A Space Vector PWM Scheme for Multilevel Inverters Based on Two-Level Space Vector PWM

Amit Kumar Gupta, *Student Member, IEEE*, and Ashwin M. Khambadkone, *Senior Member, IEEE*

***Abstract*—Multilevel inverters are increasingly being used in high-power medium voltage applications due to their superior performance compared to two-level inverters. Among various modulation techniques for a multilevel inverter, the space vector pulsewidth modulation (SVPWM) is widely used. However, the implementation of the SVPWM for a multilevel inverter is complicated. The complexity is due to the difficulty in determining the location of the reference vector, the calculation of on-times, and the determination and selection of switching states. This paper proposes a general SVPWM algorithm for multilevel inverters based on standard two-level SVPWM. Since the proposed multilevel SVPWM method uses two-level modulation to calculate the on-times, the computation of on-times for an n-level inverter becomes easier. The proposed method uses a simple mapping to achieve the SVPWM for a multilevel inverter. A general n-level implementation is explained, and experimental results are given for three-level and five-level inverters.**

***Index Terms*—Multilevel inverter, neutral point clamped (NPC), space vector pulsewidth modulation (SVPWM), switching state, two-level inverter.**

I. Introduction

MULTILEVEL inverters [1] are increasingly being used in high-power medium voltage applications due to their superior performance compared to two-level inverters, such as lower common-mode voltage, lower dv/dt, lower harmonics in output voltage and current, and reduced voltage on the power switches [2].

Among various modulation techniques [1] for a multilevel inverter, space vector pulsewidth modulation (SVPWM) is an attractive candidate due to the following merits. It directly uses the control variable given by the control system and identifies each switching vector as a point in complex (α, β) space. It is suitable for digital signal processor (DSP) implementation. It can optimize switching sequences.

The space vector diagram of any three-phase n-level inverter consists of six sectors. Each sector consists of $(n-1)^2$ triangles. The tip of the reference vector can be located within any triangle. Each vertex of any triangle represents a switching vector. A switching vector represents one or more switching states depending on its location. There are n^3 switching states in the space vector diagram of an n-level inverter. The SVPWM is performed by suitably selecting and executing the switching states of the triangle for the respective on-times. It is also known as "Nearest Three Vector" (NTV) approach. The performance of the inverter significantly depends on the selection of these switching states.

Manuscript received December 13, 2004; revised August 10, 2005. Abstract published on the Internet July 14, 2006.

The authors are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 (e-mail: eleamk@nus.edu.sg).

Digital Object Identifier 10.1109/TIE.2006.881989

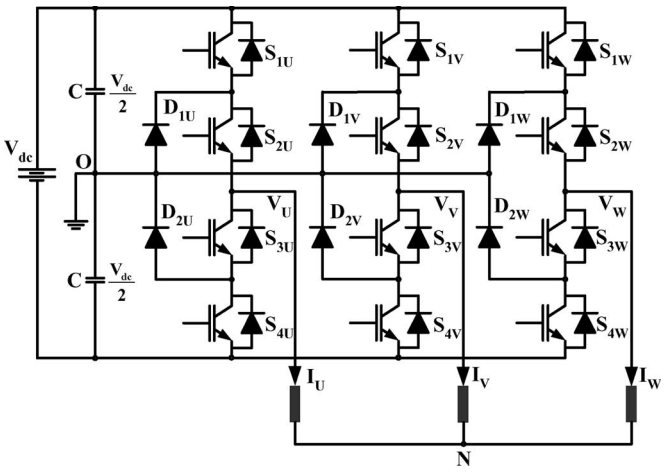

Fig. 1. Three-level NPC inverter topology.

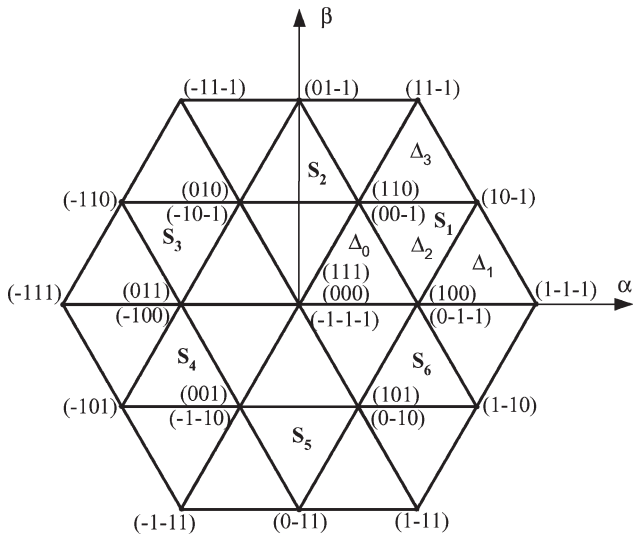

Fig. 2. Space vector diagram of a three-level inverter.

Fig. 1 shows the neutral point clamped (NPC) topology of a three-level inverter by Nabae *et al.* [3]. Fig. 2 shows the space vector diagram of a three-level inverter. There are six sectors (S_1–S_6), four triangles ($\triangle_0$–$\triangle_3$) in a sector, and a total of 27 switching states in this space vector diagram.

As level n increases, the increased number of triangles, switching states, and calculation of on-times adds to the complexity of SVPWM for multilevel inverters.

There are two common approaches to obtain the on-times. The first approach is to determine the triangle, and then solve three simultaneous equations for this triangle to obtain the on-times as in [4]. Whereas, the second approach is to determine the triangle, and then use the particular on-time equations stored in the lookup for this triangle, as in [5]. However, as the number of level increases, both of these approaches become computationally intensive.

The studies in [6] and [7] proposed a general method to obtain on-times for the SVPWM of multilevel inverter in linear modulation range [8]. A Euclidean vector system based SVPWM algorithm is presented by Celanovic and Boroyevich [6], which is quite involved due to the use of several matrix transformations. Furthermore, [6] does not provide a systematic approach for determining the switching states nor does it provide a real-time implementation. Wei *et al.* [7] propose an algorithm which is a different representation of the scheme in [6]. This scheme uses 60° coordinate system to calculate on-times and determine switching states. Since most control schemes provide a voltage reference in $\alpha-\beta$ coordinates, the 60° transformation adds to complexity.

This paper proposes a simple algorithm to perform the SVPWM for a multilevel inverter. The algorithm is based on standard two-level SVPWM, and it can be implemented for any level using one counter. Some researchers [9]–[11] have proposed multilevel SVPWM using two-level concept. However, there are some drawbacks in these methods which are alleviated in the proposed scheme.

Among the schemes based on two-level simplification, Zhang *et al.* [9] introduce a method for on-time calculation where the three-level space vector diagram is divided into six two-level space vector diagrams. The location of the centers of six virtual hexagons is found by a segregation of the three-level space vector diagram. The origin is virtually shifted to one of the six centers, and axes are rotated by 60° to use two-level on-time calculation. This method works well for three levels, as the segregation is required only for three levels. However, can this method be extended to higher levels? The study in [9] does not include the on-time calculation for level $n > 3$.

Seo *et al.* [10] also propose a scheme for a three-level inverter. Similar to Zhang *et al.* [9], the three-level space vector diagram is divided into six two-level space vector diagrams. A two-phase to three-phase conversion is needed to calculate the point to shift of origin of a virtual two-level inverter. Subsequent to the shift of origin and 60° coordinate transformation, on-times are calculated using two-level equations. Even for three levels, this scheme is relatively more computational than the proposed scheme. However, this scheme cannot be directly applied to an n-level inverter. For example, in order to get the on-times for a five-level inverter, the five-level space vector diagram has to be divided into six four-level space vector diagrams, then each four-level space vector diagram has to be divided into six three-level space vector diagrams, and finally, each three-level space vector diagram has to be divided into six two-level space vector diagrams. Therefore, it implies that as level $(n > 3)$ increases, complexity and computation both increase.

Loh and Holmes [11] also use the idea of two-level SVPWM for an n-level inverter. The authors divide the space vector diagram of the n-level into all possible two-level space vector diagrams and propose to use the transformations as in [6] to find the center of a two-level hexagon in the n-level space vector diagram. However, the transformations in [6] are not used just to find the center of a two-level hexagon in the n-level space vector diagram, but they directly calculate the on-times using a set of matrix transformations. Hence, the method proposed by Loh and Holmes [11] using transformations in [6] with two-level on-time calculation will result in total computations higher than in [6]. Thus, the prior art may use two-level simplification, but these methods cannot be extended to multilevel without substantial computational overload.

In addition to the calculation of on-times, the selection of switching states proposed in these methods is restricted to some specific switching sequence(s). As the performance of the multilevel inverter is significantly dependent on the selection of the switching states, the optimization of switching sequence might be required. The optimization cannot be easily achieved by prior art methods.

This paper presents a significantly different approach from all aforementioned references and provides a general solution. It is based on a conventional Cartesian coordinate system, and hence can be easily implemented with existing outer control loops for speed or torque. The following are the salient features of the proposed scheme.

1) The on-time calculation is simple due to the use of two-level SVPWM. The on-time calculation equations do not change with the position of reference vector like the traditional approach in [5], so there is no need for any lookup tables as well.
2) In the space vector diagram of an n-level inverter, the triangle where the reference vector is located is identified as integer $\triangle_j$ using a simple algebraic expression. We call $\triangle_j$ as triangle number, it implies the jth triangle among the $(n-1)^2$ triangles in a sector. Any switching sequence can be executed with respect to triangle $\triangle_j$, leading to a simplicity and flexibility of optimizing the switching sequence.
3) The proposed scheme can be used for any n-level $(n \geq 3)$ inverter without any significant increase in computations.
4) The proposed method can be easily implemented using a commercially available motion-control DSP or microcontroller, which normally supports only two-level modulation.

The scheme is explained for a three-level inverter and then generalized to include any level. Experimental results are provided for three-level and five-level inverters.

II. Proposed Scheme

A. Proposed Method of On-Time Calculation for a Multilevel Inverter

The basic idea of SVPWM is to compensate the required volt-seconds using discrete switching states and their on-times. Traditionally, in order to determine the on-times for a triangle

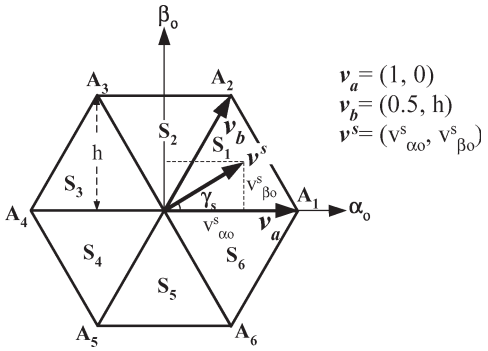

Fig. 3. Space vector diagram for two-level inverter.

of an n-level inverter, three simultaneous equations are solved. However, a classical two-level space vector geometry can be used for on-time calculation for a multilevel SVPWM.

Fig. 3 shows the space vector diagram of a two-level inverter. Every sector is an equilateral triangle of unity side and $h(=\sqrt{3}/2)$ is the height of a sector. On-time calculation for any of the six sectors S_i, $i = 1, 2, \ldots, 6$ is same, so let us consider the operation in sector 1.

On-time calculation is based on the location of the reference vector within a sector. For the sector 1 in Fig. 3, the volt-second balance is given by

$$v^s T_s = v_a t_a + v_b t_b. \tag{1}$$

Time balance is given by

$$T_s = t_a + t_b + t_o. \tag{2}$$

Resolving (1) along the $\alpha_o - \beta_o$ axis, we obtain

$$v^s_{\alpha o} T_s = t_a + 0.5 t_b \tag{3}$$

$$v^s_{\beta o} T_s = h t_b. \tag{4}$$

Solving (2)–(4), we obtain the following equations for the calculation of the on-times:

$$t_a = T_s \left[v^s_{\alpha o} - \frac{v^s_{\beta o}}{2h} \right] \tag{5}$$

$$t_b = T_s \left[\frac{v^s_{\beta o}}{h} \right] \tag{6}$$

$$t_o = T_s - t_a - t_b. \tag{7}$$

Fig. 4 illustrates the proposed method of the on-time calculation for a three-level inverter. Each sector of a three-level inverter can be split into four triangles $\triangle_j$, where $j = 0, 1, 2, 3$. To simplify the on-time calculation, these triangles can be categorized into two types; type 1 and type 2. A triangle of type 1 has its base side at the bottom, as shown in Fig. 4(b). Triangles $\triangle_0$, $\triangle_1$, and $\triangle_3$ are of type 1. A triangle of type 2 has its base side at the top, as shown in Fig. 4(d). Triangle $\triangle_2$ is of type 2.

Let us assume that the side of a triangle is 1(unity) and $h(=\sqrt{3}/2)$ is the height of the triangle. In Fig. 4(a), $\boldsymbol{v}^*$ is

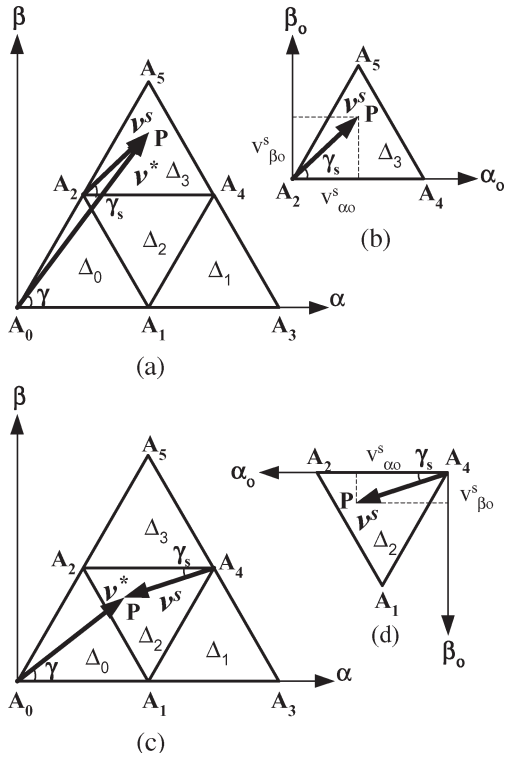

Fig. 4. Space vector diagram—virtual two-level from three-level.

the reference vector of magnitude $|\boldsymbol{v}^*|$ at an angle of γ with the α-axis. We define a small vector $\boldsymbol{v}^s$, which describes the same point in shifted system (α_o, β_o) [see Fig. 4(b) and (d)]. It makes γ_s angle with the α_o axis. The volt-seconds required to approximate the small vector $\boldsymbol{v}^s$ in the shifted system (α_o, β_o) should be equal to those required for the actual vector $\boldsymbol{v}^*$ in the original system (α, β). Hence, we can obtain the on-times for any reference vector by finding the on-times of the respective small vector $\boldsymbol{v}^s$.

To achieve the volt-seconds for any reference vector in a sector of a three-level inverter, we have to identify the triangle in which the required reference is located and then find ($v^s_{\alpha o}$, $v^s_{\beta o}$). The on-time calculations can be performed using the geometry shown in Fig. 4(b) or (d), which would result in the same on-time equations as those for a classical two-level SVPWM (5)–(7).

A triangle of type 1 is similar to a sector 1 of a virtual two-level inverter. For example; In Fig. 4(a), triangle $\triangle_3$ can be assumed similar to sector 1 of a two-level inverter if A_2 is taken as zero vector of the virtual two-level sector as shown in Fig. 4(b). Vector A_2P defines the small vector $\boldsymbol{v}^s(v^s_{\alpha o}, v^s_{\beta o})$. On-times t_a (t_{A_4}), t_b (t_{A_5}), and t_o (t_{A_2}) are calculated by using (5)–(7), where the multiplication operations are required only for (5) and (6).

A triangle of type 2 is similar to a sector 4 of a virtual two-level inverter. For example; In Fig. 4(c), triangle $\triangle_2$ can be considered similar to sector 4 of a two-level inverter if A_4 is assumed to be zero vector [see Fig. 4(d)]. In this example, A_4P represents small vector $\boldsymbol{v}^s(v^s_{\alpha o}, v^s_{\beta o})$. On-times t_a (t_{A_2}), t_b (t_{A_1}), and t_o (t_{A_4}) are calculated by using (5)–(7).

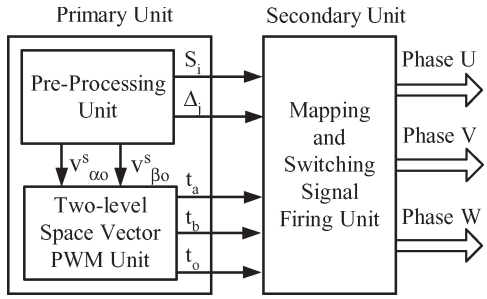

Fig. 5. Block diagram of the proposed scheme.

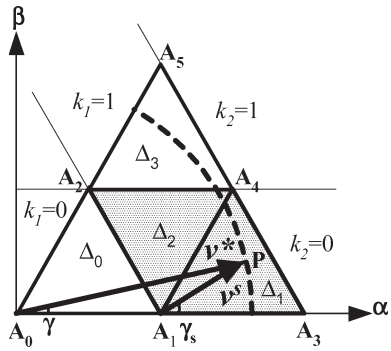

Fig. 6. Space vector diagram—sector 1 of a three-level inverter.

Since the triangles within any sector of an n-level inverter are analogous to a sector of two-level inverter, the idea can be extended to any level. Thus, multilevel on-time calculation problem is converted to a two-level on-time calculation problem.

The on-times t_a, t_b, and t_o are a function of ($v^s_{\alpha o}$, $v^s_{\beta o}$) for any triangle, using (5)–(7). Therefore, the on-time calculation for one triangle can also be used for any other triangle.

B. Block-Diagram Explanation of the Scheme

Block diagram in Fig. 5 gives an overview of the proposed method. It consists of two basic units, namely primary unit (PU) and secondary unit (SU), respectively. The PU consists of a preprocessing unit and two-level SVPWM unit.

The PU is basically a DSP or microcontroller. The preprocessing unit does two main tasks: 1) determination of small vector $\boldsymbol{v}^s$ coordinates ($v^s_{\alpha o}$, $v^s_{\beta o}$) and 2) determination of the sector S_i and the triangle $\triangle_j$ of the small vector $\boldsymbol{v}^s$. Two-level SVPWM unit obtains the on-times t_o, t_a, and t_b by using (5)–(7).

The SU is basically a mapping unit and uses memory. It fires the prestored switching sequence for the three-phase inverter based on sector S_i, triangle $\triangle_j$ for the on-times obtained from the PU. For a multilevel inverter, a vertex of any triangle can have multiple redundancies (two or more possible switching states). For a triangle, a switching sequence is formed using a combination of the most suitable switching states from all possible switching states at the vertices. The resulting switching sequence is mapped with respect to the triangle and sector number. The switching sequence is then fired for the on-times obtained from the PU. Since in the proposed method, triangle is considered as the basic unit, and the mapping takes care of the redundancies, any suitable vertex can be chosen as zero vector. While doing so, redundancies at other vertices are also made use of. The sequence, in which the on-times t_a, t_b, and t_o have to be used, will be dependent on the order of selecting the switching states. Thus, the proposed algorithm is able to make use of any redundancies for any vertex of the triangle. As opposed to this, if the two-level hexagon is used to mimic the two-level modulation, only two redundancies of zero vector are considered. Hence, for higher level where middle vectors have higher redundancies, such approach will not be able to make use of all redundancies.

First let us understand the implementation of this scheme on a three-level inverter, then we generalize it to an n-level inverter. We will thus demonstrate that unlike prior art, the proposed scheme is easily extendable to any n-level inverter.

III. Implementation of Proposed Scheme for Three-Level Inverter

A. Primary Unit for Three-Level Inverter

For any given reference vector, we determine the sector of operation S_i and its angle γ within the sector by using (8) and (9), respectively,

$$S_i = \text{int}\left(\frac{\theta}{60}\right) + 1 \tag{8}$$

$$\gamma = \text{rem}\left(\frac{\theta}{60}\right) \tag{9}$$

where θ is the angle of the reference vector with respect to α-axis, and int and rem represent standard function integer and remainder, respectively.

1) Identification of Triangle and Determination of Small Vector $\boldsymbol{v}^s$: The tip P of the reference vector $\boldsymbol{v}^*$ can be located in any of the four triangles; $\triangle_0$, $\triangle_1$, $\triangle_2$, or $\triangle_3$. As mentioned in Section II, a triangle in Fig. 6(a) can be treated as a sector of a two-level inverter. Therefore, the objective here is to identify the triangle in which the point P is located.

The search of the triangle of the small vector (or point P) can be narrowed down by using two integers k_1 and k_2. They are defined by the coordinates (v_α, v_β) of point P as

$$k_1 = \text{int}(v_\alpha + v_\beta/\sqrt{3}) \quad k_2 = \text{int}(v_\beta/h). \tag{10}$$

k_1 represents the part of the sector between the two lines joining the vertices, separated by distance h and inclined at 120° with respect to α-axis (see Fig. 6). $k_1 = 0$ signifies that the point P is below line A_1A_2. $k_1 = 1$ signifies that point P is between line A_1A_2 and line A_3A_5. k_2 represents the part of the sector between the two lines joining the vertices, separated by distance h and parallel to α-axis. $k_2 = 0$ signifies that the point P is between line A_0A_3 and line A_2A_4. $k_2 = 1$ signifies that

the point P is above line A_2A_4. Geometrically, the values of k_1 and k_2 are an intersection of two rectangular regions which is either a triangle or rhombus. In other words, the point P lies in (a) triangle $\triangle_0$ if $k_1 = 0$ and $k_2 = 0$, (b) rhombus $A_1A_3A_4A_2$ (shaded) if $k_1 = 1$ and $k_2 = 0$, and (c) triangle $\triangle_3$ if $k_1 = 1$ and $k_2 = 1$. The same analogy can be used for any level.

In Fig. 6, the reference vector is located in rhombus $A_1A_3A_4A_2$. This rhombus is made up of two triangles $\triangle_1$ and $\triangle_2$. The point P can be located in any of the two. Let $(v_{\alpha i}, v_{\beta i})$ be the coordinates of the point P with respect to the point A_1 obtained as

$$v_{\alpha i} = v_\alpha - k_1 + 0.5k_2 \quad v_{\beta i} = v_\beta - k_2 h. \tag{11}$$

The slope of $\overrightarrow{A_1P}$ is $v_{\beta i}/v_{\alpha i}$, and the slope of diagonal A_1A_4 is $\sqrt{3}$. The triangle where point P is located can be determined by comparing the slope of $\overrightarrow{A_1P}$ with the slope of A_1A_4. Slope comparison is done by evaluating the inequality $v_{\beta i} \leq \sqrt{3}v_{\alpha i}$. If $v_{\beta i} \leq \sqrt{3}v_{\alpha i}$ is true, then the point P is within the triangle $\triangle_1$, otherwise it is within the triangle $\triangle_2$.

If point P is within triangle $\triangle_1$, the small vector $\boldsymbol{v}^s$ ($v^s_{\alpha o}$, $v^s_{\beta o}$) is represented by $\overrightarrow{A_1P}$ $(v_{\alpha i}, v_{\beta i})$. Otherwise, point P is within triangle $\triangle_2$, and small vector $\boldsymbol{v}^s$ $(v^s_{\alpha o}, v^s_{\beta o})$ is represented by $\overrightarrow{A_4P}$ $(0.5 - v_{\alpha i}, h - v_{\beta i})$. These two results can be generalized to triangles of type 1 and type 2, respectively.

When the point P is in triangle $\triangle_3$, the inequality $v_{\beta i} \leq \sqrt{3}v_{\alpha i}$ will be true because triangle $\triangle_3$ is a triangle of type 1. The small vector $\boldsymbol{v}^s$ $(v^s_{\alpha o}, v^s_{\beta o})$ is represented by $\overrightarrow{A_2P}$ $(v_{\alpha i}, v_{\beta i})$. Thus, we determine the small vector $\boldsymbol{v}^s$ $(v^s_{\alpha o}, v^s_{\beta o})$ for any given reference vector.

Triangle number $\triangle_j$ is calculated along with on-times using the same idea, using k_1 and k_2.

For a type 1 triangle, the triangle number $\triangle_j$ is obtained as

$$\Delta_j = k_1^2 + 2k_2. \tag{12}$$

For a type 2 triangle, the triangle number $\triangle_j$ is obtained as

$$\Delta_j = k_1^2 + 2k_2 + 1. \tag{13}$$

To conclude, the triangle in a sector is defined by an integer $\triangle_j$. It is obtained by a simple algebraic expression (12) or (13). The triangle number $\triangle_j$ is formulated to provide a simple way of arranging the triangles, leading to ease of identification and extension to any level.

2) Determination of On-times by Two-Level SVPWM Unit: Two-level SVPWM unit gets ($v^s_{\alpha o}$, $v^s_{\beta o}$) as input at the beginning of every switching period and calculates corresponding on-times using (5)–(7). Since one of the vertices is treated as zero vector, multiplication is required only for (5) and (6).

B. Secondary Unit for Three-Level Inverter

The job of SU is to generate gating signals using the sector S_i, triangle $\triangle_j$, and on-times t_a, t_b, and t_o. These parameters are obtained from the PU for every switching period. SU is a mapping unit and uses memory. A switching state is defined as $[s_u, s_v, s_w]$. s_u, s_v, and s_w can take a value -1, 0, or 1. As shown in Fig. 2, there are 27 switching states for a three-level inverter, inclusive of 19 distinct and eight redundant states. These states are stored in a table. Normally, four of them are used by the switching sequence in every switching period, depending on the position of the reference vector.

TABLE I
SWITCHING SEQUENCE FOR $\triangle_3$ OF SECTOR 1 OF THREE-LEVEL INVERTER

Vertex	A_2	A_4	A_5	A_2	A_2	A_5	A_4	A_2
s_u	0	1	1	1	1	1	1	0
s_v	0	0	1	1	1	1	0	0
s_w	-1	-1	-1	0	0	-1	-1	-1
On-time	t_{o-}	t_a	t_b	t_{o+}	t_{o+}	t_b	t_a	t_{o-}

A switching sequence for a triangle is made up of the switching states corresponding to the vertices of the triangle. For example, Table I shows the switching sequence for the triangle $\triangle_3$ of the sector 1 of a three-level inverter using minimum commutation criteria. Here, t_{o+} and t_{o-} play an important role in dc link balancing [10] and $t_{o+} + t_{o-} = t_o$. This sequence is applied for the time duration of $2T_s$ when the reference vector $\boldsymbol{v}^*$ is in triangle $\triangle_3$. Similarly, any other switching sequence for any other triangle can be applied due to the generality of the unit.

A switching sequence is also dependent on the switching scheme used, e.g., common-mode voltage elimination [9] and flux modulation [11]. Hence, there could be many switching sequences for any triangle. However, only one such sequence can be implemented at a time. The switching sequence can be treated as a function of the sector and triangle of the reference vector. For a three-level inverter, the memory required for the storage of switching states is approximately 27B.

C. Experimental Results for Three-Level Inverter

To implement, the algorithm has been developed on a dSPACE DS1104. DS1104 was used due to its availability, but no floating-point operations are used. Therefore, the algorithm can be implemented on a fixed-point DSP as well. A laboratory prototype three-level NPC inverter and a star-connected RL load are used.

Fig. 7(a) shows line voltage and current waveforms. The dc link voltage is 170 V, modulation index is $m_i = 0.8$, switching frequency is 5 kHz, fundamental frequency is 50 Hz, load inductance per phase is 0.47 H, and load resistance per phase is 48.4 Ω. Neutral-point balancing is obtained using [10].

Fig. 7(b) shows the fast Fourier transform (FFT) of line voltage V_{VW} for $0 < f_s \leq 45$ kHz inclusive of a magnified FFT view in the inset for $0 < f_s \leq 1$ kHz, to better show the low-order harmonics. The x-scale of inset is frequency in kilohertz. The fundamental component of line–line voltage increases with modulation index. The weighted harmonic distortion V_{WTHD} [12] obtained at this modulation index is 0.133%.

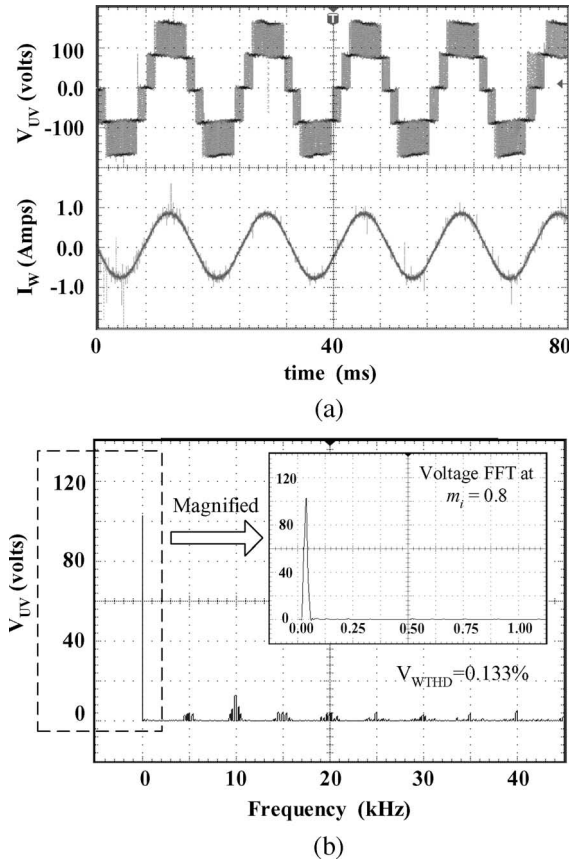

Fig. 7. Experimental results for three-level NPC inverter. (a) Line voltage and current waveforms. (b) FFT of the line voltage.

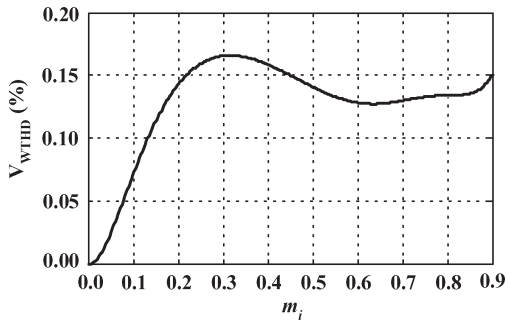

Fig. 8. Modulation index m_i versus percentage weighted total harmonic distortion V_{WTHD}.

Fig. 8 shows the variation of percentage V_{WTHD} with modulation index m_i. As we discuss, the SVPWM in sinusoidal or linear range, which is $0 \leq m_i < 0.907$ [8].

IV. Extension of the Proposed Scheme for an n-Level Inverter

The proposed scheme for a three-level inverter illustrated the simplicity of our approach. The scheme proposed can be easily applied to an n-level inverter as well. Similar to three levels, we require sector number, triangle number, on-times, and relevant switching states to implement the scheme for an n-level inverter. To understand this implementation for an n-level inverter, let us consider the same two units as explained before, i.e., PU and SU.

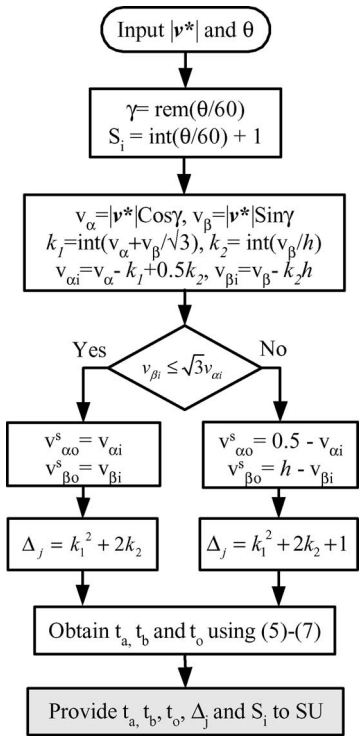

Fig. 9. Flowchart for the proposed scheme.

A. Primary Unit for n-Level Inverter

For a given $\boldsymbol{v}^*$, the PU calculates sector number, triangle number, and on-times. They are provided to SU in every switching period. The flowchart in Fig. 9 describes the job of a PU. In this flowchart for a given $\boldsymbol{v}^*$, all the arithmetic and logical operations required by main routine are independent of level n. Therefore, the number of computations remains the same for any n-level SVPWM.

In Table II, we show the steps required to obtain sector number S_i, triangle number $\triangle_j$, and on-times at a given modulation index $m_i = 0.87$ and angle $\theta = 78^\circ$ for a three-level, five-level, and seven-level inverter at switching frequency $f_s = 5$ kHz.

We see from Table II that the same number of steps are required for all the three cases. For a given $|\boldsymbol{v}^*|$ and θ, the algorithm requires nearly ten multiplications, ten additions/subtractions, and one branch instructions to calculate sector number, triangle number, and on-times. They remain the same for any level. The following points are concluded for the PU of an n-level inverter.

1) In the proposed scheme, the computations do not change with the level. The PU remains the same for any level.
2) On-time calculation equations do not change with triangle like the traditional approach as in [5], it saves storage space.
3) The computations required by the PU are significantly lesser than the methods similar to those proposed in [5] and comparatively lesser than the methods similar to those proposed in [6].
4) Triangle number $\triangle_j$ is easily identified as an integer using (12) or (13) for any level. Triangle number $\triangle_j$

TABLE II
STEPS REQUIRED FOR SVPWM OF THREE-LEVEL, FIVE-LEVEL, AND SEVEN-LEVEL INVERTERS

Steps	3-level	5-level	7-level
$\lvert v^* \rvert$	1.91	3.82	5.73
S_i	2	2	2
γ	18^o	18^o	18^o
(v_α, v_β)	(1.5788, 0.5130)	(3.1575, 1.0259)	(4.7363, 1.5389)
k_1, k_2	1, 0	3, 1	5, 1
$(v_{\alpha i}, v_{\beta i})$	(0.5788, 0.5130)	(0.6575, 0.1599)	(0.2363, 0.6729)
$v_{\beta i} \le \sqrt{3} v_{\alpha i}$?	Yes	Yes	No
$(v^s_{\alpha o}, v^s_{\beta o})$	(0.5788, 0.5130)	(0.6575, 0.1599)	(0.2637, 0.1931)
$\triangle_j$	1	11	28
t_a	28.26μs	56.52μs	15.22μs
t_b	59.24μs	18.47μs	22.3μs
t_o	12.50μs	25.01μs	62.48μs

leads to a flexibility and simplicity in switching-sequence mapping process.

5) The scheme is based on a two-level SVPWM. Therefore, an existing two-level SVPWM module can be easily adapted to the multilevel inverters. It can be used with an existing torque or speed control scheme implemented with two-level geometry. This would save reengineering cost. Commercially available motion-control DSP normally supports only two-level modulation. However, the proposed architecture in Fig. 5 can be easily implemented on it.

B. Secondary Unit for n-Level Inverter

As level increases, the number of switching states also increases. The SU can be designed to map the switching sequence for any triangle of an n-level inverter. As mentioned before, a switching sequence for a triangle is made up of the switching states corresponding to the vertices of the triangle.

For an n-level inverter, a switching state is defined as $[s_u, s_v, s_w]$, where s_u, s_v, or s_w can take a value from $-(n-1)/2$ to $(n-1)/2$. There are n^3 switching states for an n-level inverter. A vertex in the space vector diagram of an n-level inverter can have 1 to n switching states. For example; in Fig. 2, there are 27 switching states in the space vector diagram of a three-level inverter, and a vertex has 1 to 3 switching states. We provide a simple tabulation using which, the switching states associated with any vertex in the space vector diagram of an n-level inverter can be obtained. This is an off-line process.

The switching states associated with a vertex can be determined by using its (α, β) coordinates. Let us represent the (α, β) coordinates of a vertex as (v_α, v_β) for the determination of its switching states. To this end, we define five integer variables as follows:

$$m_1 = \left[v_\alpha + \frac{v_\beta}{\sqrt{3}} \right] \quad m_2 = [n - m_1] \quad m_3 = \left[\frac{n-1}{2} \right]$$
$$m_4 = [m_1 - m_3] \quad m_5 = \left[\frac{v_\beta}{h} - m_3 \right] \tag{14}$$

where m_1 is a vertex index whose value depends on the location of the vertex in $\alpha-\beta$ plane. When subtracted from n, the total number of switching states m_2 at this vertex is obtained. $-m_3$ provides the state of phase w, m_4 provides the state of phase u and m_5 provides the state of phase v.

TABLE III
SWITCHING STATES FOR A VERTEX OF AN n-LEVEL INVERTER IN SECTOR I

Switching State	s_u	s_v	s_w
1	m_4	m_5	- m_3
2	m_4 +1	m_5 +1	- m_3 +1
....	.....	.....	.....
....	.....	.....	.....
m_2	$m_4 + m_2$ -1	$m_5 + m_2$ -1	- $m_3 + m_2$ -1

Table III determines all the switching state(s) $[s_u, s_v, s_w]$ at a vertex (v_α, v_β) in sector 1 of an n-level inverter. For a three-phase inverter, it is $m_2 \times 3$ table, where $1 \le m_2 \le n$.

For example, for a three-level inverter corresponding to vertex (0.5, h), the values of m_1, m_2, m_3, m_4, and m_5 are 1, 2, 1, 0, and 0, respectively. Therefore, the switching states are [0, 0, -1] and [1, 1, 0]. For a five-level inverter corresponding to vertex (0.5, h), the values of m_1, m_2, m_3, m_4, and m_5 are 1, 4, 2, -1, and -1, respectively. Therefore, the switching states are [-1, -1, -2], [0, 0, -1], [1, 1, 0], and [2, 2, 1].

For other sectors, the switching states are determined by mapping between the Tables III and IV. Table IV is of 6×3 dimension for any n-level inverter. Columns represent the three phases and rows represent the six sectors. For example, if in sector 1 a switching state at some particular vertex is [1, 1, 0], then in sectors 2, 3, 4, 5, and 6, the switching state corresponding to the similar vertex will be [-1, 0, -1], [0, 1, 1], [-1, -1, 0], [1, 0, 1], and [0, -1, -1], respectively.

The memory required to store the switching states for an n-level inverter is $3n^3(n-1)/8$B. For example, 188B for five-level, 772B for seven-level, and 2187B for nine-level.

C. Experimental Results for Five-Level Cascaded H-Bridge Inverter

The proposed scheme can be used for both NPC and cascaded H-bridge topologies of multilevel inverter, as for a given level, both have the same space vector diagram. For both of them, the computations required by the PU are the same. However, for a given level, the actual gating signals are different for the two topologies. They are generated by a simple multiplexing operation without the need for additional hardware.

TABLE IV
SWITCHING STATES MAPPING BETWEEN SECTOR 1 AND OTHER SECTORS

Sector	Phase U	Phase V	Phase W
S_1	s_u	s_v	s_w
S_2	$-s_v$	$-s_w$	$-s_u$
S_3	s_w	s_u	s_v
S_4	$-s_u$	$-s_v$	$-s_w$
S_5	s_v	s_w	s_u
S_6	$-s_w$	$-s_u$	$-s_v$

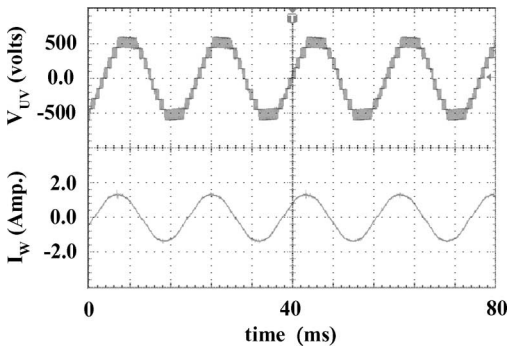

Fig. 10. Experimental line voltage and current waveforms for five-level cascaded H-bridge inverter.

We show the implementation of the proposed scheme for a five-level cascaded H-Bridge inverter including overmodulation range in [13]. Fig. 10 shows experimental waveforms $V_{\rm UV}$ and $I_{\rm W}$ for a five-level cascaded H-Bridge inverter with the star-connected RL load. The switching states are selected based on minimum commutation criteria. Experimental conditions are $m_i = 0.8$, switching frequency is 5 kHz, fundamental frequency is 50 Hz, load inductance per phase is 1.4 H, and load resistance per phase is 120 Ω.

Some multilevel carrier PWM can be implemented using a single counter. Zhang *et al.* [14] use a single counter to generate a time base for the modulation. However, it requires elaborate peripheral circuit. The complexity of the peripheral circuit and its engineering will increase with the number of levels. On the other hand, Cecati *et al.* [15] use a FPGA based scheme. This paper describes the method for only a single-phase system, hence, the number of counters will increase for three phases. Loh *et al.* [16] propose a very cumbersome scheme. This paper developed a carrier-based PWM scheme for a five-level cascaded H-bridge inverter. This scheme uses one MiniDSP controller for every H-bridge module apart from one master DSP controller and a PC controller. Though a single counter is used for an H-bridge, a combination of several DSPs, their synchronization and interrupt management makes this scheme very complex, especially for a medium voltage drives. On the other hand, the proposed scheme can make use of existing two-level SVPWM platform with minor modifications.

V. CONCLUSION

A simple SVPWM algorithm for a multilevel inverter based on a standard two-level inverter has been proposed. The computations do not increase with level. The proposed method can be easily implemented using a commercially available motion-control DSP or microcontroller, which normally supports only two-level modulation. The main advantage of the proposed scheme over earlier schemes is that, it can be used with an existing torque or speed control scheme implemented with two-level geometry. Since such schemes provide voltage reference in $\alpha-\beta$ coordinates, the proposed scheme uses most of the two-level calculation and adapts to any n-level inverter. The scheme has been implemented for three-level NPC, five-level cascaded, and seven-level cascaded inverter using the same PU, and it can be extended to any level. The scheme can be used for both NPC and cascaded H-bridge inverter topologies. The scheme can be easily extended to include overmodulation range.

REFERENCES

[1] J. Rodriguez, J. S. Lai, and F. Z. Peng, "Multilevel inverters: A survey of topologies, controls, and applications," *IEEE Trans. Ind. Electron.*, vol. 49, no. 4, pp. 724–738, Aug. 2002.

[2] R. Teodorescu, F. Beaabjerg, J. K. Pedersen, E. Cengelci, S. U. Sulistijo, B. O. Woo, and P. Enjeti, "Multilevel converters—A survey," in *Proc. EPE Conf.*, 1999, pp. 2–11.

[3] A. Nabae, I. Takahashi, and H. Akagi, "A new neutral-point clamped pwm inverter," *IEEE Trans. Ind. Appl.*, vol. IA-17, no. 5, pp. 518–523, Sep./Oct. 1981.

[4] T. Ishida, T. Miyamoto, T. Oota, K. Matsuse, K. Sasagawa, and L. Huang, "A control strategy for a five-level double converter with adjustable dc link voltage," in *Proc. Ind. Appl. Conf.*, Oct. 2002, vol. 1, pp. 530–536.

[5] S. K. Mondal, J. O. P. Pinto, and B. K. Bose, "A neural-network-based space-vector pwm controller for a three-level voltage-fed inverter induction motor drive," *IEEE Trans. Power Electron.*, vol. 38, no. 3, pp. 660–669, May/Jun. 2002.

[6] N. Celanovic and D. Boroyevich, "A fast space vector modulation algorithm for multilevel three phase converters," *IEEE Trans. Ind. Appl.*, vol. 37, no. 2, pp. 637–641, Mar./Apr. 2001.

[7] S. Wei, B. Wu, F. Li, and C. Liu, "A general space vector pwm control algorithm for multilevel inverters," in *Proc. 18th Annu. IEEE APEC*, Feb. 2003, vol. 1, pp. 562–568.

[8] J. Holtz, W. Lotzkat, and A. M. Khambadkone, "On continuous control of pwm inverters in overmodulation range including six-step," *IEEE Trans. Power Electron.*, vol. 8, no. 4, pp. 546–553, Oct. 1993.

[9] H. Zhang, A. Von Jouanne, S. Dai, A. K. Wallace, and F. Wang, "Multilevel inverter modulation schemes to eliminate common-mode voltages," *IEEE Trans. Ind. Appl.*, vol. 36, no. 6, pp. 1645–1653, Nov./Dec. 2000.

[10] J. H. Seo, C. H. Choi, and D. S. Hyun, "A new simplified space-vector pwm method for three-level inverters," *IEEE Trans. Power Electron.*, vol. 16, no. 4, pp. 545–550, Jul. 2001.

[11] P. C. Loh and D. G. Holmes, "Flux modulation for multilevel inverters," *IEEE Trans. Ind. Appl.*, vol. 38, no. 5, pp. 1389–1399, Sep./Oct. 2002.

[12] T. Bruckner and D. G. Holmes, "Optimal pulse-width modulation for three-level inverters," *IEEE Trans. Power Electron.*, vol. 1, no. 20, pp. 82–89, Jan. 2005.

[13] A. K. Gupta and A. M. Khambadkone, "A general space vector pwm algorithm for a multilevel inverter including operation in overmodulation range," in *Proc. IEEE IEMDC*, May 2005, pp. 1437–1444.

[14] Z. Zhang, J. Kuang, X. Wang, and B. T. Ooi, "Force commutated hvdc and svc based on phase-shifted multi-converter modules," *IEEE Trans. Power Del.*, vol. 8, no. 2, pp. 712–718, Apr. 1993.

[15] C. Cecati, A. Dell'Aquila, A. Lecci, M. Liserre, and V. G. Monopoli, "A discontinuous carrier-based multilevel modulation for multilevel converters," in *Proc. 30th Annu. IEEE IECON*, Nov. 2004, vol. 1, pp. 280–285.

[16] P. C. Loh, D. G. Holmes, and T. A. Lipo, "Implementation and control of distributed pwm cascaded multilevel inverters with minimal harmonic distortion and common-mode voltage," *IEEE Trans. Power Electron.*, vol. 20, no. 1, pp. 90–99, Jan. 2005.

Amit Kumar Gupta (S'04) was born in Lucknow, India, in 1978. He received the B.Eng. degree in electrical engineering from Indian Institute of Technology, Roorkee, India, in 2000. He is currently working toward the Ph.D. degree at the National University of Singapore, Singapore.

From 2000 to 2003, he was with Bechtel India Pvt. Ltd., New Delhi, India, and Samsung Heavy Industries Ltd., Korea. His research interests include power electronics and motion control.

Ashwin M. Khambadkone (M'95–SM'04) received the Dr. Ing. degree from Wuppertal University, Germany, in 1995.

At Wuppertal, he was involved in research and industrial projects in the areas of PWM methods, field-oriented control, parameter identification, and sensorless vector control. From 1995 to 1997, he was a Lecturer with the University of Queensland. In 1998, he was also with the Indian Institute of Science, Bangalore, India. Since 1998, he has been an Assistant Professor with the National University of Singapore, Singapore. His research activities are in the control of ac drives, design and control of power electronic converters, and fuel-cell-based systems.

Dr. Khambadkone was the recipient of the Outstanding Paper Award for the year 1991 and the Best Paper Award for the year 2002 from the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS.