$ and $v_{bc}^* \geq \frac{1}{2}V_{dc}$; likewise, all 24 triangles can be found using a decision tree with five comparisons or less. Then, duty cycles can be immediately derived from Fig. 3 leading to very simple expressions: as above, the duty cycle d_A can be computed as the ratio between the *height* of the point $\boldsymbol{v_o}^*$ over the segment $\boldsymbol{V_Z}$–$\boldsymbol{V_B}$ (the opposite of $\boldsymbol{V_A}$) and the *height* of the triangle $\boldsymbol{V_Z}$–$\boldsymbol{V_A}$–$\boldsymbol{V_B}$, the former height is $v_{ab}^* / \sqrt{3}$ and the later one is $\frac{1}{2}V_{dc} / \sqrt{3}$; likewise, the second duty cycle d_B can be computed using the *height* of the point $\boldsymbol{v_o}^*$ over the segment $\boldsymbol{V_Z}$–$\boldsymbol{V_A}$ (the opposite of $\boldsymbol{V_B}$), yielding to $(v_{bc}^* - \frac{1}{2}V_{dc}) / \sqrt{3}$; and finally d_Z can be computed using the *height* of the point $\boldsymbol{v_o}^*$ over the segment $\boldsymbol{V_A}$–$\boldsymbol{V_B}$ (the opposite of $\boldsymbol{V_Z}$) which is $(V_{dc} + v_{ca}^*) / \sqrt{3}$; in this case, v_{ca}^* is a negative value again. This technique avoids the need of solving three simultaneous equations at each triangle as stated at [10]. Furthermore, dividing all reference voltages by the half of the DC voltage we get the integer-like equations proposed by Celanovic *et al.* [13]:

$$u_{ab}^{*} \equiv \frac{v_{ab}^{*}}{\frac{1}{2}V_{dc}} \quad \Rightarrow \quad d_A = \frac{v_{ab}^{*}/\sqrt{3}}{\frac{1}{2}V_{dc}/\sqrt{3}} = u_{ab}^{*} \tag{16}$$

$$u_{bc}^{*} \equiv \frac{v_{bc}^{*}}{\frac{1}{2}V_{dc}} \quad \Rightarrow \quad d_B = \frac{(v_{bc}^{*} - \frac{1}{2}V_{dc})/\sqrt{3}}{\frac{1}{2}V_{dc}/\sqrt{3}} = u_{bc}^{*} - 1.0 \tag{17}$$

$$u_{ca}^{*} \equiv \frac{v_{ca}^{*}}{\frac{1}{2}V_{dc}} \quad \Rightarrow \quad d_Z = \frac{(V_{dc} + v_{ca}^{*})/\sqrt{3}}{\frac{1}{2}V_{dc}/\sqrt{3}} = 2.0 + u_{ca}^{*} \tag{18}$$

Extending these ideas to all 24 triangles for a three-level inverter is fast and direct, as demonstrated on the Table II. It also includes the switching sequences drawn in Fig. 4, where six main centers have been selected, so all switching periods start and finish at states ‘001’, ‘101’, ‘100’, ‘110’, ‘010’ or ‘011’. Inner triangles are divided in two halves and the difference between two line-to-line voltages is evaluated to decide which sequence must be followed; for example, on triangles “16” and “22” it checks if v_{ab}^{*} is greater than v_{ca}^{*} in order to choose the switching sequence starting at ‘101’ or ‘001’.

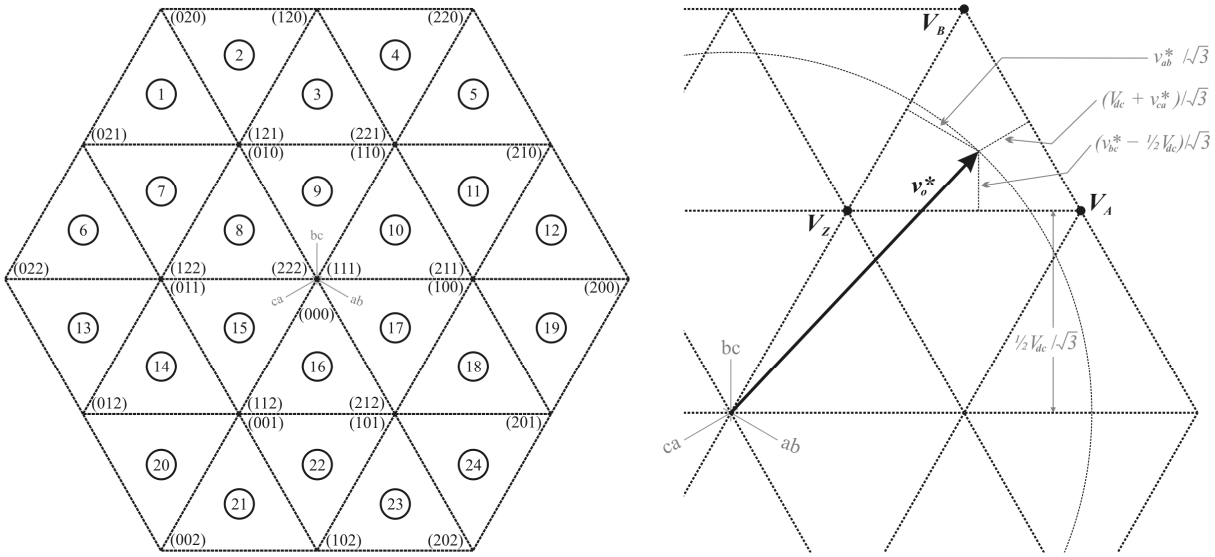

Fig. 3: At left, definition of sectors and vectors for a three-level inverter. At right, a subset of the space vector diagram used to show the application of this SVM technique to multi-level inverters.

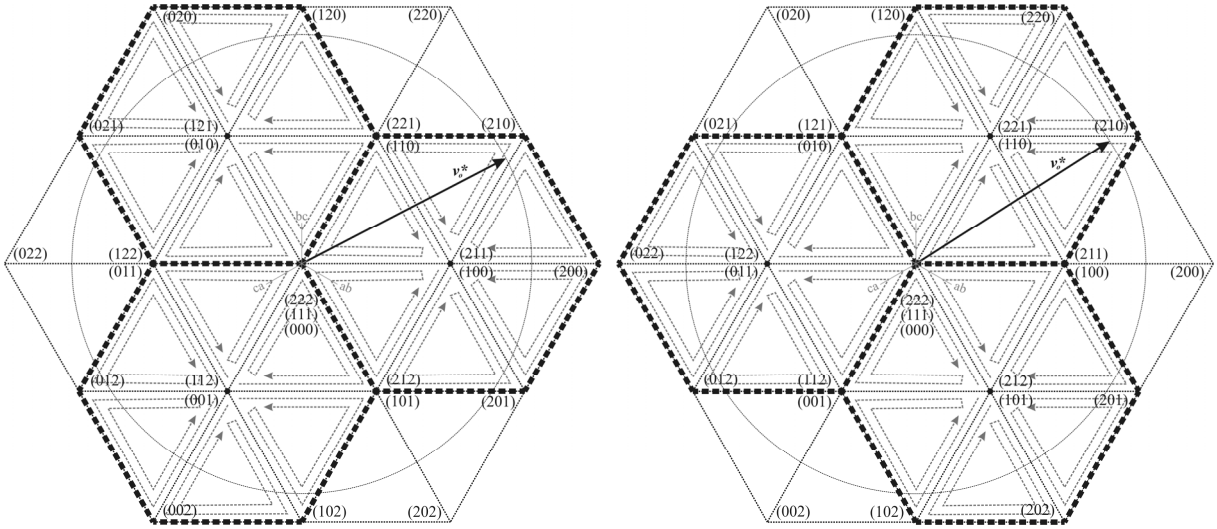

Fig. 4: Switching sequences applied on a three-level inverter. At each sampling period, different sequences are selected using the instantaneous components of the reference voltage.

Table II: Proposed expressions to compute SVM duty cycles on three-level inverters.

Sector	Conditions, duty cycles and vector sequences
1	$[(u_{bc}^* >= 1.0)\ \&\ (u_{ca}^* >= 0.0)]$ $d_Z = u_{ab}^* + 2.0 \quad d_A = u_{bc}^* - 1.0 \quad d_B = u_{ca}^*$ $\boldsymbol{V_Z} = 010{+}121 \quad \boldsymbol{V_A} = 020 \quad \boldsymbol{V_B} = 021$
2	$[(u_{ab}^* < -1.0)\ \&\ (u_{ca}^* < 0.0)]$ $d_Z = -u_{bc}^* + 2.0 \quad d_A = -u_{ab}^* - 1.0 \quad d_B = -u_{ca}^*$ $\boldsymbol{V_Z} = 010{+}121 \quad \boldsymbol{V_A} = 020 \quad \boldsymbol{V_B} = 120$
3	$[(u_{ab}^* >= -1.0)\ \&\ (u_{bc}^* >= 1.0)\ \&\ (u_{ca}^* >= -1.0)]$ if $(u_{ab}^* >= u_{ca}^*)$ $d_Z = u_{ab}^* + 1.0 \quad d_A = u_{bc}^* - 1.0 \quad d_B = u_{ca}^* + 1.0$ $\boldsymbol{V_Z} = 110{+}221 \quad \boldsymbol{V_A} = 120 \quad \boldsymbol{V_B} = 121$ else $d_Z = u_{ca}^* + 1.0 \quad d_A = u_{ab}^* + 1.0 \quad d_B = u_{bc}^* - 1.0$ $\boldsymbol{V_Z} = 010{+}121 \quad \boldsymbol{V_A} = 110 \quad \boldsymbol{V_B} = 120$
4	$[(u_{ab}^* < 0.0)\ \&\ (u_{ca}^* < -1.0)]$ $d_Z = -u_{bc}^* + 2.0 \quad d_A = -u_{ab}^* \quad d_B = -u_{ca}^* - 1.0$ $\boldsymbol{V_Z} = 110{+}221 \quad \boldsymbol{V_A} = 120 \quad \boldsymbol{V_B} = 220$
5	$[(u_{ab}^* >= 0.0)\ \&\ (u_{bc}^* >= 1.0)]$ $d_Z = u_{ca}^* + 2.0 \quad d_A = u_{ab}^* \quad d_B = u_{bc}^* - 1.0$ $\boldsymbol{V_Z} = 110{+}221 \quad \boldsymbol{V_A} = 210 \quad \boldsymbol{V_B} = 220$
6	$[(u_{bc}^* >= 0.0)\ \&\ (u_{ca}^* >= 1.0)]$ $d_Z = u_{ab}^* + 2.0 \quad d_A = u_{bc}^* \quad d_B = u_{ca}^* - 1.0$ $\boldsymbol{V_Z} = 011{+}122 \quad \boldsymbol{V_A} = 021 \quad \boldsymbol{V_B} = 022$
7	$[(u_{ab}^* < -1.0)\ \&\ (u_{bc}^* < 1.0)\ \&\ (u_{ca}^* < 1.0)]$ if $(u_{ca}^* >= u_{bc}^*)$ $d_Z = -u_{bc}^* + 1.0 \quad d_A = -u_{ab}^* - 1.0 \quad d_B = -u_{ca}^* + 1.0$ $\boldsymbol{V_Z} = 011{+}122 \quad \boldsymbol{V_A} = 021 \quad \boldsymbol{V_B} = 121$ else $d_Z = -u_{ca}^* + 1.0 \quad d_A = -u_{bc}^* + 1.0 \quad d_B = -u_{ab}^* - 1.0$ $\boldsymbol{V_Z} = 010{+}121 \quad \boldsymbol{V_A} = 011 \quad \boldsymbol{V_B} = 021$
8	$[(u_{ab}^* >= -1.0)\ \&\ (u_{bc}^* >= 0.0)\ \&\ (u_{ca}^* >= 0.0)]$ if $(u_{ca}^* >= u_{bc}^*)$ $d_Z = u_{ca}^* \quad d_A = u_{ab}^* + 1.0 \quad d_B = u_{bc}^*$ $\boldsymbol{V_Z} = 011{+}122 \quad \boldsymbol{V_A} = 111 \quad \boldsymbol{V_B} = 121$ else $d_Z = u_{bc}^* \quad d_A = u_{ca}^* \quad d_B = u_{ab}^* + 1.0$ $\boldsymbol{V_Z} = 010{+}121 \quad \boldsymbol{V_A} = 011 \quad \boldsymbol{V_B} = 111$
9	$[(u_{ab}^* < 0.0)\ \&\ (u_{bc}^* < 1.0)\ \&\ (u_{ca}^* < 0.0)]$ if $(u_{ab}^* >= u_{ca}^*)$ $d_Z = -u_{ca}^* \quad d_A = -u_{bc}^* + 1.0 \quad d_B = -u_{ab}^*$ $\boldsymbol{V_Z} = 110{+}221 \quad \boldsymbol{V_A} = 111 \quad \boldsymbol{V_B} = 121$ else $d_Z = -u_{ab}^* \quad d_A = -u_{ca}^* \quad d_B = -u_{bc}^* + 1.0$ $\boldsymbol{V_Z} = 010{+}121 \quad \boldsymbol{V_A} = 110 \quad \boldsymbol{V_B} = 111$
10	$[(u_{ab}^* >= 0.0)\ \&\ (u_{bc}^* >= 0.0)\ \&\ (u_{ca}^* >= -1.0)]$ if $(u_{bc}^* >= u_{ab}^*)$ $d_Z = u_{bc}^* \quad d_A = u_{ca}^* + 1.0 \quad d_B = u_{ab}^*$ $\boldsymbol{V_Z} = 110{+}221 \quad \boldsymbol{V_A} = 111 \quad \boldsymbol{V_B} = 211$ else $d_Z = u_{ab}^* \quad d_A = u_{bc}^* \quad d_B = u_{ca}^* + 1.0$ $\boldsymbol{V_Z} = 100{+}211 \quad \boldsymbol{V_A} = 110 \quad \boldsymbol{V_B} = 111$
11	$[(u_{ab}^* < 1.0)\ \&\ (u_{bc}^* < 1.0)\ \&\ (u_{ca}^* < -1.0)]$ if $(u_{bc}^* >= u_{ab}^*)$ $d_Z = -u_{ab}^* + 1.0 \quad d_A = -u_{ca}^* - 1.0 \quad d_B = -u_{bc}^* + 1.0$ $\boldsymbol{V_Z} = 110{+}221 \quad \boldsymbol{V_A} = 210 \quad \boldsymbol{V_B} = 211$ else $d_Z = -u_{bc}^* + 1.0 \quad d_A = -u_{ab}^* + 1.0 \quad d_B = -u_{ca}^* - 1.0$ $\boldsymbol{V_Z} = 100{+}211 \quad \boldsymbol{V_A} = 110 \quad \boldsymbol{V_B} = 210$
12	$[(u_{ab}^* >= 1.0)\ \&\ (u_{bc}^* >= 0.0)]$ $d_Z = u_{ca}^* + 2.0 \quad d_A = u_{ab}^* - 1.0 \quad d_B = u_{bc}^*$ $\boldsymbol{V_Z} = 100{+}211 \quad \boldsymbol{V_A} = 200 \quad \boldsymbol{V_B} = 210$
13	$[(u_{ab}^* < -1.0)\ \&\ (u_{bc}^* < 0.0)]$ $d_Z = -u_{ca}^* + 2.0 \quad d_A = -u_{bc}^* \quad d_B = -u_{ab}^* - 1.0$ $\boldsymbol{V_Z} = 011{+}122 \quad \boldsymbol{V_A} = 012 \quad \boldsymbol{V_B} = 022$
14	$[(u_{ab}^* >= -1.0)\ \&\ (u_{bc}^* >= -1.0)\ \&\ (u_{ca}^* >= 1.0)]$ if $(u_{bc}^* >= u_{ab}^*)$ $d_Z = u_{bc}^* + 1.0 \quad d_A = u_{ca}^* - 1.0 \quad d_B = u_{ab}^* + 1.0$ $\boldsymbol{V_Z} = 011{+}122 \quad \boldsymbol{V_A} = 012 \quad \boldsymbol{V_B} = 112$ else $d_Z = u_{ab}^* + 1.0 \quad d_A = u_{bc}^* + 1.0 \quad d_B = u_{ca}^* - 1.0$ $\boldsymbol{V_Z} = 001{+}112 \quad \boldsymbol{V_A} = 011 \quad \boldsymbol{V_B} = 012$
15	$[(u_{ab}^* < 0.0)\ \&\ (u_{bc}^* < 0.0)\ \&\ (u_{ca}^* < 1.0)]$ if $(u_{bc}^* >= u_{ab}^*)$ $d_Z = -u_{ab}^* \quad d_A = -u_{ca}^* + 1.0 \quad d_B = -u_{bc}^*$ $\boldsymbol{V_Z} = 011{+}122 \quad \boldsymbol{V_A} = 111 \quad \boldsymbol{V_B} = 112$ else $d_Z = -u_{bc}^* \quad d_A = -u_{ab}^* \quad d_B = -u_{ca}^* + 1.0$ $\boldsymbol{V_Z} = 001{+}112 \quad \boldsymbol{V_A} = 011 \quad \boldsymbol{V_B} = 111$
16	$[(u_{ab}^* >= 0.0)\ \&\ (u_{bc}^* >= -1.0)\ \&\ (u_{ca}^* >= 0.0)]$ if $(u_{ab}^* >= u_{ca}^*)$ $d_Z = u_{ab}^* \quad d_A = u_{bc}^* + 1.0 \quad d_B = u_{ca}^*$ $\boldsymbol{V_Z} = 101{+}212 \quad \boldsymbol{V_A} = 111 \quad \boldsymbol{V_B} = 112$ else $d_Z = u_{ca}^* \quad d_A = u_{ab}^* \quad d_B = u_{bc}^* + 1.0$ $\boldsymbol{V_Z} = 001{+}112 \quad \boldsymbol{V_A} = 101 \quad \boldsymbol{V_B} = 111$
17	$[(u_{ab}^* < 1.0)\ \&\ (u_{bc}^* < 0.0)\ \&\ (u_{ca}^* < 0.0)]$ if $(u_{ca}^* >= u_{bc}^*)$ $d_Z = -u_{bc}^* \quad d_A = -u_{ab}^* + 1.0 \quad d_B = -u_{ca}^*$ $\boldsymbol{V_Z} = 101{+}212 \quad \boldsymbol{V_A} = 111 \quad \boldsymbol{V_B} = 211$ else $d_Z = -u_{ca}^* \quad d_A = -u_{bc}^* \quad d_B = -u_{ab}^* + 1.0$ $\boldsymbol{V_Z} = 100{+}211 \quad \boldsymbol{V_A} = 101 \quad \boldsymbol{V_B} = 111$
18	$[(u_{ab}^* >= 1.0)\ \&\ (u_{bc}^* >= -1.0)\ \&\ (u_{ca}^* >= -1.0)]$ if $(u_{ca}^* >= u_{bc}^*)$ $d_Z = u_{ca}^* + 1.0 \quad d_A = u_{ab}^* - 1.0 \quad d_B = u_{bc}^* + 1.0$ $\boldsymbol{V_Z} = 101{+}212 \quad \boldsymbol{V_A} = 201 \quad \boldsymbol{V_B} = 211$ else $d_Z = u_{bc}^* + 1.0 \quad d_A = u_{ca}^* + 1.0 \quad d_B = u_{ab}^* - 1.0$ $\boldsymbol{V_Z} = 100{+}211 \quad \boldsymbol{V_A} = 101 \quad \boldsymbol{V_B} = 201$
19	$[(u_{bc}^* < 0.0)\ \&\ (u_{ca}^* < -1.0)]$ $d_Z = -u_{ab}^* + 2.0 \quad d_A = -u_{ca}^* - 1.0 \quad d_B = -u_{bc}^*$ $\boldsymbol{V_Z} = 100{+}211 \quad \boldsymbol{V_A} = 200 \quad \boldsymbol{V_B} = 201$
20	$[(u_{ab}^* < 0.0)\ \&\ (u_{bc}^* < -1.0)]$ $d_Z = -u_{ca}^* + 2.0 \quad d_A = -u_{bc}^* - 1.0 \quad d_B = -u_{ab}^*$ $\boldsymbol{V_Z} = 001{+}112 \quad \boldsymbol{V_A} = 002 \quad \boldsymbol{V_B} = 012$
21	$[(u_{ab}^* >= 0.0)\ \&\ (u_{ca}^* >= 1.0)]$ $d_Z = u_{bc}^* + 2.0 \quad d_A = u_{ca}^* - 1.0 \quad d_B = u_{ab}^*$ $\boldsymbol{V_Z} = 001{+}112 \quad \boldsymbol{V_A} = 002 \quad \boldsymbol{V_B} = 102$
22	$[(u_{ab}^* < 1.0)\ \&\ (u_{bc}^* < -1.0)\ \&\ (u_{ca}^* < 1.0)]$ if $(u_{ab}^* >= u_{ca}^*)$ $d_Z = -u_{ca}^* + 1.0 \quad d_A = -u_{bc}^* - 1.0 \quad d_B = -u_{ab}^* + 1.0$ $\boldsymbol{V_Z} = 101{+}212 \quad \boldsymbol{V_A} = 102 \quad \boldsymbol{V_B} = 112$ else $d_Z = -u_{ab}^* + 1.0 \quad d_A = -u_{ca}^* + 1.0 \quad d_B = -u_{bc}^* - 1.0$ $\boldsymbol{V_Z} = 001{+}112 \quad \boldsymbol{V_A} = 101 \quad \boldsymbol{V_B} = 102$
23	$[(u_{ab}^* >= 1.0)\ \&\ (u_{ca}^* >= 0.0)]$ $d_Z = u_{bc}^* + 2.0 \quad d_A = u_{ca}^* \quad d_B = u_{ab}^* - 1.0$ $\boldsymbol{V_Z} = 101{+}212 \quad \boldsymbol{V_A} = 102 \quad \boldsymbol{V_B} = 202$
24	$[(u_{bc}^* < -1.0)\ \&\ (u_{ca}^* < 0.0)]$ $d_Z = -u_{ab}^* + 2.0 \quad d_A = -u_{ca}^* \quad d_B = -u_{bc}^* - 1.0$ $\boldsymbol{V_Z} = 101{+}212 \quad \boldsymbol{V_A} = 201 \quad \boldsymbol{V_B} = 202$

Simulations and experimental results

Duty cycles computed using these techniques are exactly the same than those computed using other methods, because the problem of finding duty cycles has only one solution, except maybe on the distribution of d_Z [14]. In order to validate these ideas and check them in a real-time environment, several simulations and physical implementations have been developed. These results will be implemented on a DSP+FPGA controlled grid connected wind generator under development.

A 10 kVA 3-phase VSI connected to a pure resistive load (R = 16 Ω) by means of three 2000 μH inductances (see the structures in Fig. 1) has been simulated using our own graphical environment [15]; the main results are shown in Figs. 5 and 6. In such simulation, the DC-link was V_{dc} = 595 V, the line-to-line 50 Hz output voltage was V_{ac} = 400 V (rms) and the switching frequency was f_S = 6 kHz. Figure 5 arranges the main waveforms for a two-level inverter (above) and a three-level inverter (below). Inverter duty cycles were computed on every switching period using the equations shown in table I and II with d_Z divided at 50%, then PWM signals were generated using a triangular carrier with a resolution of 250 ns neglecting dead-band effects, and all simulated signals were sampled at 200 KSPS. Output voltages (Fig. 5-c and 5-f) are so noisy because of the pure resistive nature of the load.

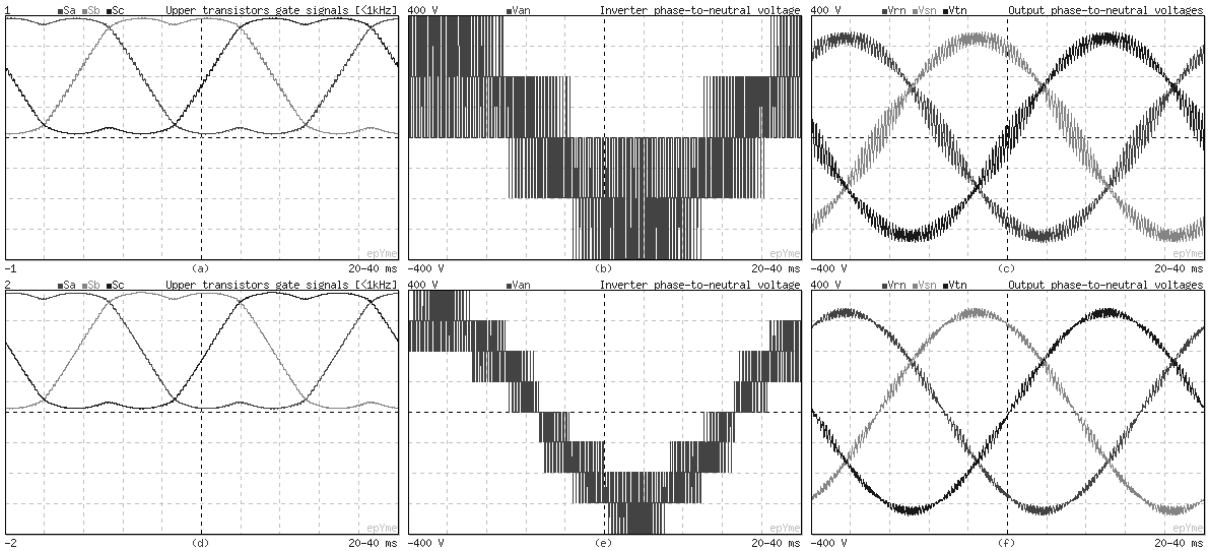

Fig. 5: Simulation of a two-level and a three-level VSI connected to a symmetrical and pure resistive 10 kW load. Images are (a) inverter gate signals through a low-pass filter (second order at 1 kHz) generated for a *two-level* inverter, (b) its phase-to-neutral voltage, (c) its phase-to-neutral voltages on the load, (d) gate signals generated for a *three-level* inverter, also filtered (second order, 1 kHz), (e) its phase-to-neutral voltage, and (f) its phase-to-neutral voltages on the pure resistive load.

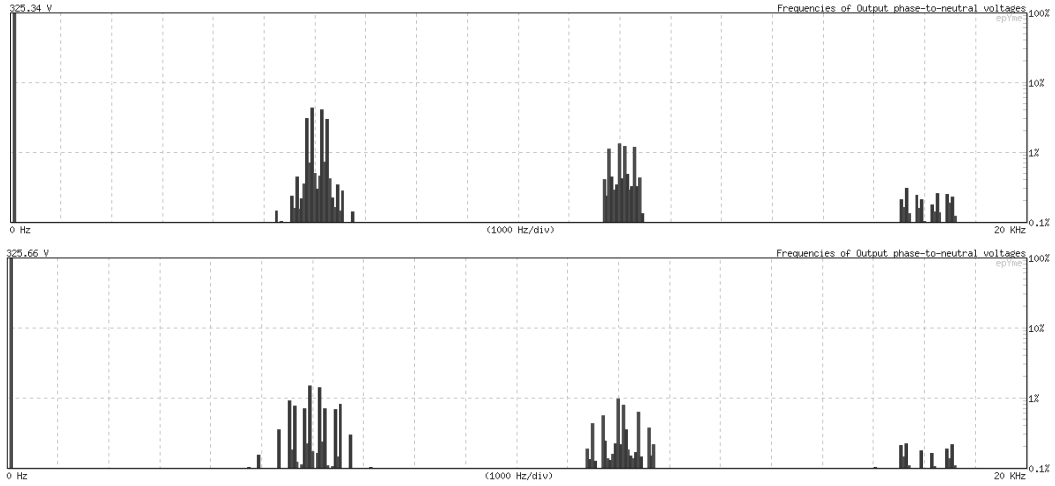

Fig. 6: Frequency spectrum, from DC to 20 kHz, computed for Fig. 5-c (above) and Fig. 5-f (below).

 ISBN: 9789075815009

These simulations validate the proposed equations and show that transitions from triangle to triangle are smooth: as far as all switching periods start at a reduced number of states, unwanted commutations occur only on inner triangles, as shown at Fig. 4; in that case, an additional commutation is produced when changing the main center from the state '100' (used while v_{ab}^* is greater than v_{bc}^*) to '110' (used when v_{bc}^* becomes greater than v_{ab}^*).

Writing the code for this simulations also revealed how this method scales when the number of level increases: the decision tree used to find the nearest three vectors at each sampling period needs 3, 5, 7 and 7 conditions for inverters with two to five levels, always less than $2 \cdot n - 1$ where n is the number of levels. However, the number of triangles is $6 \cdot (n-1)^2$, i.e. 6, 24, 54 and 96, so this method becomes hard for more than five levels.

To validate the proposed method for a real-time application, the algorithm has been implemented on a Texas Instruments TMS320F2812 development board. The source code for a two-level inverter was written in C language and compiled using Code Composer Studio. It generates three 50 Hz sinusoidal references to use them as v_{ab}^*, v_{bc}^* and v_{ca}^*, then it computes all duty cycles based on the equations of Table I, and finally it configures three PWM generators to work at 5 kHz. It can be seen at Fig. 7 that the significant component of this waveform, extracted using a first order low-pass filter, matches the typical SVM phase voltage modulator, seen also at Fig. 5. Figure 8 computes the difference between two filtered gate signals and shows how triple-*n* harmonics are removed.

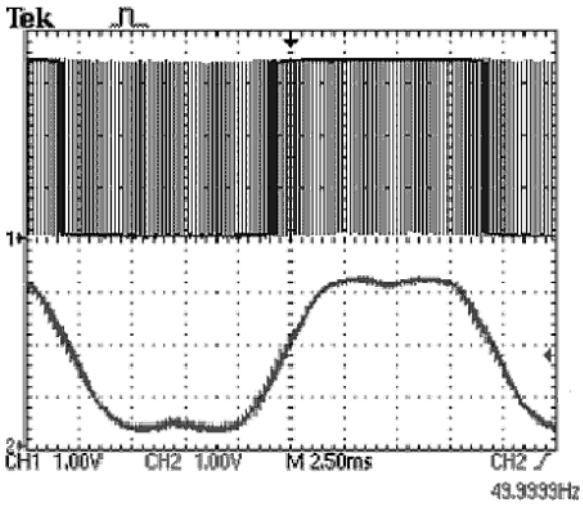

Fig. 7: A 5 kHz gate drive signal generated by a TMS320F2812 processor using the proposed SVM technique, with no filtering (*CH1*, above) and filtered at 1.7 kHz with R = 2.2 kΩ and C = 0.27 μF (*CH2*, below).

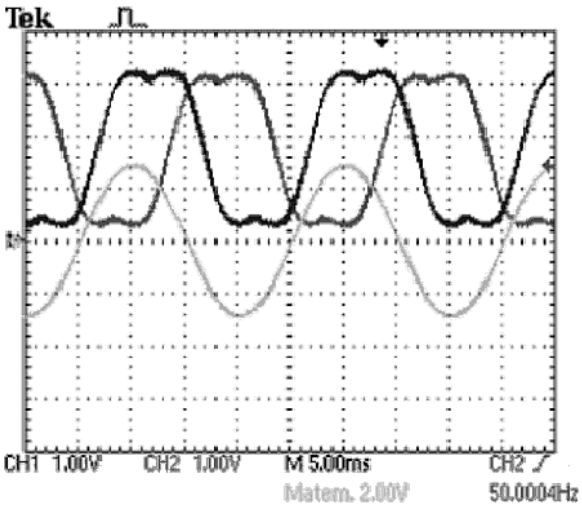

Fig. 8: The difference between two gate signals (*CH1* and *CH2*, above) will produce a line-to-line output voltage free of triple-*n* harmonics (*Matem* = *CH1* – *CH2*, in the middle).

Conclusion

A graphical method for SVM duty cycles computation, valid for two-level and small multi-level inverters have been proposed. The equations used to compute all the information required to generate suitable commutation sequences are easy to understand, thus they are useful for educational purposes. Using them to generate a microprocessor code for a real-time application has been immediate: it did not need either trigonometric functions or artificial transformations; it also avoided the time consuming task of solving lots of inverse problems. The detection of the nearest three vectors is easy and fast using the proposed method, and the complexity of the equations remains constant, but the length of the resulting code increases with the number of levels, so it may become a problem for large multi-level inverters. The final result for small multi-level inverters is a light-weight algorithm with a straightforward implementation even in low-end digital signal processors or small microcontrollers.

References

[1] H.W. Van der Broeck, H.C. Skudelny and G.V. Stanke: "Analysis and realization of a pulsewidth modulator based on voltage space vectors", *IEEE Trans. on Industry Applications*, vol. 24, no. 1, pp. 142-150, January/February 1988.

[2] K. Zhou and D. Wang: "Relationship between space-vector modulation and three-phase carrier-based PWM: a comprehensive analysis", *IEEE Trans. on Industrial Electronics*, vol. 49, no. 1, pp. 186-196, February 2002.

[3] A. Kwasinski, P.T. Krein and P.L. Chapman: "Time domain comparison of pulse-width modulation schemes", *IEEE Power Electronics Letters*, vol. 1, no. 3, pp. 64-68, September 2003.

[4] S. de Pablo, A.B. Rey, L.C. Herrero and J.M. Ruiz: "A simpler and faster method for SVM implementation", *Proceedings of the 12th European Conference on Power Electronics and Application (EPE 2007)*, Aalborg, Denmark, September 2007.

[5] S.R. Bowes and Y.S. Lai: "The Relationship Between Space-Vector Modulation and Regular-Sampled PWM", *IEEE Trans. on Industrial Electronics*, vol. 44, no. 5, pp. 670-679, October 1997.

[6] J. Youm and B. Kwon: "An Effective Software Implementation of the Space-Vector Modulation", *IEEE Trans. on Industrial Electronics*, vol. 46, no. 4, pp. 866-868, August 1999.

[7] P.G. Handley and J.T. Boys: "Space Vector Modulation: an engineering review", *4th Int. Conf. on Power Electronics and Variable-Speed Drives*, pp. 87-91, July 1990.

[8] A.K. Gupta and A.M. Khambadkone: "A Space Vector PWM Scheme for Multilevel Inverters Based on Two-Level Space Vector PWM", *IEEE Trans. on Industrial Electronics*, vol. 53, no. 5, pp. 1631-1639, October 2006.

[9] J. Rodríguez, J.S. Lai and F.Z. Peng: "Multilevel Inverters: A Survey of Topologies, Controls, and Applications", *IEEE Trans. on Industrial Electronics*, vol. 49, no. 4, pp. 724-738, August 2002.

[10] K.A. Corzine: *Operation and Design of Multilevel Inverters*, University of Missouri - Rolla, Tech. Report developed for the Office of Naval Research, on-line at *http://www.motorlab.com* (reviewed June 2009), June 2005.

[11] G. Mondal, F. Sheron, A. Das, K. Sivakumar and K. Gopakumar: "A DC-link Capacitor Voltage Balancing with CMV Elimination using only the Switching State Redundancies for a Reduced Switch Count Multi-Level Inverter fed IM Drive", *EPE Journal*, vol. 19, no. 1, pp. 5-15, March 2009.

[12] K.A. Corzine and X. Kou: "Capacitor Voltage Balancing in Full Binary Combination Schema Flying Capacitor Multilevel Inverters", *IEEE Power Electronic Letters*, vol. 1, no. 1, March 2003.

[13] N. Celanovic and D. Boroyevich: "A Fast Space-Vector Modulation Algorithm for Multilevel Three-Phase Converters", *IEEE Trans. on Industry Applications*, vol. 37, no. 2, March/April 2001.

[14] B.P. McGrath, D.G. Holmes and T. Lipo: "Optimized Space Vector Switching Sequences for Multilevel Inverters", *IEEE Trans. on Power Electronics*, vol. 18, no. 6, pp. 1293-1301, November 2003.

[15] epYme workgroup: "SimulEP, a web simulator for Power Electronics", on-line at <*http://www.epYme.uva.es/SimulEP.php*>, updated on June 2007.