
by

Roland Blümel, Uni der Bundeswehr München, Werner-Heisenberg-Weg 39, 8014 Neubiberg, FRG
Reinhold Blümel†, Department of Physics, University of Delaware, Newark, DE 19716, USA

Indexing terms: PWM techniques, CAD, Bellman’s Dynamic Programming

Abstract: Today’s snubberless VSI-topologies with hard-driven IGBTs are well known for the simplicity they bring to adjustable speed drives. PWM techniques are also very popular. But a rather simple evaluation of switching losses shows, that the lowest number of switching operations does not always indicate minimal switching losses as well. Due to the nonlinearity of switching losses, one switching cycle (turn-on & turn-off) having a current level of “i” involved dissipates far more energy than two cycles involving “i/2”. In this paper, the two conflicting demands -low inverter losses and low current distortion- are settled by a CAD program built on Bellman’s Dynamic Programming.

INTRODUCTION

PWM preliminaries - The performance of PWM VSI-fed inverter drives depends crucially on the way the control makes use of the 7 (8) inverter topologies. Since switching the inverter is the only input of the plant, every control operation is derived from an answer to one of the following key questions.

Q1: What inverter state is next? (→switching sequence)
Q2: What is the right time to switch? (→firing angles)

In general these two questions can be answered on-line, off-line, machine-side or inverter-side, and the Q1 and Q2 answering algorithm now serves as a distinguishing feature to classify different PWM techniques as follows:

Q1: off-line, inverter-side; Q2: on-line, inverter-side

The first category often found in continuous-time regulators requires the use of a so called modulator. This modulator decomposes the analog reference voltage at its inputs into a pre-defined switching sequence. Next, the modulator combines the switching sequence with on-line calculated firing angles. Now the pulse pattern is ready to be asserted. Several examples of these MF techniques e. g. natural sampling (1), symmetrical PWM (2), third harmonic injection (3), discontinuous mode PWM (4) are depicted in Fig. 1 above. A detailed discussion may be found in [1,2]. All these techniques have in common, that the switching sequence is fixed and never changed. The modulator then passes the system through its pre-defined switching sequence, while controlling the dwell time of each inverter state on-line (Q2). The entire control process is done in the Q2-domain. This principle also holds for vector modulation. Fig. 3, for instance, illustrates how a reference voltage vector \( \hat{u}_{\text{ref}} \) is synthesized using adjacent vectors i. e. \( U_0, U_2 \) and \( \hat{u} \) are asserted using a pre-defined switching sequence.

† Heisenberg fellow of the Deutsche Forschungsgemeinschaft.
Control concepts of the second category are oriented toward more advanced DSP-based schemes. A suitable inverter state is asserted, and the load controls the dwell time usually upon a hysteresis violation. So Q2 is answered by the induction machine itself and it is up to the microprocessor to evaluate a suitable sequence of inverter states that guarantee stability and low switching losses, in this order. As Fig. 2 depicts, the transition to another inverter state must instantaneously go into effect every time the current trajectory is about to exceed a dead-band (space vector notation: hexagon). These instants are determined load-side and are hardly foreseeable due to nonlinearities, state-variable and parameter estimation errors. This time Q2 is answered by the load and proper switching sequences had to be generated - of course - on-line. Advantages and disadvantages of all these techniques are well known and can be summarized as follows:

**MF techniques** are easy to design and work best for the steady-state operation.

**On-line optimized patterns** feature excellent transient behaviour and are doing outstandingly well when it comes to dealing with parameter and state-variable errors.

But is it really possible to find an optimum trade-off between switching losses and current distortion when using only one unique switching sequence (MF techniques) or leave the decision about firing angles totally up to the load (on-line optimized patterns)? Although the development of MF techniques, for instance, can be traced back in the literature for more than 20 years, a simple showcase example given in the next subsection indicates, that there is still room for improvement.

### SHOWCASE EXAMPLE

Taking for instance Fig. 3 as a basis, the inverter delivers a current vector \( \vec{i} \) that trails 30° behind the reference voltage vector \( \vec{U}_{\text{ref}} \). Usually, the reference voltage vector is synthesized by adjacent vectors \( \vec{U}_a, \vec{U}_b, \) and \( \vec{U}_c \) with sequence (I), which has minimal turn-on and turn-off count (Fig. 4). Sequence (I) is employed in almost all MF techniques which operate in the continuous mode, and in most of the vector modulators as well. Another sequence (II), for instance, wraps around voltage vectors in a different way. Compared with sequence (I), sequence (II) demands a higher (33%) switching frequency. Fig. 4 gives an additional insight into the switching states of the inverter, as only the active components are drawn. The particular topologies are derived from the actual current vector \( \vec{i} \) (direction of each phase current) and the switching state of the inverter. During the transition to another inverter state it is seen that an IGBT either breaks the actual phase current or is turned on into an existing forward diode current. Both switching operations (ON/OFF) cause rather different losses. Fig. 7 illustrates these losses as a function of the phase current level that has to be commutated. We are now in the position to compare the energy loss of each sequence. And, surprisingly, the energy balance is clearly in favour of the sequence with the increased switching frequency (II), because far more energy is dissipated.
for switching \( i_b \) ("1" once), than for switching \( i_a \) and \( i_c \) ("1/2" twice). This is clearly demonstrated by the energy balance reflected in Fig. 4. Due to the nonlinearity of switching losses, the showcase example shows clearly, that there are considerable discrepancies between switching losses and pulse frequency.

Moreover, sequence (I) also produces a lower current ripple. The error triangles prescribed by the arrowhead of the current vector (Fig. 3) share the same centre of gravity, whereas the error triangles related to sequence (I) are back-to-back aligned. Evidently, sequence (I) cannot be the best choice. But is sequence (II) the only choice? Are there any other sequences competing with sequence (II), and if so, how long should each sequence be asserted, before transition to the next becomes necessary? It is another intent of the paper to clarify these questions by submitting the control problem to a CAD program. The CAD program presented is able to reach an optimum trade-off between switching losses and current distortion. The program development takes three stages:

- **KNOWLEDGE ENGINEERING**
  - PULSE PATTERNS SYNTHESIS
  - POST PROCESSING

---

**CAD APPROACH**

**KNOWLEDGE ENGINEERING**

Since primary emphasis is on system efficiency, switching losses of the inverter and the impact of current distortion on additional machine losses represent the input data pool (Knowledge Domain), processed by the CAD program. Input data preparation (Knowledge Engineering) is outlined in the following two paragraphs.

a) **Switching Losses (Inverter)**

Turn on - Fig. 5 shows typical switching waveforms for a clamped inductive load at various phase current levels. When the IGBT is turned on into an existing diode current, the collector current is ramped up a specific \( di/dt \) rate, irrespective of whether the forward current was high or low. A unique \( di/dt \) rate, only determined by a given set of circuit conditions \((U_d, L_{po}, \delta_j, R_G)\) is actually a very desirable feature, because it makes the calculation and prediction of turn-on losses easy. Thus, turn-on intervals and current levels are almost strictly proportional

\[
\text{ton} \sim i. 
\]  

(1)

Since the diode is still in the conducting state, the full dc-link voltage (minus \( L_{po} \) voltage drop) appears on the collector. A constant voltage across the IGBT
during this back-to-back operation causes turn-on losses to become prevalent and again facilitates their prediction. After the diode regains its blocking capability, the collector current settles to the load current (soft recovery) and the collector voltage falls to the saturation voltage. Because the turn-on interval according to \( I \) increases along with the current, turn-on losses are traceable to the square of the actual phase current, which has to be commutated. Energy dissipation at turn-on may be approximated by

\[
W = \int u \cdot i \, dt = t \left( I^2 \right),
\]

\[
W = \frac{1}{2} (U_d - L_p \frac{di}{dt})(i + i_{RM}(i)) \cdot t_{on}(i).
\]

But the evaluation of turn-on losses should cover both: turn-on losses of the IGBT and turn-off losses of the associated diode caused by its soft recovery. To improve the accuracy, hence, experimental investigation is preferred, because the influence of unintelligible design, layout and device characteristics is eliminated this way. The quantity of energy that a complete inverter leg dissipates at turn-on is then found by a simple energy balance. Extensive measurements of the power draw and the power delivery of a single inverter leg were performed at phase current intervals of 10 A. The difference, i.e., the energy dissipation, is shown in Fig. 7.

**Turn off** - Fig. 6 shows typical turn-off waveforms for an IGBT in a clamped inductive load switching application, where collector current (actual phase current) is the parameter subjected to change. As soon as the IGBT is turned off, the collector voltage starts to rise. Operated in a snubberless environment, the collector current cannot be bypassed to a RCD network, because it simply does not exist. That is why the full load current continues to flow through the IGBT until the collector emitter voltage reaches (and exceeds) the dc-link voltage. At this moment, the associated free-wheeling diode, high-side or low-side, respectively, is forward biased and the IGBT commutates the current to the diode. Turn-off waveforms differ from turn-on waveforms in that the switching interval is not affected by the actual current level. IGBTs are capable of breaking hundreds of Amperes as fast as tens of Amperes. This is a serious problem for overcurrent shut-down and Power Modul design. Internal wiring impedances should be kept as low as possible and at the same time diodes should have a fast forward recovery to get rid of the voltage overshoot the IGBT displays during turn-off. The procedure of measuring the turn-off losses was carried out in analogy to the measurement of turn-on losses. This time, turn-off losses of a complete inverter leg appeared to be almost strictly proportional to the actual current level commutated. The result is
shown in Fig. 7.

Fig. 7 makes up one part of the Knowledge Domain and completes the Knowledge Engineering on inverter switching losses. The showcase example given above has already outlined how to access the Knowledge Domain.

b) Additional Motor Losses due to Current Distortion

This is a very difficult problem to cope with. According to IEEE Standard 519-1981 [3] from the machine's point of view, any pulse pattern is considered to be optimal that causes a RMS value of the harmonic content in the phase currents of up to 5%. In this case no derating of the inverter-fed machine due to current distortion is required and further reduction of the current ripple is not reasonable here. As a matter of fact, specification of the distortion factor is not necessary for a demonstration of the working principle of the CAD program. This concludes the Knowledge Engineering.

PULSE PATTERN SYNTHESIS

The description of the optimization problem given so far qualifies Bellman's Dynamic Programming D. P. superbly for use as a shell program to base the CAD approach on. Two important points will underline this. Firstly, D. P. is concerned with the solution of optimization problems, which can be formulated as a sequence of decisions. This style is reminiscent of Q1 and Q2. Secondly, D. P. can easily be applied to control problems for nonlinear and time-varying systems and the more discontinuities and constraints there are on the state variables and the control, the easier the solution! Furthermore, completely elaborated scalar optimization algorithms built on D. P. can be extended to more dimensions without running into additional problems. So before entering into the final CAD approach (three phase VSI-IM), for the sake of clarity, a one-dimensional example (single phase inverter) is in order. This example is seen to deal with all of the nonlinearities encountered in connection with a VSI-fed induction machine. Hence, reduction of dimensions, even to one, is by no means an unfair simplification of the problem.

Optimal Control of a Discrete-Time System Using D. P.

Let the system depicted in Fig. 5 (single phase inverter-fed R-L-EMK load) be governed by a difference equation

\[ l(n+1) = 0.8l(n) + 2.0u(n) . \]  

(3)

According to the two-point characteristic of the switch-mode power supply, the control \( u(n) \) is constrained to take on values of \( u = 0 \) and \( u = 1 \). The task is to keep the output quantity \( i \) closely aligned to a constant reference value \( i_{\text{ref}}(n) = 7 \). Fig. 8 illustrates the control loop of the fixed command control, and the control problem is to find a switching strategy which minimizes the associated performance index \( J \).

\[ J = \sum_{n=0}^{N} \left( \left( i_{\text{ref}} - i(n) \right)^2 + C \cdot W_{\text{d}}(n, l(n), \omega_{\text{ref}}(0)) \right) \]

(4)

where the final time of interest is \( N = 21 \). The first term of the sum \( \left( e^2 \right) \) reflects the RMS deviation (i.e. current ripple) which should be minimized to lessen additional (machine) losses due to current distortion, see Knowledge Domain, paragraph (b). The second term is attributed to the switching losses a certain pulse pattern causes. By varying \( C \) the weight of the losses can be attenuated or reinforced. So every arbitrary trade-off between distortion and losses is achievable. It is an important point to note that the expenditure of energy for the desired RMS deviation is always an optimum.

D. P. Notation - In order to be able to formulate D. P. problems in a precise way, straightforward for computation, the state and control values must first be quantized, that is, restricted to some finite set of admissible values. Quantization levels (not required to be equidistant) selected are:

\[ \hat{i}_m(n) = (5.0 + m/2.0) |_{m=0.8} ; \quad u(n) = \begin{bmatrix} 0 \\ 1 \end{bmatrix} . \]  

(5)

While the levels for \( u(n) \) are given (Inverter leg), a finer quantization for \( i(n) \) would probably be selected in an actual application. Next, states and stages are introduced. The stage, indexed by a stage variable \( q \), indicates the number of stages which remain until a terminal state is reached. In fact, states correspond to the sampling instants, only the indexing is time-reversed. A state is actually a certain configuration of the system. In this particular application a state consists of the position of the
switch, indexed by $u$, the quantization level of the current $i$, indexed by $m$ and the sampling instant $n$ or the stage index $q$, respectively. The notation will be: $(u,m,q)$. Fig. 9 illustrates how to describe the system’s behaviour with states and also shows the correct indexing. The two restrictions $i \geq 5.0$ and $i \leq 9.0$ are meant to discard unreasonable solutions.

Restricting the control range is often referred to as “junking” and very popular in the field of AI-research. In order to circumvent the extension to the case of Boltzmann models the initial value of the state variable $i(0)$ may be given and the final state $i(21)$ is free (transversality condition).

from the final stage (backward-in-time problem). To begin, let $q = 0$ and write

$$L(0, m, 0) = L(1, m, 0) = (i_{ref} - \hat{i}_m(21))^2$$

(6)

which is the penalty, or the value of the states assigned to states via a matrix $L(u,m,q)$, for ending up in the state variable values $i_m$ at the final time $N = 21$. According to (3), the final control action occurs at $n = 20$. Consequently, no decision is required at stage 0 and only the RMS deviation contributes to the cost index (4). At stage 1 and ascending stages in turn there is a set of feasible actions (i.e., switching the inverter) assigned to each state. Inherent to state $(0,0,1)$, however, there is only one possible action: $u = 1$. Keeping the inverter idle ($u = 0$) to save on switching losses violates the $i \leq 5.0$ restriction. The state trajectory from this node would otherwise terminate at a value of $i(21) = 0.8 \times 5.0 = 4.0$ and exceed the control range. This is a very good example to demonstrate how restrictions serve to accelerate the optimization. For $(0,0,1)$ the optimal plan ($u = 1$) or the optimal policy was frankly determined without invoking any algorithm. The optimal plan will be stored in a matrix $P(u,m,q)$ or will be inserted into Fig. 9 with an arrow. Hence, $P(0,0,1) = 1$. In general most of the decisions are not dictated by the restrictions and in accordance with the performance index (4) a choice must be made which minimizes equ. (4) or in other words generates an optimum return. At every state the return with respect to control actions available can be calculated by the recurrence relation

$$\begin{align*}
L(0, m, q=1) &= [i_{ref} - \hat{i}_m(20)]^2 + \\
&+ \min \left\{ C \cdot W_{dy}(on, \hat{i}_m(20)) + L(1, m, q=0) \right\}
\end{align*}$$

(7)

which turns out to be the main computational D.P. algorithm. The first term on the right hand side takes the RMS deviation of the state $(u,m,q=1)$ itself into account. The second term is split up to give each return corresponding to a decision $u = 1$ (upper) and $u = 0$ (lower). Being at state $(0, m, 1)$ the inverter must be switched to move to state $(1, m = 1, 0)$. This particular transition will cause turn-on losses $W_{dy}(on, \hat{i}_m)$ and a cost to go of $L(1, m, 0)$ representing the optimal cost of the rest of the state trajectory. At $q = 1$ the cost to go degenerates to the RMS deviation of the final states at $q = 0$. If the switch idles, no switching losses are generated, and only a cost to go of $L(0, m, 0)$ is accumulated. In accordance with the given objective to minimize the
performance index \( J \) (equation 4) the minimum found according to equ. (7) provides the optimal action - stored in \( P \) - and the optimal value of the state stored in \( L \). In a similar fashion, the values of states of the stealth plane are computed

\[
L(1,m,q=1) = \left[ i_{ref} - \hat{\omega}_m(20) \right]^2 +
+ \min\left\{ C \cdot W_d(y_{off}, \hat{\omega}_m(20)) + L(0, \overline{m}, q=0) \right\}
\]

(8)

By successively incrementing \( q \) and continuing to compare the control possibilities allowed by the principle of optimality, the remainder of the \( P \) and \( L \) matrix elements can be initialized. The optimal control is then expressed as a state-variable feedback in graphical (following the arrows, see Fig. 11) or tabular (\( P \) matrix) form. The grid or the matrix \( P \) is valid for arbitrary initial states and must only be re-done if the weight of the switching losses (\( C \)) is changed. If \( C \) is chosen to take on a value of zero, switching losses are cancelled. The algorithm then passes the system through its stages, while keeping the RMS deviation as low as possible. Consequently, a rather low current ripple can be expected, however, at the expense of the switching losses. Fig. 9 shows another solution to this extreme value problem where the RMS deviation was disregarded (\( C \to \infty \). The current ripple is quite high, because the emphasis is now on the switching losses. By varying \( C \) a program run synthesizes any optimal pulse pattern, because both questions Q1 and Q2 are submitted to the algorithm. In order to straighten out the design process, the two returns -RMS deviation and switching losses- that are definitely a function of \( C \) can be calculated in advance. Due to the simplicity of the example, the RMS deviation is an ascending progression and switching losses a descending progression. There are no local maxima or minima. Fig. 10 points out the expenditure of switching energy \( J_w \) for a desired RMS deviation \( J_{rms} \). So \( C \) can be picked from Fig. 10 and the associated optimal pulse pattern is going to be synthesized in the first program run, thus reversing the design procedure.

As the stage-index exceeds \( q=1 \), two minor practical problems are observed. No matter how finely \( \hat{\omega}_m \) is quantized, relation (3) might result in values that do not coincide with quantization levels. In this event, the values of \( L \) are derived from a simple interpolation [4]. This was already being denoted by the bar in equ. (7) and (8). The other problem concerning the plan that should be followed in such a state-miss is solved with a Post Processor.

**POST PROCESSOR**

If a state-miss occurs (Fig. 11) the plan of the state must be followed, that increases the slope of the sub-trajectory involved. In other words, the trend of the actual movement (converging on the reference value or drifting away) must be reinforced. This choice prevents the discretization error from piling up, and invites optimal trajectory control [5].

![Fig. 11: Post Processor](image)

Fig. 11 illustrates that the inverter is switched instantly if the actual trajectory reaches the theoretical one.

Extension of the one-dimensional CAD approach to the complete VSI-IM system is quite straightforward. Equ. (3) must be replaced by the system equation of an induction machine, and suitable reference tracks can be derived from Blaschke's field oriented control, for instance. The result of the complete CAD approach is depicted in Fig. 12.
Although the highly nonlinear optimization problem has been solved the technical application of Dynamic Programming schemes on-line in the area of adjustable speed drives isn't mature until the next generation of RISC computers is available at affordable price. For the time being this approach can be readily applied to drives which are running in the steady-state condition most of the time. Like in former table based SHE PWMs (selected harmonic elimination) the optimal pulse patterns of a few operating points can be stored in look-up tables and read back during operation. The Bellman approach may also be used as a training program to learn, how optimal pulse patterns should look like. So from this CAD program very good tentative, but maybe not entirely optimal pulse patterns can be derived and implemented on low cost computing platforms.

CONCLUSION

The paper points out that a reduction of switching frequency does not always correspond to a reduction of switching losses. Sometimes it is better to switch more often at rather low current levels. This applies especially when turn-on and turn-off losses are highly nonlinear like in the voltage source inverter with IGBT, now commonplace. In order to determine these current levels, CAD support is one of the essentials. A detailed description of the CAD program is intended to save the effort of going through a similar design process each time a switch-mode power supply needs to be charged with an optimal PWM pattern. In addition, the results of launching the program can easily be incorporated into existing drive systems in a non intrusive way to enhance their efficiency. So computer aided pulse pattern synthesis can have a profound influence on power conversion efficiency. We emphasized the merits of Bellman's D. P. which has not yet gained the credit and recognition it deserves.

REFERENCES


