

UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO
EM ENGENHARIA ELÉTRICA

Jordan Pauleski Zucuni

MODULAÇÃO *SPACE VECTOR* ÓTIMA PARA
CONVERSORES ESTÁTICOS

Santa Maria, RS, Brasil
2019

Jordan Pauleski Zucuni

**MODULAÇÃO *SPACE VECTOR* ÓTIMA PARA CONVERSORES
ESTÁTICOS**

Dissertação apresentada ao Curso de Mestrado do Programa de Pós-Graduação em Engenharia Elétrica, Área de Concentração em Processamento de Energia Elétrica, da Universidade Federal de Santa Maria (UFSM-RS), como requisito parcial para obtenção do grau de **Mestre em Engenharia Elétrica**.

Orientador:
Humberto Pinheiro, Prof. Ph.D.

Santa Maria, RS, Brasil
2019

Ficha catalográfica elaborada através do Programa de Geração Automática da Biblioteca Central da UFSM, com os dados fornecidos pelo(a) autor(a).

Zucuni, Jordan Pauleski

Modulação *Space Vector* Ótima Para Conversores Estáticos /
Jordan Pauleski Zucuni - 2019

141 p.; 30 cm

Orientador: Humberto Pinheiro, Ph.D

Dissertação (mestrado) – Universidade Federal de Santa
Maria, Centro de Tecnologia, Programa de Pós-Graduação em
Engenharia Elétrica, RS, 2019

Modulação ótima *Space Vector*, MPC, Função Custo

© 2019

Todos os direitos autorais reservados a Jordan Pauleski Zucuni. A reprodução de partes ou do todo deste trabalho só poderá ser feita com autorização por escrito do autor.

Endereço: Av. Roraima, Nº 1000, Bairro Camobi, Santa Maria, RS, Brasil, CEP: 97105-900;

Fone: (55) 9 9100 5262;

Endereço Eletrônico: jzucuni@gmail.com.

Jordan Pauleski Zucuni

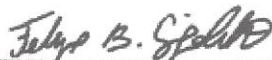
MODULAÇÃO *SPACE VECTOR* ÓTIMA PARA CONVERSORES
ESTÁTICOS

Dissertação apresentada ao Curso de Mestrado do Programa de Pós-Graduação em Engenharia Elétrica, Área de Concentração em Processamento de Energia Elétrica, da Universidade Federal de Santa Maria (UFSM-RS), como requisito parcial para obtenção do grau de Mestre em Engenharia Elétrica.

Aprovado em 03 de Junho de 2019:



Humberto Pinheiro, Prof. Ph.D. (UFSM)
(Presidente/ Orientador)



Felipe Bovolini Grigoletto, Dr. (UNIPAMPA)



Fernanda de Moraes Carnielutti, Dr^a. (UFSM)

Santa Maria, RS, Brasil
2019

RESUMO

MODULAÇÃO *SPACE VECTOR* ÓTIMA PARA CONVERSORES ESTÁTICOS

AUTOR: JORDAN PAULESKI ZUCUNI
ORIENTADOR: HUMBERTO PINHEIRO

Este trabalho propõe uma metodologia de modulação *Space Vector* ótima aos moldes da abordagem FCS-MPC para conversores estáticos de modo a basear-se em uma função custo para decidir, dentre os vetores redundantes, aqueles que irão compor a sequência de comutação. Para tanto, são necessários algoritmos computacionais para a detecção dos vetores que englobam a referência, para o cálculo das razões cíclicas, avaliação da função custo e aplicação dos vetores selecionados ao conversor em um período de amostragem. Dessa maneira, a metodologia é desenvolvida para três conversores: i) Monofásico Ponte Completa; ii) Trifásico Dois Níveis; iii) Trifásico NPC três níveis. No caso do último, além da tensão de saída a sintetizar, é necessário realizar o equilíbrio das tensões nos capacitores internos da topologia, processo o qual pode ser feito através da correta formulação da função custo. Por fim, é descrito o modulador vetorial desenvolvido, o qual é essencial para a implementação da técnica apresentada, o qual é capaz de aplicar, de modo simples, a sequência de comutação com as respectivas razões cíclicas. Para tanto, os resultados são obtidos através do emulador Typhoon HIL - Hardware In The Loop.

Palavras-chave: Modulação *Space Vector* ótima, *Model Predictive Control*, Função Custo.

ABSTRACT

**OPTIMAL SPACE VECTOR MODULATION
FOR STATIC CONVERTERS**

AUTHOR: JORDAN PAULESKI ZUCUNI
ADVISOR: HUMBERTO PINHEIRO

This work proposes an optimal space vector modulation methodology based on the FCS-MPC approach for static converters relying on a cost function in order to decide, among the redundant vectors, those which forms the switching sequence. Thus, computational algorithms are necessary for the detection of the vectors that encompass the reference, for the duty cycle calculation, cost function evaluation and application of the selected vectors over one sampling period. In this way, the proposed methodology is developed for three converters: i) Single Phase Full Bridge; ii) Three-Phase Two-Level; iii) Three-Phase NPC. In the last case, besides the output voltage to synthetize it is necessary to implement the DC bus capacitors voltage balancing, making it possible through the correct cost function formulation. Finally, a space vector modulator is described, which is essencial to implement the presented technique, being able to apply, in a simple way, any switching sequence with the associated duty cycles.

Keywords: Optimal Space Vector Modulation, Model Predictive Control, Cost Function.

LISTA DE FIGURAS

Figura 1.1	– (a) Forma de onda de um sinal PWM, denominado S_x , a ser aplicado em um interruptor (b).	20
Figura 1.2	– Etapas de comutação em um braço de inversor.	22
Figura 1.3	– (a) Inversor Half Bridge (b) Esquema de Modulação TCM (c) Formas de onda de saída.	24
Figura 1.4	– (a) Inversor Half Bridge (b) Modulação por comparação com portadora (c) Formas de onda da portadora, sinal modulante e sinais PWM.	26
Figura 1.5	– Espaço geométrico das tensões de saída dos braços de um inversor Ponte Completa.	27
Figura 1.6	– Transformação linear de modo a relacionar as tensões de saída dos braços com as variáveis de interesse.	28
Figura 1.7	– (a) Esquema de modulação geométrica; (b) Formas de onda de saída.	29
Figura 1.8	– Inversor trifásico dois níveis a três fios.	29
Figura 1.9	– (a) Tabela com os vetores de comutação para o inversor trifásico da Figura 1.8; (b) Digrama vetorial do inversor em coordenadas $\alpha\beta$	30
Figura 2.1	– Inversor trifásico dois níveis conectado à rede.	36
Figura 2.2	– Estratégia 1, $T_s = 100 \mu s$: (a) Correntes de saída. (b) Resposta transitória para um degrau de fase de 180°	40
Figura 2.3	– Estratégia 1, $T_s = 30 \mu s$: (a) Correntes de Saída. (b) Resposta transitória para um degrau de fase de 180°	40
Figura 2.4	– Estratégia 2, $T_s = 100 \mu s$: (a) Correntes de Saída. (b) Resposta transitória para um degrau de fase de 180°	41
Figura 2.5	– Espectro harmônico para os três casos considerados.	42
Figura 2.6	– (a) Correntes de linha para um indutor de saída $L = 500 \mu H$. (b) Oscilação na escolha dos setores na transição entre eles.	42
Figura 3.1	– Inversor Monofásico Ponte Completa.	45
Figura 3.2	– Diagrama vetorial para o Inversor Monofásico Ponte Completa.	46
Figura 3.3	– Fluxograma dos passos realizados pelo algoritmo de seleção dos vetores redundantes a serem aplicados ao conversor.	51
Figura 3.4	– Corrente de saída e tensão de linha considerando a sequência de comutação (3.5) e a função custo (3.8).	56
Figura 3.5	– Tensão de linha de saída e a distribuição dos pulsos sobre interruptores.	56
Figura 3.6	– Demonstração da lógica do segundo termo da função custo da equação (3.10).	58
Figura 3.7	– Seção 5 do fluxograma da Figura 3.3 reformulada para contemplar a função custo (3.10).	59
Figura 3.8	– Corrente de saída e tensão de linha considerando a sequência de comutação (3.5) e a função custo (3.10).	60
Figura 3.9	– Tensão de linha de saída e a distribuição dos pulsos sobre interruptores.	60
Figura 4.1	– Inversor Trifásico Dois Níveis.	63
Figura 4.2	– Diagrama vetorial em coordenadas abc para o conversor trifásico dois níveis.	64
Figura 4.3	– Detecção dos vetores que formam uma sequência anti-horária	67
Figura 4.4	– Fluxograma com os passos da rotina de modulação para o inversor trifásico dois níveis.	71

Figura 4.5	– Saturação em módulo para um vetor de referência fora dos limites do diagrama vetorial.	72
Figura 4.6	– Correntes de saída e tensões de linha considerando a sequência de comutação (4.6) e a função custo (4.15).	77
Figura 4.7	– Distribuição dos pulsos entre os interruptores.	77
Figura 4.8	– Atualização do Bloco 6 do fluxograma da Figura 4.4 com o novo critério de seleção dos vetores.	79
Figura 4.9	– Correntes de saída e tensões de linha considerando o fluxograma da Figura 4.8.	79
Figura 4.10	– Distribuição dos pulsos entre os interruptores.	80
Figura 5.1	– Inversor trifásico NPC.	81
Figura 5.2	– Diagrama vetorial para o inversor trifásico NPC.	84
Figura 5.3	– Exemplo para a metodologia de escolha dos vetores que compõem a sequência de comutação.	87
Figura 5.4	– Fluxograma com os passos da rotina de modulação para o inversor trifásico NPC.	91
Figura 5.5	– Correntes de saída e tensões de linha considerando a sequência de comutação (5.3) e a função custo (5.10).	99
Figura 5.6	– Tensões sobre os capacitores internos, tanto em acoplamento CC como em CA para evidenciar a pequena oscilação de tensão.	99
Figura 5.7	– Tensão de linha de saída e distribuição dos pulsos entre os interruptores.	100
Figura 5.8	– Tensão de linha de saída e distribuição dos pulsos entre os interruptores.	100
Figura A.1	– Estrutura de dados para o modulador vetorial.	115
Figura A.2	– Arquitetura interna do modulador vetorial.	119
Figura A.3	– Diagrama de tempo dos sinais envolvidos na geração dos pulsos PWM.	120
Figura A.4	– Mapa de memória contendo os registradores do modulador vetorial.	120
Figura B.1	– Função periódica com período T	129
Figura B.2	– Função periódica quadrada e Série de Fourier sobreposta com harmônicas de até 7 ^a ordem.	131
Figura B.3	– Sinal PWM relacionado à razão cíclica d	132
Figura B.4	– Instante de comutação do sinal modulante em relação à portadora.	136
Figura B.5	– Espectro harmônico para um sinal PWM obtido através da comparação de uma modulante senoidal com uma portadora triangular.	139
Figura B.6	– Inversor monofásico Ponte Completa.	140
Figura B.7	– Cancelamento de harmônicos em um inversor Ponte Completa: Tensões de fase e de linha correspondente.	141

LISTA DE TABELAS

Tabela 2.1	–	Parâmetros considerados para implementação prática das estratégias MPC apresentadas.....	40
Tabela 2.2	–	Tempo Computacional.....	41
Tabela 3.1	–	Mapemento dos vetores no espaço das tensões de linha para o espaço dos interruptores.....	49
Tabela 4.1	–	Mapemento dos vetores no espaço das tensões de linha para o espaço dos interruptores.....	70
Tabela 5.1	–	Tensão de fase em função dos estados de s_{xa} e s_{xb} , $x \in \{1..3\}$	82
Tabela 5.2	–	Mapeamento dos vetores no espaço das tensões de linha para o espaço dos interruptores.....	83

LISTA DE ABREVIATURAS E SIGLAS

CC	Corrente Contínua
CA	Corrente Alternada
DSP	<i>Digital Signal Processor</i> - Processador de Sinais Digitais.
FCS–MPC	<i>Finite Controle Set - MPC</i> - Conjunto Finito de Controle - MPC.
FPGA	<i>Field Programmable Gate Array</i> - Arranjo de Portas Programáveis em Campo.
MPC	<i>Model Predictive Controle</i> - Controle Preditivo Por Modelo
NPC	<i>Neutral Point Clamped</i> - Grampeamento do Ponto Central
OSS–MPC	<i>Optimal Switching Sequence - MPC</i> - Sequência de Comutação Ótima - MPC.
OSV–MPC	<i>Optimal Switch Vector - MPC</i> - Vetor de Comutação Ótimo - MPC.
PWM	<i>Pulse Width Modulation</i> - Modulação Por Largura de Pulso
RMS	<i>Root Mean Square</i> - Raiz do Valor Quadrático Médio
ZVS	<i>Zero Voltage Switching</i> - Comutação em Zero de Tensão

LISTA DE APÊNDICES E ANEXOS

Apêndice A – MODULADOR VETORIAL - FPGA	113
Apêndice B – SÉRIE DUPLA DE FOURIER PARA SINAL PWM	129

SUMÁRIO

1 INTRODUÇÃO E REVISÃO BIBLIOGRÁFICA	19
1.1 INTRODUÇÃO GERAL	19
1.2 MODULAÇÃO PWM COM PORTADORA	25
1.3 ABORDAGEM GEOMÉTRICA	26
1.3.1 Modulação Vetorial (<i>Space Vector</i>)	28
1.4 OBJETIVOS DA DISSERTAÇÃO	31
1.4.1 Objetivos Específicos	31
1.5 ORGANIZAÇÃO DA DISSERTAÇÃO	32
2 CONTROLE PREDITIVO BASEADO EM MODELO - MPC	33
2.1 INTRODUÇÃO	33
2.2 ESTUDO DE CASO: INVERSOR TRIFÁSICO DOIS NÍVEIS CONECTADO À REDE	36
2.2.1 FCS-OSV	37
2.2.2 FCS-OSS M²PC	38
2.3 COMPENSAÇÃO DO ATRASO DE TRANSPORTE	39
2.4 RESULTADOS EXPERIMENTAIS	39
2.5 CONCLUSÃO	42
3 MODULAÇÃO VETORIAL ÓTIMA PARA CONVERSOR MONOFÁSICO	45
3.1 INTRODUÇÃO	45
3.2 INVERSOR PONTE COMPLETA	45
3.3 ESTRATÉGIA DE MODULAÇÃO PROPOSTA	46
3.3.1 Detecção dos dois vetores mais próximos	47
3.3.2 Escolha da Sequência de Comutação	48
3.3.3 Mapeamento no espaço das tensões de fase e no dos interruptores	48
3.3.4 Função Custo	49
3.4 FLUXOGRAMA	50
3.5 RESULTADOS EXPERIMENTAIS: FUNÇÃO CUSTO 1	55
3.5.1 Discussão dos Resultados	55
3.6 RESULTADOS EXPERIMENTAIS: FUNÇÃO CUSTO 2	57
3.6.1 Discussão dos Resultados	60
3.7 CONCLUSÃO	61
4 MODULAÇÃO VETORIAL ÓTIMA PARA CONVERSOR TRIFÁSICO DOIS NÍVEIS	63
4.1 INTRODUÇÃO	63
4.2 INVERSOR TRIFÁSICO DOIS NÍVEIS	63

4.3 ESTRATÉGIA DE MODULAÇÃO PROPOSTA	64
4.3.1 Detecção dos três vetores mais próximos	64
4.3.2 Escolha da sequência de comutação	66
4.3.3 Mapeamento no espaço das tensões de fase e no dos interruptores	68
4.3.4 Função custo	69
4.4 FLUXOGRAMA	70
4.5 RESULTADOS EXPERIMENTAIS: SEQUÊNCIA 1	76
4.5.1 Discussão dos Resultados	77
4.6 RESULTADOS EXPERIMENTAIS: SEQUÊNCIA 2	78
4.6.1 Discussão dos Resultados	78
4.7 CONCLUSÃO	80
5 MODULAÇÃO VETORIAL ÓTIMA PARA CONVER-	
SOR TRIFÁSICO NPC	81
5.1 INTRODUÇÃO	81
5.2 INVERSOR TRIFÁSICO NPC	81
5.3 ESTRATÉGIA DE MODULAÇÃO PROPOSTA	84
5.3.1 Detecção dos três vetores mais próximos	85
5.3.2 Escolha da sequência de comutação	86
5.3.3 Mapeamento no espaço das tensões de fase e no dos interruptores	87
5.3.4 Função custo	89
5.4 FLUXOGRAMA	90
5.5 RESULTADOS EXPERIMENTAIS	98
5.5.1 Discussão dos Resultados	98
5.6 CONCLUSÃO	101
6 CONSIDERAÇÕES FINAIS	103
REFERÊNCIAS	105
APÊNDICES	111
B.1 SÉRIE DUPLA DE FOURIER	133

1 INTRODUÇÃO E REVISÃO BIBLIOGRÁFICA

1.1 INTRODUÇÃO GERAL

Este capítulo tem por finalidade contextualizar o leitor acerca da estratégia de modulação desenvolvida. Para tanto, uma revisão bibliográfica com as principais estratégias de modulação é realizada em um primeiro momento, as quais consistem da modulação PWM por Portadora, Modulação Geométrica e Vetorial. Embasando-se na última, a técnica é, então, desenvolvida para três conversores, de forma a torná-la ótima. Antes, porém, é necessário visualizar a importância do estudo das técnicas de modulação em um cenário global.

Neste contexto, cabe observar que o consumo de energia elétrica está fortemente relacionado ao desenvolvimento humano. A presença da energia elétrica pode ser observada desde a mais simples atividade corriqueira até aquelas de maior complexidade, a citar as presentes nos ambientes comerciais e industriais. Inúmeros são os equipamentos eletroeletrônicos utilizados pela sociedade com as mais diversas finalidades.

A energia elétrica, para ser utilizada da maneira como a conhecemos, necessita ser gerada através de processos de conversão energética. Diversas são as plantas de geração elétrica, fazendo uso de diferentes fontes de energia primária, como a hidráulica, a eólica e a solar (THOMAS et al., 2017; NETO et al., 2015; YEN; TSAI; BAI, 2009). É importante ressaltar que a energia elétrica disponível por tais unidades geradoras apresentam-se sob diferentes condições, podendo ser de corrente contínua ou alternada, apresentando amplitude e frequência variáveis, ainda, com capacidade de potência não constante.

Como consequência, estágios de processamento de energia elétrica são requeridos de modo a adequar a energia convertida aos padrões necessários para consumo e interligação ao sistema elétrico. Neste ponto, encontra-se a eletrônica de potência por meio dos conversores estáticos. O princípio de funcionamento dos últimos embasa-se em interruptores semicondutores comutando em alta frequência a fim de sintetizar e controlar grandezas de interesse, como a corrente injetada na rede por um inversor.

O princípio de funcionamento dos conversores estáticos de potência, pode, no entanto, ser expandido para outros dispositivos. Nesse contexto, encontra-se o controle de máquinas elétricas, capazes de regular velocidade e torque em motores, além da potência fornecida no caso de geradores (DE ALMEIDA; LOPES; BARREIROS, 2004). Ainda, pode-se citar aquelas aplicações destinadas à performance do sistema elétrico, como os sistemas de compensação de reativos DSTATCOM (TISCHER; SCHERER; DE CAMARGO, 2015; PAWAR; DESHPANDE; MURALI, 2015), além daquelas referentes à integração de sistemas de energia renovável (MIN et al., 2007; DESAI; SHAH, 2013) e à

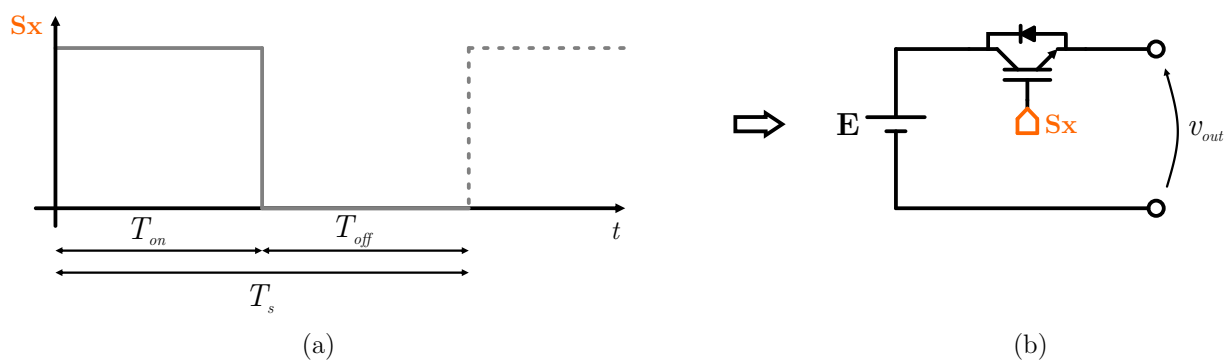
fontes ininterruptas de energia (DONG et al., 2015; MOHAMADIAN et al., 2016).

Em qualquer caso, o projeto de equipamentos de eletrônica de potência envolvem uma configuração de interruptores designada por topologia, devendo ser adequada para o fim com que o conversor é proposto (ver Figura 1.4). Determinadas topologias podem atingir números elevados de interruptores para comutar, sendo que esses devem ser acionados de maneira satisfatória, a fim de ser possível o controle da variável de interesse, observando fatores de mérito a exemplo da distorção harmônica total - THD, a taxa de distorção harmônica ponderada - WTHD, espectro harmônico, número de comutações e padrão de pulsos na saída (SCHNICK et al., 2008; WEIBIN; YANRU; SHUN, 2006).

Modulação é o termo designado à maneira de acionar tais interruptores de modo a atingir os resultados desejados (GUBIA et al., 2007). Deve-se observar que, conforme a topologia empregada, as possíveis combinações entre eles podem atingir valores extremamente grandes, tornando complexa a tarefa de encontrar aquelas que atendem às especificações de projeto. Além disso, a solução pode não ser única ou, ainda, não otimizada, restringindo a exploração dos recursos do conversor de potência. Conseqüentemente, a formulação/implementação de uma estratégia de comutação que atinja os objetivos desejados pode se tornar uma tarefa não trivial.

O padrão de comutação mais comum é aquele referente à modulação por largura de pulso (PWM - do inglês, *Pulse Width Modulation*). Através dessa, os interruptores permanecem uma parcela de tempo no estado alto e outra no estado baixo, de forma que a soma desses tempos (chamada de período de comutação T_s) permanece sempre fixa - Figura 1.1.

Figura 1.1 – (a) Forma de onda de um sinal PWM, denominado S_x , a ser aplicado em um interruptor (b).



Fonte: Autor.

Dessa maneira, o valor médio da variável a ser sintetizada se dá pela razão entre os tempos do estado alto e baixo em relação à T_s , a chamada razão cíclica (*duty cycle*, do

inglês). Por exemplo, pode-se definir a razão cíclica d como:

$$d = \frac{T_{on}}{T_s} \quad (1.1)$$

Normalmente, a modulação PWM é desejada, uma vez que, apresentando frequência de comutação fixa (dada pelo inverso do período de comutação T_s), torna-se mais simples o projeto dos filtros de saída, uma vez que o espectro da tensão de saída apresenta um comportamento bem conhecido. Além disso, fatores adicionais se tornam mais previsíveis, como a temperatura de trabalho do conversor, devido ao aquecimento dos interruptores e o ruído audível, tendo o primeiro impacto no projeto de dissipadores de calor.

Cabe ressaltar, também, que atualmente os sistemas de controle são implementados de maneira digital. Nesses, através de uma interrupção, se amostram, de maneira periódica, as variáveis necessárias ao controle; a rotina referente à ação de controle é executada e os valores nos registradores relacionados à implementação da modulação são atualizados.

A parcela de tempo entre dois desses instantes é chamada de período de amostragem, geralmente designada também por T_s , sendo o seu inverso denominado de frequência de amostragem. Como a rotina executada em cada período de amostragem consome certo tempo computacional, deve-se garantir que o microprocessador seja capaz de lidar com a demanda de processamento sem se estender ao próximo período de amostragem. Como as interrupções de amostragem devem estar sincronizadas com os instantes de atualização do padrão PWM, se torna mais fácil assegurar que o processamento da rotina de controle se dará de forma adequada, uma vez que o padrão de chaveamento apresenta período de comutação fixo.

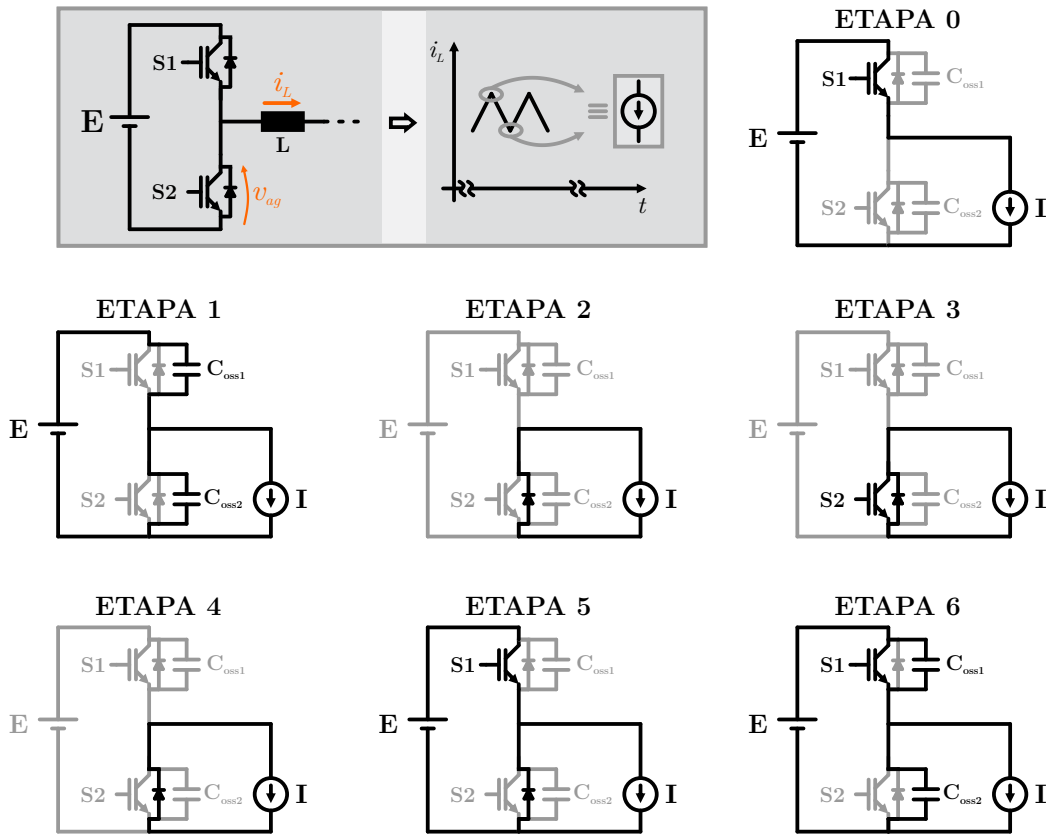
No entanto, há outras formas de modulação, a citar a Modulação Triangular de Corrente - TCM (*Triangular Current Modulation*) (MARXGUT et al., 2014). Diferentemente do padrão PWM, a última é voltada à comutação suave de braços de inversores, realizando a troca dos estados dos interruptores somente em instantes propícios, quando a corrente na saída do braço troca de direção, de modo a garantir comutação ZVS (*Zero Voltage Switching*, do inglês) (NEUMAYR et al., 2017).

A fim de melhor elucidar essa técnica, considere as etapas de comutação em um braço de inversor apresentadas na Figura 1.2, onde concentra-se na análise da comutação entre os interruptores S1 e S2 para uma corrente de saída sempre positiva.

ETAPA 0: S1 conduz toda corrente I . O capacitor C_{oss1} está descarregado, enquanto que sobre C_{oss2} mantém-se a tensão E .

ETAPA 1: No momento em que S1 sai do estado de condução, a corrente I se divide entre as capacitâncias C_{oss1} e C_{oss2} , aumentando a tensão sobre C_{oss1} ao passo que se diminui a sobre C_{oss2} . Como um capacitor não admite variações bruscas em seus

Figura 1.2 – Etapas de comutação em um braço de inversor.



Fonte: Autor.

valores de tensão, e como inicialmente C_{oss1} encontra-se descarregado, pode-se afirmar que S1 abre em zero de tensão (ZVS). Pode-se perceber que ambos interruptores S1 e S2 permanecem desligados. A duração em que eles permanecem nesse estado compreende ao tempo morto, intervalo de segurança para evitar que S1 e S2 conduzam ao mesmo tempo, uma vez que a comutação dos interruptores não se dá de maneira instantânea.

ETAPA 2: Quando a tensão sobre o capacitor C_{oss2} chegar a zero, o diodo de corpo do interruptor S2 entra em condução, assumindo toda corrente I .

ETAPA 3: Como o diodo de corpo de S2 encontra-se em condução, a tensão sobre esse interruptor é nula. Conseqüentemente, S2 comuta em zero de tensão. Normalmente, S2 é acionado para resultar em menores perdas de condução caso a corrente I fosse mantida apenas pelo diodo e para garantir o nível de tensão de saída caso haja inversão de corrente.

ETAPA 4: O Interruptor S2 sai de condução, em zero de tensão, dando início ao segundo intervalo de tempo morto.

ETAPA 5: Etapa onde ocorre comutação forçada no interruptor S1. Como o diodo de S2 ainda está em condução, grampeando sua tensão em zero, sobre S1 encontra-se toda tensão E . A corrente sobre S1 começa a subir ao passo que a do diodo diminui. Quando

a última chegar a zero, o diodo sai de condução, dando início à ETAPA 6.

ETAPA 6: Uma vez com o diodo de S2 bloqueado, as tensões sobre os capacitores C_{oss1} e C_{oss2} começam a variar até que C_{oss1} se descarregue e C_{oss2} fique com a tensão E .

Através das etapas descritas, fica fácil observar que se antes da entrada de S1 em condução (ETAPA 5) a corrente inverte o sentido, de modo que o diodo de corpo de S1 entre em condução, S1 entrará em condução em ZVS. Dessa forma, os interruptores são acionados de modo que o sentido de corrente no indutor seja propício para a comutação suave. A análise feita na Figura 1.2 é análoga para o sentido de corrente entrando no braço.

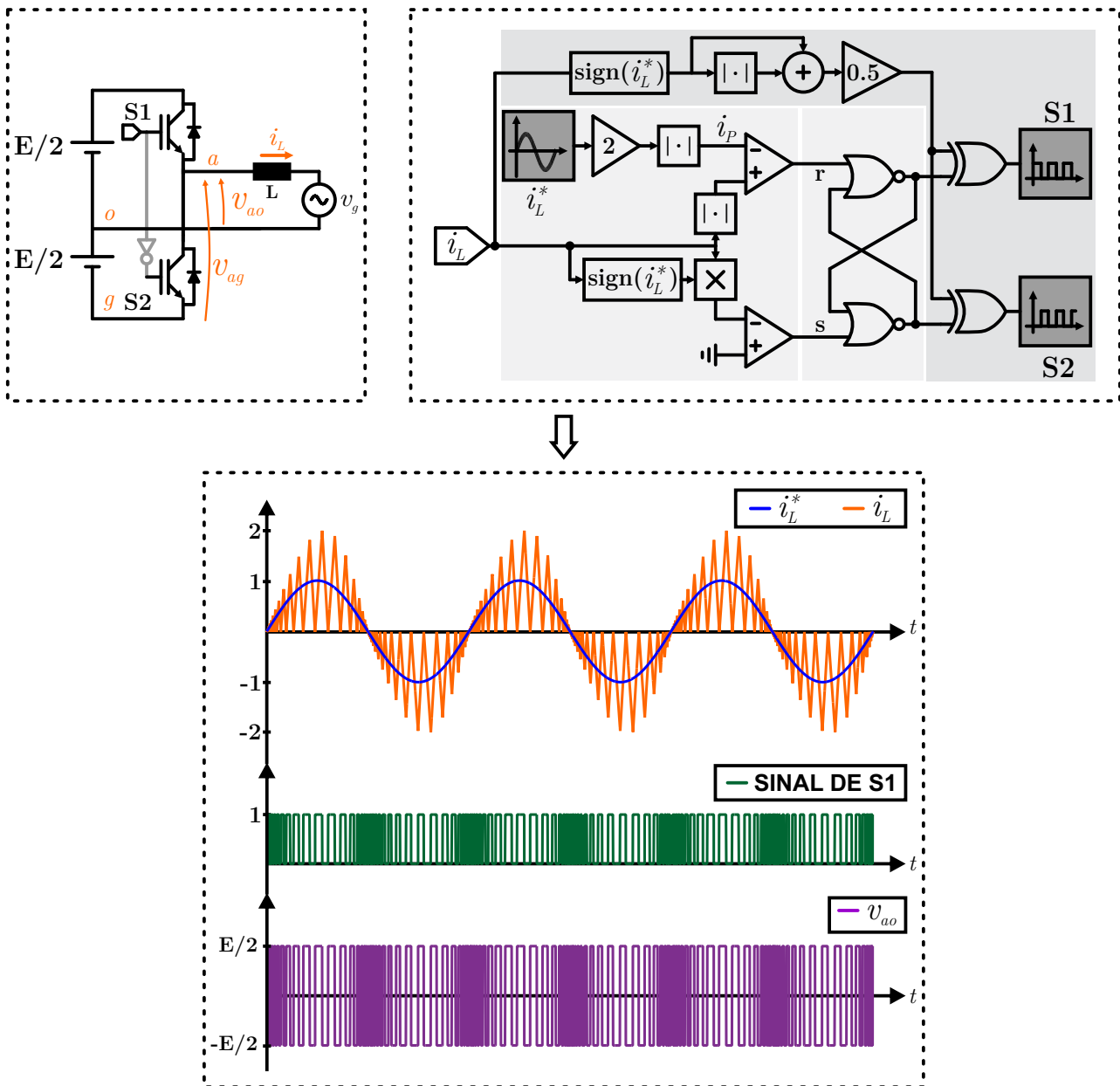
O resultado da modulação TCM é uma forma de onda triangular, conforme mostrado na Figura 1.3, com alta ondulação de corrente e frequência variável. Pode-se observar que o valor de pico da corrente precisa ser o dobro da de referência, sendo a frequência inversamente proporcional ao valor de pico. Nas proximidades do cruzamento por zero, a frequência de comutação aumenta consideravelmente, o que pode trazer danos para os semicondutores e para a rotina de controle, devendo-se limitar a frequência de comutação (KNECHT; BORTIS; KOLAR, 2016).

Além disso, pela alta ondulação de corrente apresentada, normalmente a modulação TCM é aplicada em configurações com braços intercalados (*interleaved*, do inglês) (MARXGUT; BIELA; KOLAR, 2010), de modo a suavizar a ondulação na saída do conversor. Dessa forma, a modulação TCM normalmente é utilizada quando se deseja alta eficiência, devendo ser levada em conta a variação de frequência, juntamente com as consequências no projeto do filtro, no ruído audível e na execução da rotina de controle, conforme mencionado anteriormente.

Uma outra metodologia muito difundida é a Model Predictive Control - MPC aplicada à eletrônica de potência (VAZQUEZ et al., 2017; GEYER; QUEVEDO, 2014a). Por meio dela, há a possibilidade de mesclar controle e modulação em um mesmo estágio. Basicamente, há a geração de uma interrupção em intervalos fixos de tempo, onde a rotina de controle é executada. A rotina, então, seleciona, dentre todas as combinações de interruptores possíveis (chamadas de vetores de comutação), aquela que resulta no menor valor de uma função custo definida pelo projetista, baseando-se, para tanto, no modelo da planta.

A função custo pode englobar vários objetivos desejados, a citar o rastreamento da referência, o equilíbrio de tensões em capacitores e número reduzido de comutações. Entre as desvantagens desse método encontram-se o alto esforço computacional, a frequência de comutação variável e o espectro harmônico fortemente espraído. Essa abordagem apresenta muito afinidade com a proposta desta dissertação e, por isso, ela é discutida em detalhes no Capítulo 2.

Figura 1.3 – (a) Inversor Half Bridge (b) Esquema de Modulação TCM (c) Formas de onda de saída.



Fonte: Autor.

Apesar da existência de diferentes opções de modulação, o padrão PWM é, sem dúvidas, o mais utilizado. O objeto de estudo torna-se, então, como gerá-lo, dada uma topologia específica. No campo da eletrônica de potência, existem várias estratégias de modulação PWM, a citar: PWM com portadora (exemplificado na Figura 1.4), abordagem geométrica e vetorial (conhecida também pelo termo em inglês *space vector*). Cada uma delas apresenta particularidades que se tornam mais atraentes de acordo com o problema a ser resolvido pelo projetista de eletrônica de potência. Na sequência, essas abordagens

são apresentadas.

1.2 MODULAÇÃO PWM COM PORTADORA

Esta estratégia consiste em gerar um sinal PWM através da comparação de um sinal modulante com uma onda portadora linear, normalmente dente de serra ou triangular. Pode-se definir estado alto toda vez que o valor instantâneo do sinal modulante for maior que o da portadora e baixo quando do contrário.

Para exemplificar a aplicação da modulação PWM, pode-se considerar a Figura 1.4. Em (a), encontra-se o esquemático de um inversor meia ponte (*Half Bridge*, do inglês), o qual consiste de dois interruptores que comutam de maneira complementar. Isto é, quando um está conduzindo, o outro encontra-se bloqueado. Observa-se, portanto, que, quando S1 está conduzindo, a tensão v_{ao} corresponde a $E/2$, e, quando do contrário, v_{ao} corresponde a $-E/2$.

Com o objetivo de gerar o sinal PWM que será aplicado à S1, compara-se um sinal modulante com uma portadora que consiste de uma onda triangular variando entre -1 e 1 . Toda vez que o sinal modulante for maior que a portadora, S1 encontra-se em condução, enquanto, do contrário, em modo de bloqueio. Esse sinal consiste daquele desejado na saída, com amplitude normalizada em relação ao valor de pico da portadora. Dessa forma, desejando uma tensão de saída v_{ao} senoidal, pode-se definir:

$$v_m^*(t) = m_i \sin(\omega t + \phi) \quad (1.2)$$

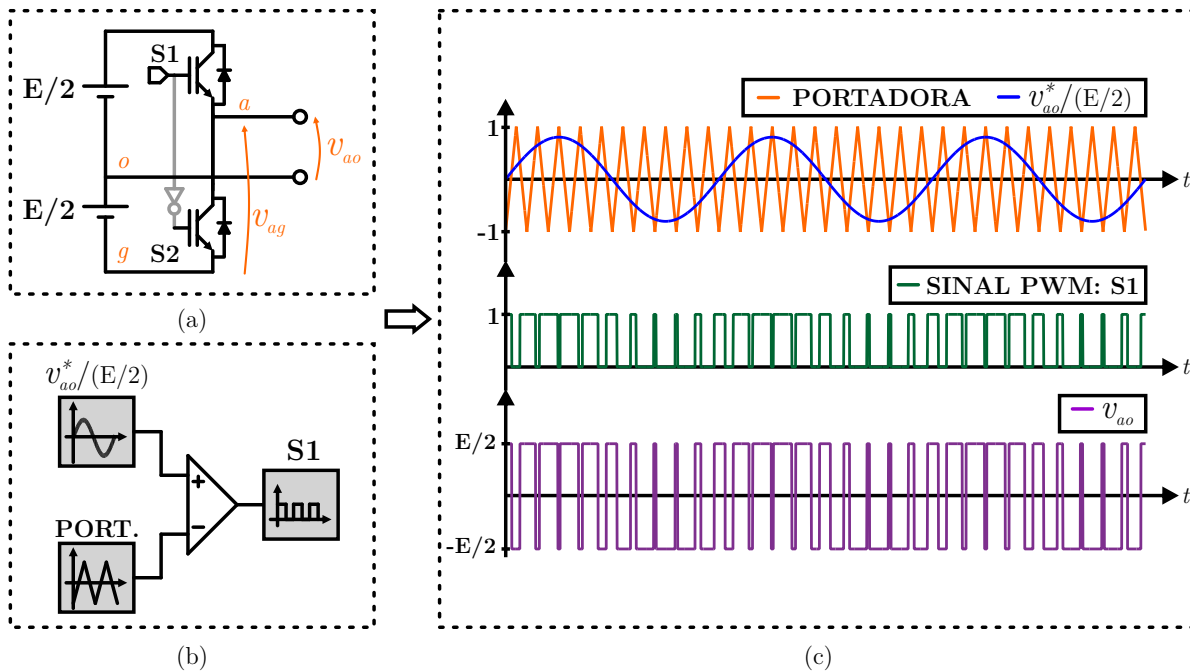
Onde v_m^* é o sinal modulante; m_i , o índice de modulação (razão entre o pico da senoide e o pico da portadora); ω , a frequência angular; e ϕ , a defasagem angular. Na figura 1.4 (b), encontra-se o esquema de geração do sinal PWM e, em (c), as formas de onda resultantes. Pode-se observar que quando a amplitude do sinal modulante corresponde a um, a tensão média de saída v_{ao} corresponde à $E/2$.

Através dessa estratégia, fica-se relativamente simples de se prever o espectro harmônico de saída e se propor soluções para o cancelamento de harmônicos. Além disso, torna-se relativamente fácil estendê-la para conversores multiníveis, através de metodologias como phase-shift e phase-disposition (CARRARA et al., 1992; MCGRATH; HOLMES, 2002a).

O espectro de um sinal PWM pode ser obtido através da aplicação da série dupla de Fourier (BENNETT, 1933). O seu desenvolvimento bem como a forma do espectro resultante para o caso de um sinal PWM obtido pela comparação de um sinal modulante com uma portadora encontra-se no Apêndice B.

O resultado, considerando conversores alimentados em tensão, é uma tensão pulsada que contém um valor médio. No caso da Figura 1.1, o valor médio da tensão de

Figura 1.4 – (a) Inversor Half Bridge (b) Modulação por comparação com portadora (c) Formas de onda da portadora, sinal modulante e sinais PWM.



Fonte: Autor.

saída pode ser dado por $v_{out} = dE$. Por se tratar de uma tensão de saída pulsada, são necessários filtros, normalmente volumosos e pesados, a fim de se atenuar os harmônicos indesejados.

1.3 ABORDAGEM GEOMÉTRICA

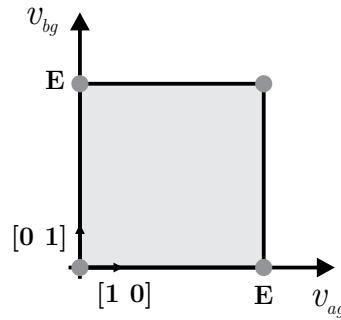
A abordagem geométrica (RYAN; LORENZ; DE DONCKER, 1999; ZHOU; WANG, 2002), também baseada na comparação de um sinal modulante com uma portadora triangular, permite explorar os graus de liberdade do conversor, a exemplo da tensão de modo comum entre os braços de um inversor ponte completa - Figura 3.1. Neste caso, as possíveis tensões nas saídas dos braços dos conversores são expressas através de uma base ortonormal onde cada componente refere às tensões dos braços do inversor. Após, uma transformação linear é realizada, onde se relaciona as tensões anteriores com as variáveis de interesse.

Para fins de esclarecimento, considere as tensões v_{ag} e v_{bg} da Figura Figura 3.1. Note que elas podem ser expressas sob o espaço linear \mathbb{R}^2 conforme a Figura 1.5. Como consequência, através de um padrão PWM, qualquer vetor de tensão que se encontre na área hachurada pode ser sintetizado pelo inversor. No entanto, normalmente uma variável de interesse é a tensão diferencial $v_{ab} = v_a - v_b$. Dessa maneira, a seguinte transformação

linear pode ser feita:

$$\begin{bmatrix} v_{ab} \\ v_0 \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 0,5 & 0,5 \end{bmatrix} \begin{bmatrix} v_{ag} \\ v_{bg} \end{bmatrix} \quad (1.3)$$

Figura 1.5 – Espaço geométrico das tensões de saída dos braços de um inversor Ponte Completa.



Fonte: Autor.

Para que a matriz do operador linear da equação (1.3) não seja singular, uma segunda variável, v_0 , deve ser definida. Neste caso, ela é a média das duas tensões de saída dos braços, constituindo, portanto, a tensão de modo comum. Dessa maneira, através da abordagem geométrica, é possível trabalhar não somente com a tensão diferencial de saída, mas, também, com todos os graus de liberdade do conversor.

Realizando a relação inversa da equação (1.3), tem-se:

$$\begin{bmatrix} v_{ag} \\ v_{bg} \end{bmatrix} = \begin{bmatrix} 0,5 & 1 \\ -0,5 & 1 \end{bmatrix} \begin{bmatrix} v_{ab} \\ v_0 \end{bmatrix} \quad (1.4)$$

Onde as colunas do operador linear da equação (1.4) podem ser interpretadas como os vetores que definem a base que leva as tensões de saída e de modo comum para o espaço linear das tensões sobre os braços do inversor. Essa relação pode ser expressa na Figura 1.6.

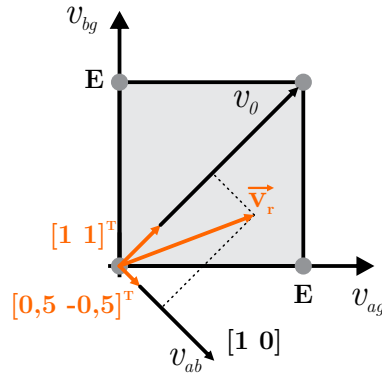
Conforme ilustrado na Figura 1.6, o vetor médio \vec{v}_r necessita estar dentro da área hachurada. Como consequência, restrições necessitam ser impostas. Assim, pode-se verificar os seguintes limites:

$$\begin{aligned} 0 &\leq v_{ag} \leq E \\ 0 &\leq v_{bg} \leq E \end{aligned} \quad (1.5)$$

Logo, realizando a transformação da equação (1.4), pode-se reescrever (1.5) como:

$$\begin{aligned} 0 &\leq 0,5v_{ab} + v_0 \leq E \\ 0 &\leq -0,5v_{ab} + v_0 \leq E \end{aligned} \quad (1.6)$$

Figura 1.6 – Transformação linear de modo a relacionar as tensões de saída dos braços com as variáveis de interesse.



Fonte: Autor.

Como normalmente deseja-se maximizar o valor médio da tensão de saída \bar{v}_{ab} , é conveniente rearranjar (1.6) para se obter os limites dos valores médios \bar{v}_0 em função de \bar{v}_{ab} . Assim obtém-se as seguintes restrições:

$$\begin{aligned} -0,5\bar{v}_{ab} &\leq \bar{v}_0 \leq E - 0,5\bar{v}_{ab} \\ 0,5\bar{v}_{ab} &\leq \bar{v}_0 \leq E + 0,5\bar{v}_{ab} \end{aligned} \quad (1.7)$$

Analisando as desigualdades de (1.7), é possível reescrevê-las da seguinte maneira:

$$0,5|\bar{v}_{ab}| \leq \bar{v}_0 \leq E - 0,5|\bar{v}_{ab}| \quad (1.8)$$

Então, para \bar{v}_{ab} tem-se ainda a possibilidade de se ter diferentes valores de \bar{v}_0 . Consequentemente, pode-se escolher, por exemplo, \bar{v}_0 como sendo o mínimo valor possível:

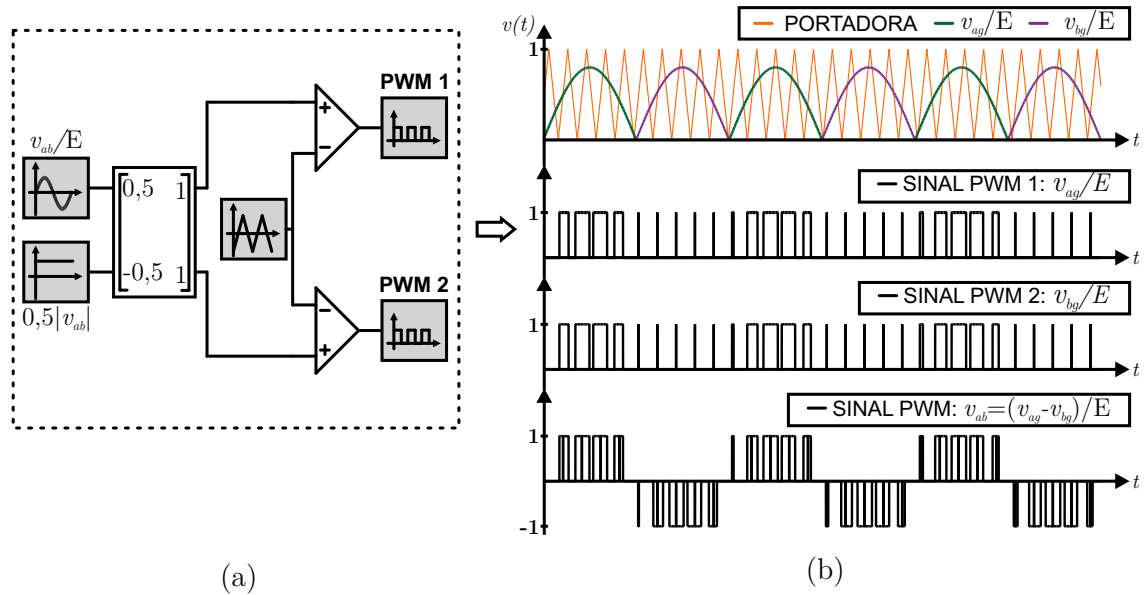
$$\bar{v}_0 = 0,5|\bar{v}_{ab}| \quad (1.9)$$

O mesmo procedimento anterior pode ser realizado de maneira análoga para outros conversores. A geração do sinal PWM em si é também realizada pela comparação de sinais modulantes com uma portadora. Para o exemplo dado, tem-se a modulação esquematizada na Figura 1.7, com as respectivas formas de onda da portadora, dos sinais modulantes e dos consequentes sinais PWM.

1.3.1 Modulação Vetorial (*Space Vector*)

Nesta modulação, também conhecida por Modulação *Space Vector*, o espaço de todas as possíveis tensões na saída nos braços do inversor é mapeado em um diagrama sob a forma de vetores de comutação (KUMAR; CHATTERJEE, 2017; BABU; AGARWAL, 2016). Através desse diagrama (diagrama *space vector*) e de acordo com o vetor de

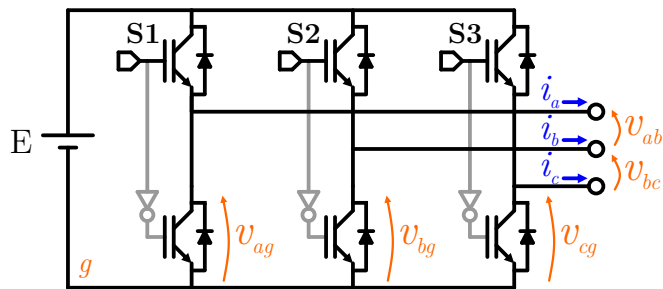
Figura 1.7 – (a) Esquema de modulação geométrica; (b) Formas de onda de saída.



Fonte: Autor.

referência de tensão, é escolhido um conjunto de vetores comutação, normalmente os mais próximos da referência, de modo a sintetizar a tensão desejada (A; AGARWAL, 2018). A região delimitada por eles é, então, chamada de setor. O mesmo procedimento pode ser realizado de maneira análoga para conversores alimentados em corrente. Para fins de clareza, considere o inversor trifásico da Figura 1.8.

Figura 1.8 – Inversor trifásico dois níveis a três fios.



Fonte: Autor.

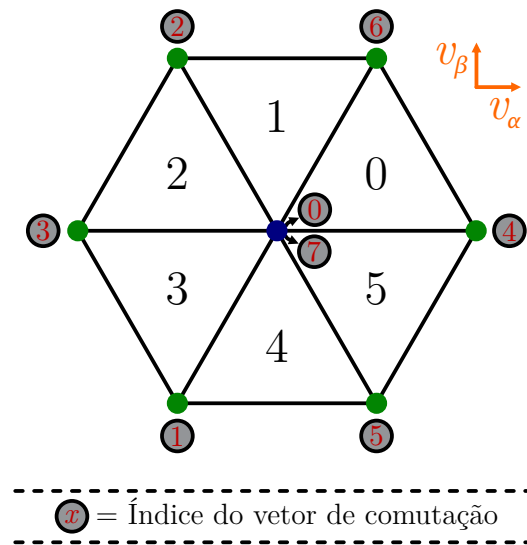
As possíveis tensões de saída, v_{ab} , v_{bc} , do inversor da Figura 1.8 podem ser representadas, considerando a transformação linear para o sistema de coordenadas $\alpha\beta$, de acordo com a Figura 1.9.

Dessa maneira, uma vez com o vetor de referência de tensão, é possível escolher os $n + 1$ vetores mais próximos que englobam a referência, de maneira a sintetizar as tensões de saída desejadas, onde n é a dimensão do espaço dos vetores de comutação. Conseqüentemente, no caso da Figura 1.9, escolhe-se os três vetores mais próximos (MONGE et al.,

Figura 1.9 – (a) Tabela com os vetores de comutação para o inversor trifásico da Figura 1.8; (b) Digrama vetorial do inversor em coordenadas $\alpha\beta$.

Índice	S1	S2	S3	v_α	v_β
0	0	0	0	0	0
1	0	0	1	$-1/3$	$-\sqrt{3}/2$
2	0	1	0	$-1/3$	$\sqrt{3}/2$
3	0	1	1	$-2/3$	0
4	1	0	0	$2/3$	0
5	1	0	1	$1/3$	$-\sqrt{3}/2$
6	1	1	0	$1/3$	$\sqrt{3}/2$
7	1	1	1	0	0

(a)



(b)

Fonte: Autor.

2004). É importante observar que, desde que o poliedro definido pelos vetores de comutação selecionados englobem a referência, será possível sintetizar a tensão de saída. Porém, os $n + 1$ vetores são selecionados a fim de se obter menor THD na saída (STOJILOVIC; PEJOVIC, 2012).

Uma vez determinado o setor, deve-se calcular as durações de cada um dos vetores que o definem, também chamadas de razão cíclica (do inglês, *duty cycle* ou *dwell time*). Posteriormente, uma sequência de comutação deve ser definida, isto é, uma forma de distribuir os vetores, sendo mandatório que cada um dos $n + 1$ vetores apareça pelo menos uma vez ao longo do período de amostragem. Por fim, uma maneira de implementar a sequência de comutação deve ser formulada.

Uma grande vantagem da abordagem vetorial consiste na liberdade da escolha da sequência de comutação. Através dela, pode-se, por exemplo, distribuir os pulsos dos interruptores de maneira a tornar a comutação entre eles mais homogênea, ou, ainda, otimizar o padrão dos pulsos PWM da saída, melhorar a taxa de distorção harmônica, controlar a corrente no divisor capacitivo como no caso de um conversor NPC, etc (ABDULVELEEV; KHRAMSHIN; KORNILOV, 2016).

Há várias maneiras de se implementar uma modulação vetorial. Normalmente, as etapas anteriormente descritas são realizadas *offline* (de forma pré-processada), de modo a economizar esforço computacional. Porém, é possível, também, realizar uma modulação vetorial de maneira *online* (em tempo real de execução da rotina), a exemplo

de quando o conversor se torna complexo para realizar um algoritmo *offline* destinado à seleção dos vetores de comutação, como no caso de conversores multiníveis (BROECK; SKUDELNY; STANKE, 1988; CHAMARTHI; CHHETRI; AGARWAL, 2016). Outro caso a exemplificar é quando se tem variáveis intermediárias a controlar, como a tensão nos capacitores de conversores, também no caso de multiníveis.

1.4 OBJETIVOS DA DISSERTAÇÃO

Dada a revisão sobre as estratégias apresentadas, já consolidadas na literatura, observa-se que cada uma apresenta suas particularidades, estando a seleção dentre delas atreladas ao objetivo desejado pelo projetista de eletrônica de potência. No entanto, é natural cogitar uma estratégia de modulação alternativa, flexível e dinâmica, que possa combinar cada uma das vantagens das estratégias apresentadas, podendo formular uma abordagem de modulação que possa ser adaptada ao objetivo que o projetista desejar.

Neste contexto, surge a proposta desta dissertação, a qual consiste em elaborar uma metodologia de modulação *space vector online*, que possa ser estendida a qualquer conversor, implementada através de uma função custo, de modo a se aproximar de um problema de otimização, com padrões de pulsos ótimos e balanço de variáveis internas. Tal abordagem, apesar de não entrar no mérito do controle de variáveis externas, é muito semelhante ao que se tem na literatura conhecido por Controle Preditivo baseado em Modelo - MPC (*Model Predictive Control*, do inglês) aplicado à eletrônica de potência, pois aproveita a natureza discreta do conversor para a avaliação de uma função custo.

A abordagem MPC consiste em um método que se baseia no modelo matemático da planta para gerar a ação de controle e que se subdivide em algumas subclassificações, como explicitadas no Capítulo 2. Duas delas consistem em escolher, de modo direto, vetores de comutação a serem aplicados à planta, conhecidos como FCS-OSV-MPC e FCS-OSS-MPC. Essas últimas fornecem o embasamento para a abordagem de modulação diferenciada deste trabalho.

1.4.1 Objetivos Específicos

Desenvolver uma metodologia de modulação vetorial baseada em função custo, a qual tem por foco apenas a geração da tensão de referência. Isto é, que, apesar da semelhança, não combine, como no MPC, controle e modulação, explorando apenas a disposição dos vetores de comutação no período de amostragem, conforme Capítulos 3, 4 e 5. Deseja-se, para tanto, validar resultados experimentais através do uso do emulador de *hardware* Typhoon HIL402.

Ainda, objetiva-se desenvolver um modulador vetorial que implemente qualquer sequência de comutação através de uma FPGA, conforme apêndice A.

1.5 ORGANIZAÇÃO DA DISSERTAÇÃO

Primeiramente, a abordagem MPC é explanada no Capítulo 2, a qual embasa a técnica de modulação proposta. Neste capítulo, fundamenta-se, ainda, a importância de uma abordagem diferente da MPC, que trate apenas da modulação em si. Após, nos Capítulos 3, 4, 5, a técnica de modulação *space vector* ótima é desenvolvida, respectivamente, para os conversores Monofásico Ponte Completa, Trifásico Dois Níveis e Trifásico NPC 3 níveis. Por fim, encerra-se a dissertação com considerações finais, seguidas do Apêndice A, o qual descreve o *hardware* necessário para a implementação da técnica proposta.

2 CONTROLE PREDITIVO BASEADO EM MODELO - MPC

2.1 INTRODUÇÃO

Este capítulo tem, por objetivo, introduzir o leitor no contexto do Controle Preditivo Baseado em Modelo - MPC. Para tanto, é realizado um resumo de tal técnica aplicada à eletrônica de potência, visto que tal metodologia fornece a ideia básica para o desenvolvimento da modulação vetorial ótima. Neste contexto, duas técnicas MPC são apresentadas, uma considerando frequência de comutação variável e outra, fixa. Resultados experimentais comparando as duas também são mostrados. Por fim, fecha-se o Capítulo com as conclusões, embasando a finalidade da técnica proposta.

Considerando o controle de conversores de eletrônica de potência, é possível diferenciar duas classes de controladores. Uma consiste em se ter o controle e o modulador desenvolvidos e implementados de forma independente, como é feito convencionalmente. Já a outra, consiste de apenas um estágio onde se encontram o controle e a modulação combinados (RODRIGUEZ et al., 2013).

Quanto à primeira classe, há uma variedade de trabalhos na literatura, consolidando-a por meio de técnicas bem conhecidas e estabelecidas, a exemplo de controladores lineares, já havendo estudos quanto à estabilidade, robustez e performance (OGATA, 2003). Por outro lado, a segunda técnica engloba controladores do tipo Histerese (KARAARSLAN, 2008), *Sliding Mode* (MUHAMAD; AZIZ, 2008) e *Finite Control Set - Model Predictive Control*, FCS-MPC (PANTEN; HOFFMANN; FUCHS, 2016). As duas primeiras já estão presentes na literatura há décadas e são adequadas para a implementação analógica. Já a metodologia FCS-MPC se baseia na natureza discreta do conversor, sendo propícia para a implementação em sistemas digitais.

O Controle Preditivo Baseado em Modelo, *Model Predictive Control* - MPC, tem sido largamente empregado na indústria, sendo conhecido desde as últimas três décadas, apresentando, primeiramente, aplicação na indústria de petróleo (LEE, 2011). No entanto, atualmente, a sua aplicação tem se expandido para outras áreas, alcançando a eletrônica de potência (GEYER; QUEVEDO, 2014b; KIEFERNDORF et al., 2012; Geyer; Papafotiou; MORARI, 2009; LARRINAGA et al., 2007), a exemplo do controle da potência e a corrente injetada na rede, além do balanço das tensões de capacitores inseridos em conversores estáticos com topologias multiníveis (MOON et al., 2017).

De acordo com (VAZQUEZ et al., 2017), MPC, na eletrônica de potência, pode ser dividido em duas grandes áreas: *Continuous Control Set* MPC (CCS-MPC) e o já mencionado FCS-MPC (AHMED; KOH; LEE, 2018). No primeiro caso, o sinal de controle, calculado por meio de controle preditivo, é aplicado ao conversor através de um modula-

dor convencional. Já o FCS-MPC, o qual tem se mostrado uma alternativa atraente para sua implementação, seleciona o vetor ou a combinação ótima de vetores de comutação através de uma função custo, como mostrado em (RODRIGUEZ et al., 2007; ZHANG et al., 2017). Nesta categoria, valores futuros das variáveis são preditos através do modelo da planta, para cada vetor de comutação. Baseando-se nas predições comparadas às referências através da minimização de uma função custo, escolhe-se diretamente os vetores de comutação a serem aplicados ao conversor.

É de se observar que essa metodologia se torna intuitiva ao mesmo tempo que se mostra simples em adicionar restrições e em se trabalhar com não-linearidades. Assim, problemas complexos tornam-se relativamente simples de solucionar, controlando diretamente as variáveis de interesse, podendo-se empregar a referida técnica a uma vasta gama de aplicações, incluindo especificações adicionais por parte do projetista.

Ainda de acordo com (VAZQUEZ et al., 2017), o FCS-MPC pode ser subdividido em dois grupos: o que emprega apenas um vetor por período de amostragem, chamado de *Optimal Switching Vector* - OSV-MPC e o que emprega uma sequência de vetores, denominado de *Optimal Switching Sequence* - OSS-MPC. Um ponto a frisar quanto à diferença entre os dois é que a técnica OSV-MPC pode repetir um vetor ao longo de dois ou mais períodos, implicando em uma frequência de comutação variável. Já por outro lado, a técnica OSS-MPC, por empregar uma sequência de comutação, garante frequência fixa de comutação (DONOSO et al., 2018).

Um dos primeiros trabalhos que retrataram o princípio do FCS-MPC apresenta-se através do controle direto de torque (*Direct Torque Control*) - DTC (TAKAHASHI; NOGUCHI, 1986). Já em (ZHANG; ZHANG, 2014), a ligeira diferença entre FCS-MPC e o controle direto de corrente (*Direct Current Control*) - DCC é explanada. Em ambos os casos, o princípio reside em encontrar um vetor de comutação que apresente um resultado desejado através de uma função custo. No entanto, DCC considera uma função custo específica, a qual pode ser pensada como uma caso particular do FCS-MPC, não necessitando realizar uma busca exaustiva envolvendo todos os vetores como no caso do FCS-MPC.

Cabe, neste ponto, ressaltar que a técnica MPC apresenta um alto custo computacional, sendo possível de ser implementada no âmbito da eletrônica de potência graças à disponibilidade de microcontroladores e FPGAs the alta performance. No entanto, a técnica mostra muito boa resposta tanto transitória como em regime permanente.

Ainda nessa linha, um ponto a considerar é o horizonte de predição, denominado por N , que pode ser de um ou de vários períodos de amostragem. Quando se diz $N = 1$, por exemplo, entende-se um horizonte de predição de um intervalo de amostragem à frente. No caso de um horizonte $N > 1$, o que se tem como resultado do controlador é a formação de um sequência de vetores ótimos ao longo do horizonte. No entanto, é aplicado ao conversor apenas o vetor pertencente à predição do primeiro período de amostragem.

Na próxima interrupção, a rotina recalcula as predições e, então, aplica a próxima ação de controle, utilizando o mesmo horizonte de predição. A esse ciclo se dá o nome de *receding horizon* (QUEVEDO; GOODWIN; DE DONA,).

Quanto ao impacto do horizonte de predição no resultado do controlador, é mostrado que horizontes de predição mais longos levam a melhores respostas em regime permanente (BORDONS; MONTERO, 2015). No entanto, o esforço computacional aumenta de modo exponencial com o tamanho do horizonte, em função das resultantes combinações possíveis entre os vetores de comutação (GEYER; QUEVEDO, 2014a).

A fim de exemplificar a técnica FCS-MPC, dois estudos de caso serão considerados, ambos para o inversor trifásico dois níveis: o caso em que se emprega a estratégia OSV-MPC e o que utiliza uma abordagem OSS-MPC. Objetiva-se, com isso, demonstrar a diferença entre as duas técnicas. A resposta em regime permanente, assim como a referente a transitórios será investigada.

Quanto à metodologia OSV-MPC, será considerada abordagem semelhante ao exposto em (RODRIGUEZ et al., 2007). No caso do inversor trifásico dois níveis da Figura 2.1, o conjunto finito de entradas de controle é composto por todos os oito vetores de comutação, presentes na Figura 1.9. Desse conjunto, apenas um vetor é selecionado a fim de minimizar uma dada função custo. No entanto, há a possibilidade, como já mencionado, de o controlador selecionar o mesmo vetor de comutação em dois períodos de amostragem subsequentes. Além disso, pelo fato de empregar apenas um vetor de comutação em cada amostragem, além de variável, resulta-se em uma baixa frequência de comutação.

A fim de contornar esse problema, há várias técnicas OSS-MPC (XIE et al., 2018; MORA et al., 2019). Nesta seção, será tratada a técnica chamada de Controle Preditivo Modulado Baseado em Modelo (*Modulated Model Predictive Control*) - M²PC (DONOSO et al., 2018), que organiza os vetores de comutação segundo a lógica de modulação vetorial. Basicamente, os vetores são organizados em setores, da mesma forma com que na modulação vetorial. No entanto, os custos são atribuídos aos setores.

Dessa maneira, não se tem a escolha de um único vetor; ao contrário, seleciona-se o setor que retorna o menor custo. A maneira como a sequência de comutação é formada pelos vetores que integram o setor escolhido pode ser qualquer, segundo critérios do projetista. De qualquer forma, por meio desta abordagem, uma frequência fixa de comutação será alcançada ao mesmo tempo que se mantém as vantagens do FCS-MPC, tais como formulação intuitiva, habilidade para facilmente desenvolver soluções multiobjetivo e inclusão de não linearidades de maneira direta.

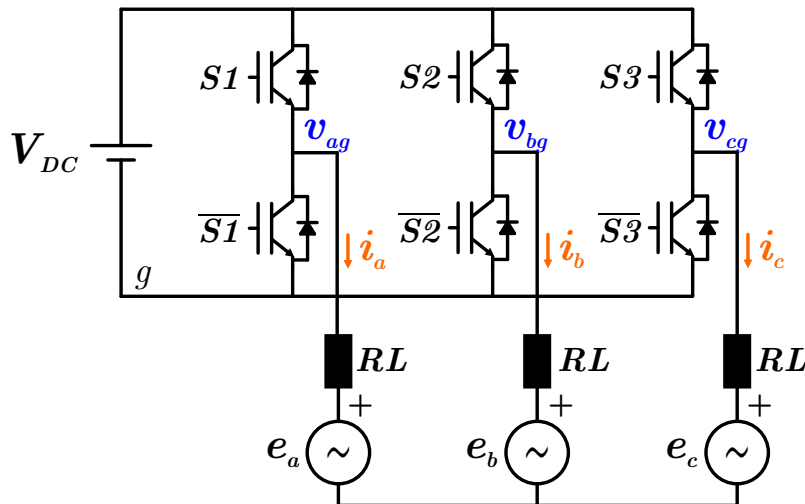
Em função da técnica MPC apresentar um grande esforço computacional (GEYER; QUEVEDO, 2014a), o tempo de processamento envolvido nas duas estratégias MPC serão explicitados. A abordagem OSV-MPC se torna um pouco mais simples que a OSS-MPC, o que implica em um tempo de processamento menor quando comparada à outra. Por outro lado, apesar da abordagem OSS-MPC demandar um maior tempo

computacional, ela apresenta as vantagens da modulação vetorial convencional. Dessa maneira, objetiva-se demonstrar, através da implementação das duas estratégias, suas vantagens e desvantagens.

2.2 ESTUDO DE CASO: INVERSOR TRIFÁSICO DOIS NÍVEIS CONECTADO À REDE

Nesta seção, o inversor trifásico dois níveis conectado à rede será descrito, bem como a estratégia de controle FCS-MPC para ele implementada. Esta topologia é tratada, em um primeiro momento, por sua simplicidade e devido ao fato de ser popular e largamente empregada em muitas aplicações de geração distribuída.

Figura 2.1 – Inversor trifásico dois níveis conectado à rede.



Fonte: Autor.

Cada braço do inversor da Figura 2.1 contém duas chaves que operam de maneira complementar, correspondendo a tensão de saída a 0 V ou V_{DC} . O diagrama vetorial do conversor já foi apresentado na Figura 1.9, bem como todos os possíveis estados dos interruptores com as correspondentes tensões de fase na referência $\alpha\beta$.

É importante notar que, neste caso, há uma redundância para o vetor nulo. Isto é, dois vetores de comutação diferentes resultam na mesma tensão de fase $\alpha\beta$. Isso pode ser observado na Figura 2.1 quando observadas a primeira e a última linha dos vetores de comutação. Cada triângulo do diagrama vetorial é chamado de setor e é denominado pelo número que se encontra em seu interior.

Considerando a referência $\alpha\beta$, as correntes de saída podem ser modeladas por:

$$\frac{di_{\alpha,\beta}}{dt} = \frac{1}{L} (-Ri_{\alpha,\beta} + v_{\alpha,\beta} - e_{\alpha,\beta}) \quad (2.1)$$

A equação (2.1) pode ser discretizada, considerando um período de amostragem T_s pela aproximação de Euler, resultando em:

$$i_{\alpha,\beta}[k+1] = i_{\alpha,\beta}[k] + \frac{T_s}{L} (-Ri_{\alpha,\beta}[k] + v_{\alpha,\beta}[k] - e_{\alpha,\beta}[k]) \quad (2.2)$$

Em uma forma compacta, a Equação (2.2) pode ser escrita como:

$$x[k+1] = ax[k] + b(v[k] - e[k]) \quad (2.3)$$

onde $a = 1 - \frac{RT_s}{L}$ e $b = \frac{T_s}{L}$. A fim de implementar as duas estratégias MPC, a equação (2.2) será utilizada.

2.2.1 FCS-OSV

Nesta estratégia, em que o seu princípio de funcionamento é apresentado em (RODRIGUEZ et al., 2007), em cada período de amostragem T_s , a seguinte função custo para o i -ésimo vetor é avaliada:

$$g_i = \|i^* - x_i^p\|^2, \quad i \in \{0...7\} \quad (2.4)$$

onde i^* e x_i^p representam, respectivamente, o vetor das correntes de referência e o i -ésimo vetor das correntes preditas no período de amostragem $k+1$. Uma vez com todos os oito custos avaliados, a rotina seleciona o vetor que resulta no menor custo, aplicando-o ao conversor por todo T_s .

No próximo instante de amostragem, o mesmo procedimento é repetido com os valores das variáveis atualizadas. O critério para a seleção do vetor é baseado apenas no valor de $g_i, i \in \{0...7\}$. Dessa forma, os vetores são escolhidos de maneira errática, sem, necessariamente, uma ordem. Consequentemente, um mesmo vetor pode ser aplicado consecutivamente, ocasionando uma frequência de comutação não fixa, com um espectro harmônico espraído como mostrado na Figura 2.5.

Esta é uma abordagem simples e intuitiva que pode incluir soluções multiobjetivo, lembrando um problema de otimização através de uma função custo. Tal função pode incluir mais termos e restrições com o objetivo de, por exemplo, evitar o uso do mesmo vetor consecutivamente ou impor que a tensão em capacitores permaneçam em certa faixa de tolerância. Ainda nessa linha, pode-se, por exemplo, fazer com que haja uma distribuição uniforme entre as comutações nos interruptores de modo a não haver discrepâncias no aquecimento deles.

2.2.2 FCS-OSS M²PC

Nesta estratégia, cujo princípio é apresentado em (DONOSO et al., 2018), a mesma função custo apresentada na equação (2.2) é empregada para cada um dos oito vetores de comutação. No entanto, ao invés de selecionar diretamente o vetor que a minimiza, um custo é associado a cada setor de seu diagrama vetorial - Figura 1.9. O setor com o menor custo é selecionado para ser aplicado ao inversor e a sequência de comutação pode ser qualquer uma, desde que cada um dos três vetores que compõe o setor escolhido apareça ao menos uma vez dentro do período de amostragem T_s . Para fins de implementação, a seguinte sequência foi adotada:

$$\mathbf{s} = \mathbf{s}_0 \mathbf{s}_1 \mathbf{s}_2 \mathbf{s}_1 \mathbf{s}_0 \quad (2.5)$$

onde \mathbf{s}_0 significa o vetor $[0 \ 0 \ 0]^T$, \mathbf{s}_1 é o primeiro vetor ativo encontrado quando percorrido o diagrama vetorial da Figura 1.9 no sentido anti-horário, e \mathbf{s}_2 é o vetor remanescente do setor sob consideração.

Para calcular o mencionado custo do j -ésimo setor, deriva-se equações baseadas na minimização do valor RMS (*Root Mean Square*) dos custos individuais dos vetores, ponderados pelas correspondentes razões cíclicas, encontrando a solução via condições de Karush-Kuhn-Tucker. As seguintes equações são utilizadas, sendo implementadas na ordem em que se seguem:

$$Q_j = \frac{1}{\sum g_{ji}^{-1}} \quad (2.6)$$

$$d_{ji} = \frac{Q_j}{g_{ji}} \quad (2.7)$$

$$G_j = \sum g_{ji} d_{ji}^2 \quad (2.8)$$

Para as equações acima, a variável genérica x_{ji} significa aquela associada ao i -ésimo vetor do j -ésimo setor. O termo Q_j é uma variável auxiliar introduzido na solução matemática; o custo g_{ji} é o mesmo da equação 2.2, e a variável d_{ji} refere-se à razão cíclica do vetor correspondente e G_j é o custo do setor de índice j .

Como três vetores de comutação são sempre aplicados em um T_s , esta estratégia apresenta frequência de comutação fixa ao mesmo tempo que mantém a lógica do FCS-MPC, podendo facilmente acrescentar não linearidades e restrições através da função custo.

2.3 COMPENSAÇÃO DO ATRASO DE TRANSPORTE

Em implementações práticas, é impossível, no mesmo instante de amostragem k , obter as variáveis medidas, executar a rotina de controle e aplicar o resultado ao inversor, uma vez que cada um desses processos leva uma janela de tempo. Dessa maneira, a saída do controle calculada no instante k será efetivamente aplicada ao conversor no instante $k+1$. Como consequência, as predições são realizadas considerando o intervalo de $k+1$ a $k+2$ (VAZQUEZ et al., 2015).

Pode-se depreender disso que os valores das variáveis em $k+1$ devem ser primeiramente estimadas com a saída de controle que está sendo aplicada em k (que foi calculada no instante anterior, $k-1$). Então, por meio desses valores, se prevê o valor das variáveis em $k+2$. Como consequência, necessita-se, também, estimar a referência em $k+2$. Normalmente a referência apresenta um comportamento já conhecido, o que a torna fácil de prever.

A fim de realizar a estimativa em $k+1$, a equação 2.2 pode ser utilizada, considerando, porém, os vetores aplicados no período de amostragem anterior. Contanto que T_s seja suficientemente pequeno, tanto a referência futura quanto os valores da tensão da rede podem ser consideradas constantes. Para baixas frequências de amostragem, no entanto, assumir uma tensão de rede constante pode não ser muito prudente. Nesse caso, uma extrapolação por Lagrange pode ser utilizada.

2.4 RESULTADOS EXPERIMENTAIS

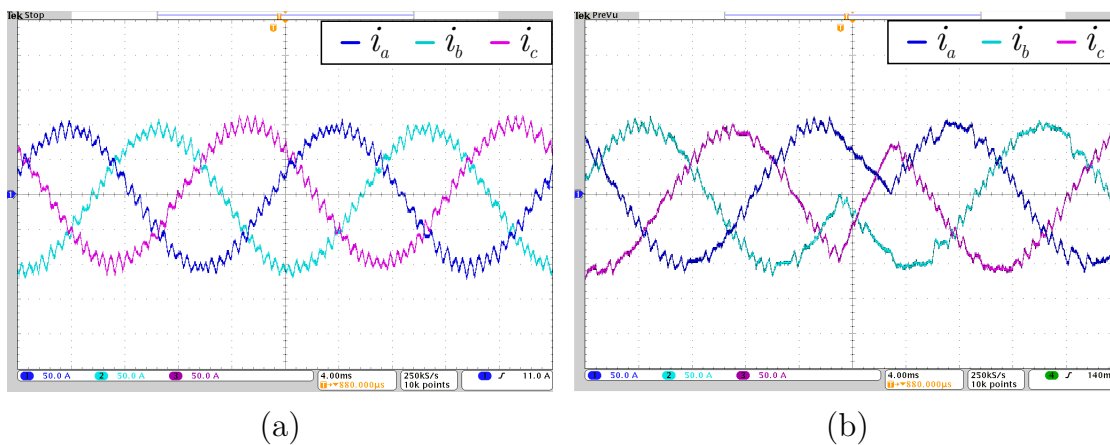
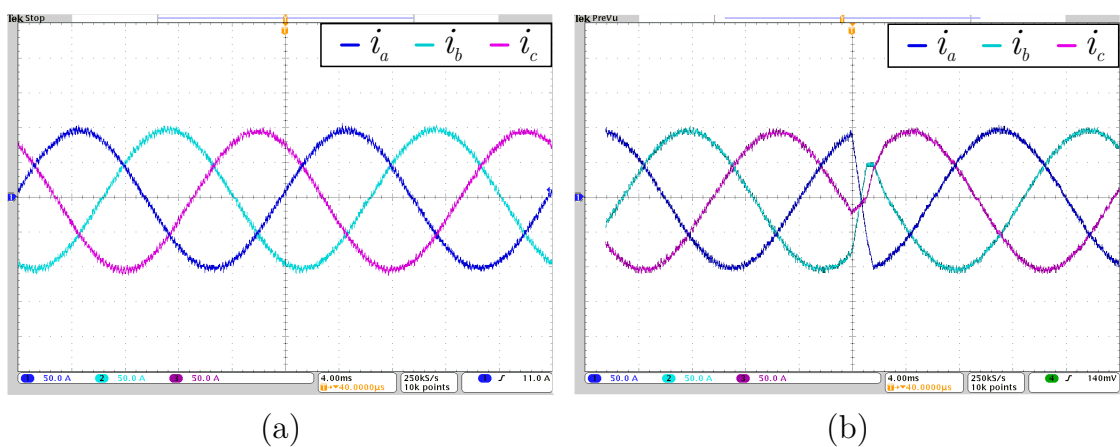
Nesta seção, os resultados experimentais serão apresentados, mostrando as formas de onda das correntes de saída bem como seus conteúdos espectrais e o tempo necessário para executar a rotina de controle. A montagem prática utilizou o Typhoon HIL402, que é um emulador de *hardware*, fazendo o papel da bancada prática, um DSP TEXAS TMS320F28335 e uma FPGA Xilinx[®] Spartan[®] 3E-500. Enquanto o Typhoon HIL402 emulava o conversor trifásico dois níveis, o DSP executava a rotina de controle, enviando os vetores de comutação para a FPGA; esta trabalhando como um modulador vetorial, responsável por gerar os sinais discretos de acionamentos para os interruptores semicondutores - ver Apêndice A.

Os parâmetros considerados para o conversor encontram-se na Tabela 2.1. Na sequência, as formas de onda das correntes de saída são mostradas através das Figuras 2.2 a 2.4. A duração da execução da rotina de controle para cada estratégia é apresentada na Tabela 2.2.

Como pode ser visto das Figuras 2.2 a 2.4, ambas as técnicas MPC rastreiam a corrente de referência satisfatoriamente e mostram boa resposta transitória. Pode ser

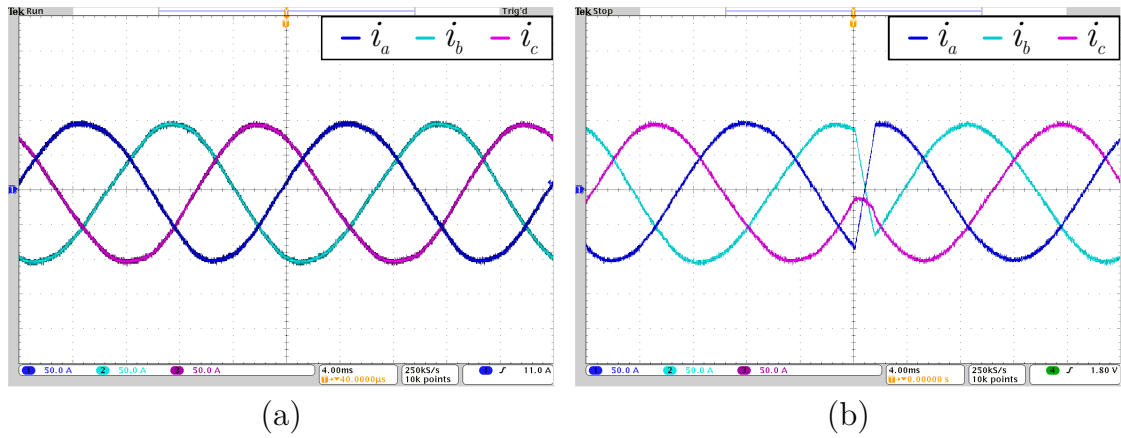
Tabela 2.1 – Parâmetros considerados para implementação prática das estratégias MPC apresentadas.

Parâmetro	Descrição	Valor
P	Potência Nominal	50 kW
V_{CC}	Tensão de Barramento	800 V
e_a, e_b, e_c	Tensões de rede	220 V (RMS)
R	Resistência do Filtro de Saída	0.5Ω
L	Indutância do Filtro de Saída	5 mH
T_s	Período de Amostragem	$100 \mu\text{s}$; $30 \mu\text{s}$
i_g^*	Corrente de Referência	100 A (pico)

Figura 2.2 – Estratégia 1, $T_s = 100 \mu\text{s}$: (a) Correntes de saída. (b) Resposta transitória para um degrau de fase de 180° .Figura 2.3 – Estratégia 1, $T_s = 30 \mu\text{s}$: (a) Correntes de Saída. (b) Resposta transitória para um degrau de fase de 180° .

Fonte: Autor.

Figura 2.4 – Estratégia 2, $T_s = 100 \mu\text{s}$: (a) Correntes de Saída. (b) Resposta transitória para um degrau de fase de 180° .



Fonte: Autor.

Tabela 2.2 – Tempo Computacional

Estratégia 1: FCS-OSV MPC	$22 \mu\text{s}$
Estratégia 2: FCS M ² PC	$62 \mu\text{s}$

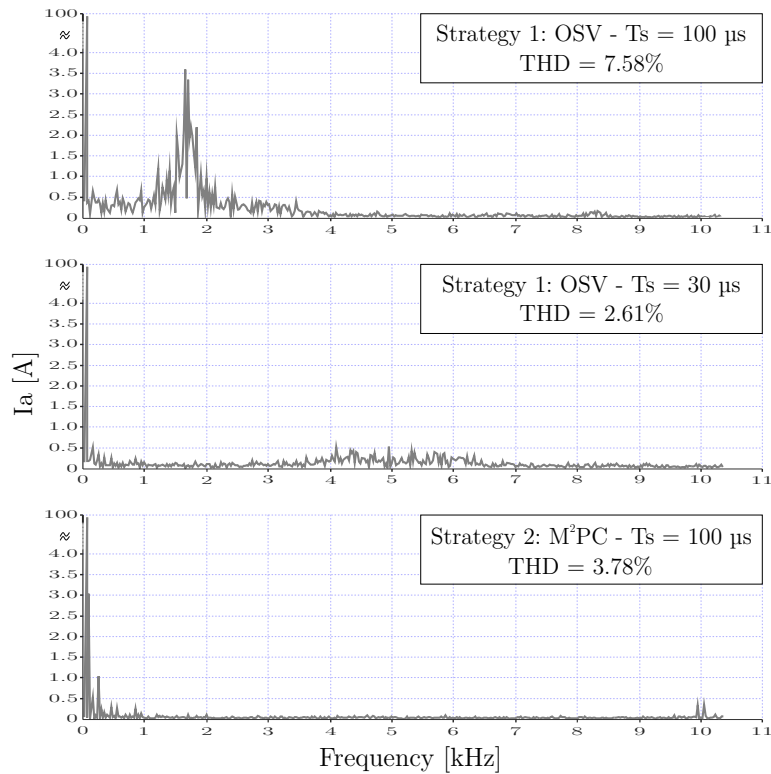
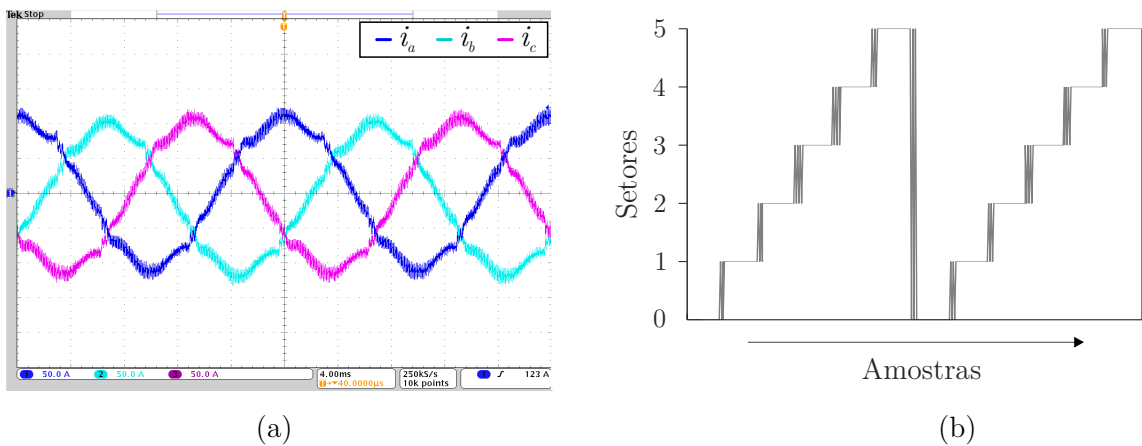
observado que, devido ao fato de que a estratégia 1: OSV-MPC aplicar um único vetor por período de amostragem, a ondulação de corrente se torna bastante pronunciada quando empregado um T_s relativamente grande, mesmo com um indutor de filtro significativo. No entanto, como o esforço computacional é menor, T_s pode ser reduzido, desde que não se comprometa as perdas de comutação.

A estratégia 2: FCS-M²PC, por outro lado, apresenta um esforço computacional muito maior. No entanto, como cinco vetores de comutação são aplicados (ver equação 2.5), a ondulação de corrente resultante se torna menor. Na Figura 2.5, uma comparação considerando o conteúdo harmônico para os três casos é apresentada.

Da Figura 2.5, pode ser observado que a técnica OSV-MPC apresenta um espectro espalhado, enquanto que a M²PC apresenta harmônicos concentrados em torno da frequência de amostragem. No entanto, existe a introdução de harmônicos de baixa ordem. Este fato pode ser observado na forma de onda da corrente de saída, especialmente quando o filtro de indutor é reduzido, por exemplo, para $L = 500 \mu\text{H}$.

Analisando a Figura 2.6 (b), pode ser observado que há uma oscilação dos setores adjacentes escolhidos próximo às fronteiras que os delimitam. Como consequência, é possível ver uma distorção na corrente de saída. Dessa maneira, estudos futuros mais aprofundados precisam ser realizados nessa área para clarificar e contornar essa limitação.

Figura 2.5 – Espectro harmônico para os três casos considerados

Figura 2.6 – (a) Correntes de linha para um indutor de saída $L = 500 \mu H$. (b) Oscilação na escolha dos setores na transição entre eles.

2.5 CONCLUSÃO

Este capítulo realiza uma revisão sobre a técnica MPC, da qual salienta-se a sub-classe FCS-MPC, sendo que essa aproveita a natureza discreta dos estados de comutação para a realização do controle. Além disso, uma comparação entre duas abordagens FCS-MPC é feita através de resultados experimentais.

Analisando as duas estratégias de controle preditivo para a corrente de saída de um inversor trifásico três níveis, verifica-se que a principal diferença entre as duas técnicas

consiste no número de vetores aplicados em um período de amostragem. Pode ser concluído que a técnica OSV-MPC exige um esforço computacional significativamente menor, permitindo reduzir o período T_s , desde que não se comprometa as perdas por comutação.

Por outro lado, a abordagem OSV-MPC apresenta um espectro espraiado enquanto a M²PC lembra um modulador convencional, com as harmônicos concentrados em torno da frequência de amostragem, resultando em menor ondulação de corrente para o mesmo período de amostragem T_s . Entretanto, a oscilação presente nos entornos de setores adjacentes permanece como uma questão a ser explorada em termos de pesquisa, investigando as suas causas bem como alternativas para contornar esse problema.

FCS-MPC tem demonstrado uma metodologia atrativa para o controle de conversores de potência, havendo a possibilidade de aproximá-lo da abordagem referente à modulação vetorial. No entanto, não se discute como os vetores são distribuídos em um período de amostragem e o conseqüente impacto na forma de onda na saída do conversor.

Por exemplo, a sequência de comutação (2.5) foi escolhida para se ter simetria na aplicação dos vetores. No entanto, não há um foco na maneira como esses vetores são escolhidos, não otimizando o número de comutações resultantes, por exemplo. Além disso, a estabilidade do MPC não foi completamente ainda demonstrada (GEYER; AGUILERA; QUEVEDO, 2013).

Em função do exposto, a proposta desta dissertação considera um sistema de controle convencional no qual o controlador e a modulação são desenvolvidos independentemente, sendo possível obter as vantagens das técnicas de controle já consolidadas. No entanto, o objeto de discussão desta dissertação é a modulação em si, a qual se baseia na metodologia *space vector* e na abordagem FCS-MPC para selecionar a sequência de comutação ótima.

3 MODULAÇÃO VETORIAL ÓTIMA PARA CONVERSOR MONOFÁSICO

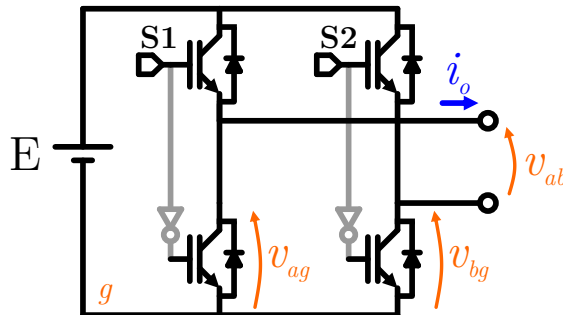
3.1 INTRODUÇÃO

Nesta seção, será desenvolvida a abordagem sugerida por essa dissertação, que se embasa no princípio MPC para formular uma modulação vetorial baseada em uma função custo. Deve-se salientar que o assunto da dissertação se limita à modulação, ficando o controlador a cargo de outro estágio, podendo esse empregar técnicas de controle já estabelecidas pela literatura, das quais já se tem estudos quanto ao quesito de estabilidade e robustez (OGATA, 2003). Por fim, são apresentados os resultados experimentais bem como a discussão dos mesmos, encerrando o capítulo com a conclusão do exposto.

3.2 INVERSOR PONTE COMPLETA

A fim de demonstrar a referida técnica, considere o inversor ponte completa apresentado na Figura 3.1.

Figura 3.1 – Inversor Monofásico Ponte Completa.



Fonte: Autor.

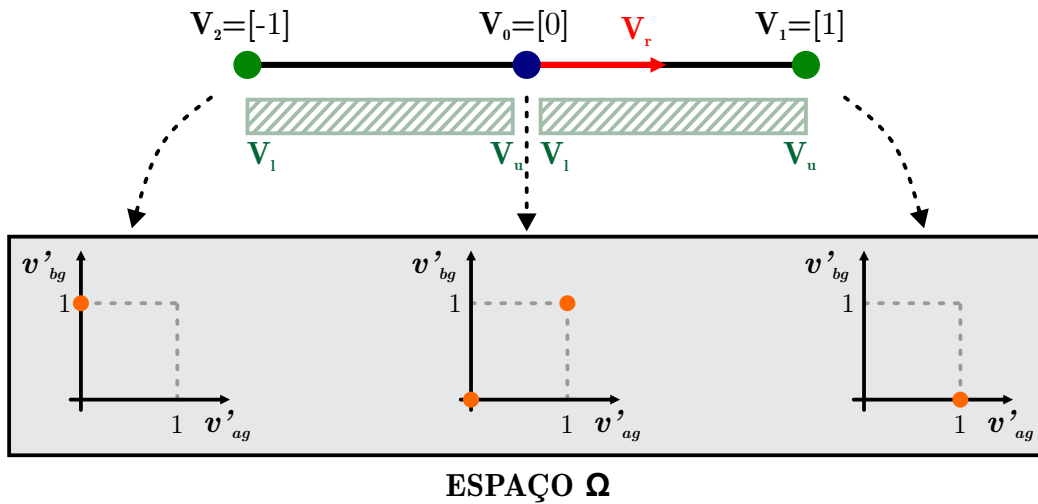
Este é um conversor muito simples e é adequado neste primeiro momento para expor a técnica de modulação proposta. Ele é constituído por dois braços contendo, em cada um, dois interruptores que comutam de forma complementar. O inversor ponte completa apresentado na Figura 3.1 é capaz de gerar apenas dois níveis em seus braços. Isto é, quando o interruptor de cima encontra-se fechado, tem-se a tensão de fase de valor E . Do contrário, tem-se a tensão de 0 V. Na sequência, a metodologia de modulação vetorial ótima é proposta.

3.3 ESTRATÉGIA DE MODULAÇÃO PROPOSTA

Nesta seção, será desenvolvida a metodologia de modulação ótima para o caso específico do conversor monofásico considerado. Tal conversor, por sua simplicidade, é indicado para a apresentação do estudo realizado e servirá de base para os capítulos subsequentes. Isto é, os próximos capítulos serão extensões da metodologia apresentada nesta seção.

O diagrama dos vetores de comutação do inversor monofásico, referentes às tensões de linha v_{ab} estão representados na Figura 3.2, onde Ω_x , $x \in \{0, 1, 2\}$ são os vetores de comutação (tensões possíveis de saída), \mathbf{V}_r , em vermelho, representa um vetor de referência a sintetizar e v'_{ag} e v'_{bg} são as tensões de fase. Em todos os casos, os vetores são normalizados em relação ao barramento CC.

Figura 3.2 – Diagrama vetorial para o Inversor Monofásico Ponte Completa.



Fonte: Autor.

É necessário observar que cada vetor de linha possui um ou mais vetores correspondentes no espaço das tensões de braço v'_{ag} e v'_{bg} , os quais são reunidos dentro conjunto Ω . Observa-se que, no caso do vetor nulo, há dois vetores correspondentes em Ω . Esses vetores estão sumarizados no mapeamento (3.1), onde cada coluna das matrizes representa um vetor $[v'_{ag} \ v'_{bg}]^T$.

$$\begin{aligned}
 \mathbf{v}_1 = [1] & \rightarrow \Omega_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
 \mathbf{v}_2 = [-1] & \rightarrow \Omega_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
 \mathbf{v}_0 = [0] & \rightarrow \Omega_0 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}
 \end{aligned} \tag{3.1}$$

É possível observar dois vetores redundantes no espaço Ω_0 para o vetor \mathbf{v}_0 . Como os dois sintetizam a mesma tensão de linha de saída, necessita-se de um critério adicional retratado pela função custo (3.8) para determinar a escolha entre os dois vetores. Normalmente, essa escolha é feita visando o menor número de comutações de acordo com a sequência de comutação.

Após mapear \mathbf{v}_x em Ω_x , $x \in \{0, 1, 2\}$ é necessário mapear Ω_x em \mathbf{s}_x , $x \in \{0, 1, 2\}$, onde os últimos denotam os vetores no espaço dos interruptores. No caso do conversor ponte completa, o tamanho do vetor \mathbf{s}_x é igual ao Ω_x , uma vez que não há redundâncias entre esses dois espaços. Dessa maneira, pode-se obter a equação (3.2).

$$\begin{aligned} \Omega_1 &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \rightarrow & \mathbf{s}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ \Omega_2 &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} & \rightarrow & \mathbf{s}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \Omega_0 &= \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} & \rightarrow & \mathbf{s}_0 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \end{aligned} \quad (3.2)$$

Com o objetivo de melhor organizar a apresentação da modulação, esta seção será subdividida em tópicos referentes à detecção dos dois vetores mais próximo, à escolha da sequência de comutação e ao mapeamento das tensões de linha no espaço das tensões de fase e, posteriormente, no espaço dos interruptores.

3.3.1 Detecção dos dois vetores mais próximos

Dado um vetor de referência \mathbf{v}_r qualquer, são necessárias estratégias computacionais para selecionar os dois vetores mais próximos, calcular as razões cíclicas correspondentes, bem como para empregar as relações (3.1) e (3.2). A estratégia utilizada é a mesma empregada em (DE CASTRO; CORREA; JACOBINA, 2013), consistindo em arredondamentos através das funções $\text{floor}(\cdot)$ e $\text{ceil}(\cdot)$ presentes na linguagem C, onde em ambos se tem como entrada um número real, sendo que a primeira arredonda para o inteiro imediatamente inferior ao argumento e, a segunda, para o imediatamente superior. Dessa maneira, pode-se definir as expressões presentes na equação (3.3).

$$\begin{aligned} \mathbf{v}_u &= \text{ceil}(\mathbf{V}_r) \\ \mathbf{v}_l &= \text{floor}(\mathbf{V}_r) \end{aligned} \quad (3.3)$$

Onde \mathbf{v}_u representa o vetor mais à direita de \mathbf{v}_r e \mathbf{v}_l , o mais à esquerda. Assim, se \mathbf{v}_r estiver à direita de \mathbf{v}_0 , $\mathbf{v}_u = [1]$ e $\mathbf{v}_l = [0]$. Caso contrário, $\mathbf{v}_u = [0]$ e $\mathbf{v}_l = [-1]$, conforme indicado na Figura 3.2 pelas regiões hachuradas.

Após essa etapa, é necessário realizar o cálculo das razões cíclicas de cada um dos

vetores. É fácil deduzir que essas razões podem ser escritas como no conjunto de equações (3.4).

$$\begin{aligned} d_u &= \|\mathbf{v}_r - \mathbf{v}_l\| \\ d_l &= 1 - d_u \end{aligned} \quad (3.4)$$

Onde d_u representa a razão cíclica associada ao vetor \mathbf{v}_u e d_l , ao vetor \mathbf{v}_l , sendo $\|\cdot\|$ a norma euclidiana do argumento.

3.3.2 Escolha da Sequência de Comutação

A fim de implementar a modulação proposta, primeiramente se deve definir uma sequência de comutação. Essa pode ser dada por:

$$\mathbf{s} = \mathbf{v}_u \mathbf{v}_l \mathbf{v}_u \quad (3.5)$$

Sendo que cada vetor, \mathbf{v}_u ou \mathbf{v}_l , pode conter vetores redundantes no espaço das tensões de fase e estes, por sua vez, podem conter vetores redundantes no espaço dos interruptores (o que não acontece no caso deste conversor monofásico).

Essa sequência de comutação foi escolhida pelo fato de que, independentemente dos dois setores onde a referência se encontra, não haverá uma troca abrupta entre os vetores [1] e [-1] da Figura 3.2. Isto é, os vetores ativos estarão sempre intermediados pelo vetor nulo, ficando a função custo, para esse último, responsável por selecionar o vetor redundante ótimo no espaço dos interruptores.

3.3.3 Mapeamento no espaço das tensões de fase e no dos interruptores

A fim de realizar essa seleção, é necessário, antes então, mapear o espaço dos vetores de comutação de linha no das tensões de fase e, posteriormente, no dos interruptores. Por se tratar de um conversor relativamente simples, esse mapeamento pode ser realizado diretamente das tensões de linha para os interruptores conforme mostrado em (3.6).

$$\begin{aligned} \mathbf{s}_u &= \left[\begin{array}{c} \text{floor} \left(\mathbf{M} \left[\mathbf{v}_u \quad 0,5 \right]^T \right) \\ \text{ceil} \left(\mathbf{M} \left[\mathbf{v}_u \quad 0,5 \right]^T \right) \end{array} \right]^T \\ \mathbf{s}_l &= \left[\begin{array}{c} \text{floor} \left(\mathbf{M} \left[\mathbf{V}_l \quad 0,5 \right]^T \right) \\ \text{ceil} \left(\mathbf{M} \left[\mathbf{V}_l \quad 0,5 \right]^T \right) \end{array} \right]^T \end{aligned} \quad (3.6)$$

Onde a matriz auxiliar \mathbf{M} pode ser dada por:

$$\mathbf{M} = \begin{bmatrix} 0,5 & 1 \\ -0,5 & 1 \end{bmatrix} \quad (3.7)$$

A Tabela 3.1 mostra um resumo do mapeamento dos vetores \mathbf{v}_u e \mathbf{v}_l nos consequentes vetores \mathbf{s}_u e \mathbf{s}_l .

Tabela 3.1 – Mapeamento dos vetores no espaço das tensões de linha para o espaço dos interruptores

v_l	v_u	s_l	s_u
[0]	[1]	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$
[-1]	[0]	$\begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$

3.3.4 Função Custo

Para elaborar o algoritmo de implementação, para cada vetor da sequência (3.5), a função custo (3.8) é avaliada sob os vetores nos espaços dos interruptores correspondentes. Sob a lógica MPC, essa pode ser definida como:

$$g = \|(\mathbf{s}_L - \mathbf{s}_i)\|_1, \quad i \in 0..1 \quad (3.8)$$

Onde \mathbf{s}_L é o vetor no espaço dos interruptores imediatamente anterior dentro da sequência de comutação aplicado ao conversor, \mathbf{s}_i é o i -ésimo vetor da sequência sob teste e $\|\cdot\|_1$ é a norma um do argumento, definida como:

$$\|\mathbf{x}\|_1 = \sum_{i=1}^N |\mathbf{x}_i| \quad (3.9)$$

Onde N é a dimensão do vetor e \mathbf{x}_i é o i -ésimo elemento do vetor \mathbf{x} qualquer. Dessa maneira, o algoritmo percorre os vetores da sequência de comutação e, para cada um, a função custo (3.8) é calculada. Posteriormente, o vetor \mathbf{s}_i que retornou o menor custo é selecionado para compor a sequência de comutação, tornando-se, então, o vetor \mathbf{s}_L para efeito de comparação para o próximo vetor da sequência. Em termos práticos, essa sequência, com as devidas razões cíclicas, é enviada para uma FPGA, a qual desempenha o papel de um modulador vetorial, aplicando os vetores ao conversor.

3.4 FLUXOGRAMA

Na Figura 3.3, tem-se um fluxograma com os passos de seleção dos vetores realizados pelo algoritmo. Para fins de clareza, cada bloco denotado no fluxograma pelos símbolos de chave serão explicados.

BLOCO 1: A rotina, a qual começa a ser executada a cada instante de amostragem, recebe um vetor de tensão de linha de referência \mathbf{v}_r normalizado e o último vetor aplicado, \mathbf{s}_L , no espaço dos interruptores. Esse vetor está na cor cinza para enfatizar que ele não é uma variável informada pelo usuário, mas sim uma variável interna do próprio algoritmo, resultante da execução do intervalo de amostragem anterior. Para que seja possível executar o fluxograma pela primeira vez, \mathbf{s}_L é definido como sendo o vetor nulo, isto é: $\mathbf{s}_L = [0 \ 0]^T$, na rotina principal.

É importante observar que \mathbf{v}_r , o vetor proveniente do controlador, pode ser tal que esteja além dos recursos do conversor. Isto acontece quando se solicita uma tensão maior que a máxima possível de sintetizar, implicando em $\|\mathbf{v}_r\| > 1$. Dessa forma, $\|\mathbf{v}_r\|$ é saturada no valor um.

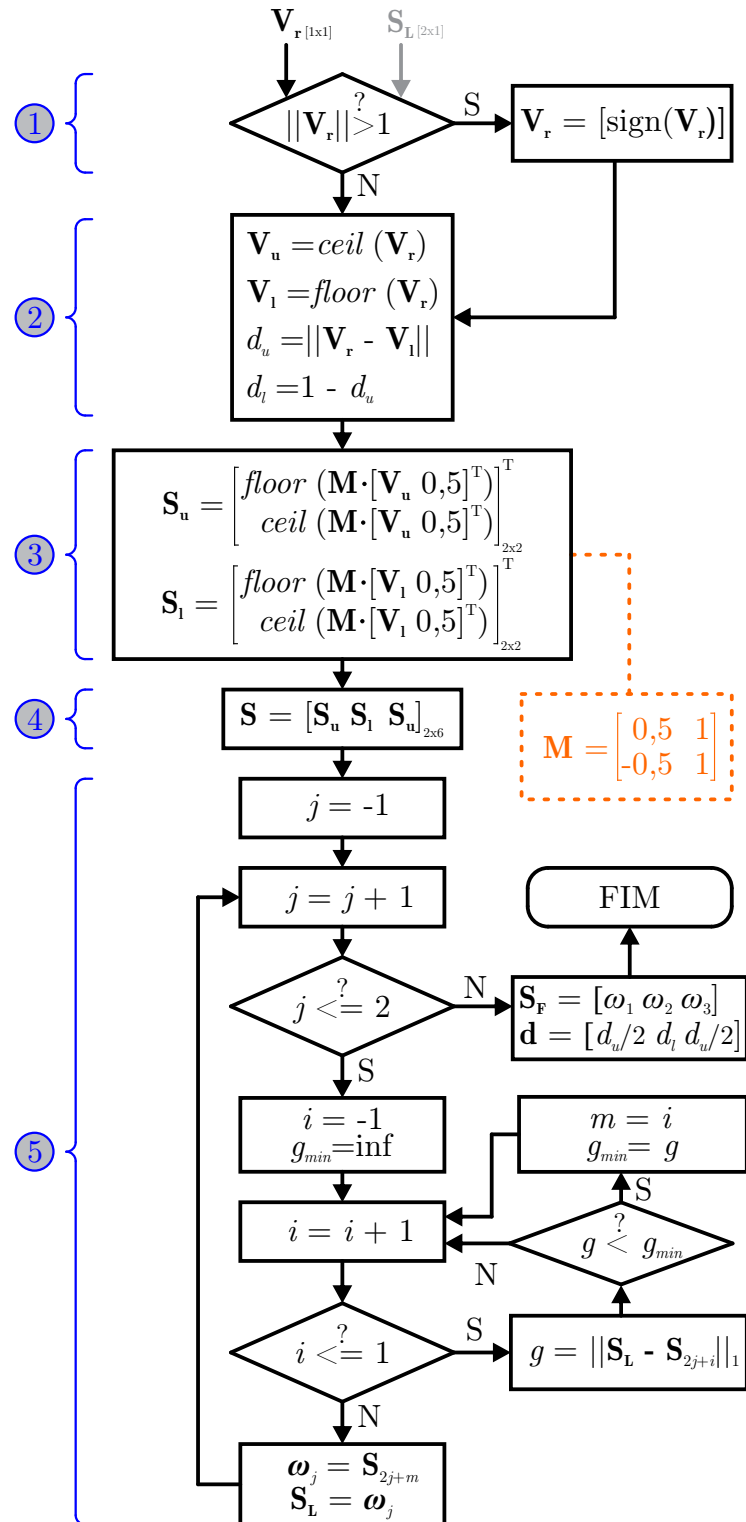
BLOCO 2: A partir de \mathbf{v}_r , que está na faixa de $[0]$ a $[1]$, os vetores \mathbf{v}_u e \mathbf{v}_l , indicados na relação (3.3) e representados na Figura 3.2, são encontrados através dos truncamentos realizados através das funções $\text{floor}(\cdot)$ e $\text{ceil}(\cdot)$ da linguagem C. Como consequência, esses vetores recebem exclusivamente o valor de $[0]$ ou $[1]$. Na sequência, as razões cíclicas podem ser calculadas através da norma Euclidiana do vetor \mathbf{v}_r , conforme relação (3.4).

BLOCO 3: Uma vez com os vetores \mathbf{v}_u e \mathbf{v}_l encontrados, é necessário mapeá-los no espaço correspondente dos interruptores, denotando o último por \mathbf{s}_u e \mathbf{s}_l , respectivamente. Para fazer isso, recorre-se à artifício de multiplicar a matriz auxiliar \mathbf{M} (equação 3.7) por um vetor contendo aquele a ser mapeado e truncar cada elemento da matriz resultante para o número imediatamente superior e inferior, da forma como é mostrada no fluxograma e no conjunto de equações (3.6).

Como consequência, para cada vetor \mathbf{v}_u e \mathbf{v}_l , se tem dois vetores coluna no espaço dos interruptores. Para o vetor de tensão de linha $[0 \ 0]^T$, os vetores em \mathbf{s}_u e \mathbf{s}_l serão distintos. Em outras palavras, haverá dois vetores redundantes. Para os demais vetores, haverá colunas iguais, indicando que não há vetores redundantes, como mostra a Tabela 3.1, que explicita o mapeamento entre os dois espaços.

BLOCO 4: Uma sequência de comutação é definida, indicando a ordem com que os vetores \mathbf{v}_u e \mathbf{v}_l aparecem. Essa sequência impacta diretamente no resultado final, sendo que a sua escolha é passível de análise criteriosa. Como, no momento, o principal objetivo é formular a metodologia de modulação em si, considerou-se essa sequência, que é tipicamente empregada na modulação vetorial. A formulação do impacto da sequência de comutação se destina, portanto, a um trabalho futuro. Como cada vetor \mathbf{s}_u e \mathbf{s}_l tem dois vetores colunas, da maneira como a sequência foi definida, a dimensão final da matriz

Figura 3.3 – Fluxograma dos passos realizados pelo algoritmo de seleção dos vetores redundantes a serem aplicados ao conversor.



Fonte: Autor.

da sequência de comutação \mathbf{s} é 2×6 .

BLOCO 5: Essa é a parte do algoritmo onde seleciona-se, de cada conjunto \mathbf{s}_u e

\mathbf{s}_1 da sequência \mathbf{s} um vetor a ser efetivamente aplicado ao conversor. Para tanto, uma função custo é aplicada para cada vetor da sequência $\mathbf{s} = \mathbf{s}_u \mathbf{s}_l \mathbf{S}_u$, selecionando-se aquele que resulta no menor custo para cada conjunto \mathbf{s}_u e \mathbf{s}_l . Como consequência, os vetores $\omega_1 \omega_2 \omega_3$ que compõem a sequência final \mathbf{s}_F a ser aplicada ao conversor correspondem a \mathbf{s}_u , \mathbf{s}_l e \mathbf{S}_u , respectivamente.

A função custo pode ser qualquer uma que atenda aos objetivos do projeto. Neste caso, foi considerada a norma um da diferença entre o vetor candidato e o último aplicado. Em outras palavras, é selecionado aquele vetor que apresentará o menor número de comutações em relação ao último. Esse critério é vantajoso no sentido de diminuir o número de comutações dos interruptores, reduzindo as perdas de comutação, o aquecimento do conversor e aumentando a vida útil dos componentes. No entanto, especialmente em conversores mais complexos, onde se tenham tensões internas de capacitores a balancear, parâmetros adicionais podem ou devem ser adicionados à função custo.

Por fim, é necessário definir, para cada vetor ω_x , $x \in \{1, 2, 3\}$, a razão cíclica correspondente. No BLOCO 2 do algoritmo, as razões totais já foram calculadas. Isto é, \mathbf{v}_u deve compor a sequência de comutação por um tempo referente à d_u ; e \mathbf{v}_l , à d_l . Como a modulação vetorial apresenta como grau de liberdade a escolha da sequência de comutação, se faz necessário, apenas, dividir as razões cíclicas entre os vetores de modo a respeitar a parcela de tempo total que cada vetor deve aparecer no período de amostragem. Dessa maneira, pode-se definir um vetor \mathbf{d} contendo as respectivas razões cíclicas para cada vetor da sequência de comutação na ordem com que aparecem, resultando em $\mathbf{d} = [d_u/2 \ d_l \ d_u/2]$.

Implementando a técnica de modulação proposta de acordo com o fluxograma da Figura 3.3 e com as descrições expostas dos respectivos blocos do algoritmo, pode-se chegar ao seguinte código, considerando a linguagem C, largamente utilizada pelos DSPs.

```

1 float Vr, du, dl, gmin, g;
2 int i, j, m, Vu, Vl, SL[2], Su[2][2], Sl[2][2], S[2][6],
3   SF[2][3];
4
5 //Testa se a norma do vetor de referência é maior que um
6 if(abs(Vr)>1)
7   Vr = sign(Vr);
8
9 //Encontra os vetores Vu, Vl além das razões cíclicas du e dl
10 Vu = ceil(Vr);
11 Vl = floor(Vr);
12 du = abs(Vr-Vl);
13 dl = 1-du;
14

```

```
15 //Mapeia Vu e Vl em Su e Sl
16 Su[0][0] = floor( 0.5*(1+Vu) );
17 Su[1][0] = floor( 0.5*(1-Vu) );
18 Su[0][1] = ceil ( 0.5*(1+Vu) );
19 Su[1][1] = ceil ( 0.5*(1-Vu) );
20
21 Sl[0][0] = floor( 0.5*(1+Vl) );
22 Sl[1][0] = floor( 0.5*(1-Vl) );
23 Sl[0][1] = ceil ( 0.5*(1+Vl) );
24 Sl[1][1] = ceil ( 0.5*(1-Vl) );
25
26 //Monta a disposição dos vetores na sequência de comutação
27 S[0][0] = Su[0][0]; S[1][0] = Su[1][0];
28 S[0][1] = Su[0][1]; S[1][1] = Su[1][1];
29 S[0][2] = Sl[0][0]; S[1][2] = Sl[1][0];
30 S[0][3] = Sl[0][1]; S[1][3] = Sl[1][1];
31 S[0][4] = Su[0][0]; S[1][4] = Su[1][0];
32 S[0][5] = Su[0][1]; S[1][5] = Su[1][1];
33
34 //Seleciona os vetores baseando-se na função custo
35 for (j=0; j<=2; j++){
36     gmin=10000000;
37     for (i=0; i<=1; i++){
38         g=abs(SL[0]-S[0][2*j+i])+abs(SL[1]-S[1][2*j+i]);
39         if(g<gmin){
40             gmin = g;
41             m = i;
42         }
43     }
44
45     //Monta a sequência de comutação a ser aplicada ao conversor
46     SF[0][j] = S[0][2*j+m]; SF[1][j] = S[1][2*j+m];
47     SL[0]=SF[0][j]; SL[1]=SF[1][j];
48 }
49
50 //Envia a sequência para o modulador vetorial (FPGA)
51 EsvmRegs.DATA.DWT.DW0 = EsvmDwPerc(du/2.0*100.0, 0);
52 EsvmRegs.DATA.DWT.DW1 = EsvmDwPerc(dl*100.0, 1);
53
```

```

54 EsvmRegs.DATA.VEC.W0 = EsvmSequence(SF[0][0]*10+SF[1][0]);
55 EsvmRegs.DATA.VEC.W1 = EsvmSequence(SF[0][1]*10+SF[1][1]);
56 EsvmRegs.DATA.VEC.W2 = EsvmSequence(SF[0][2]*10+SF[1][2]);

```

Cabe realizar algumas observações a respeito do código apresentado. Pode-se notar que as variáveis ω_1 , ω_2 e ω_3 que aparecem no fluxograma da Figura 3.3 não são necessárias para a elaboração do algoritmo, armazenando os vetores diretamente na variável \mathbf{s}_F à medida que se vai percorrendo o laço *for* de índice j . Observa-se, também, que a matriz \mathbf{M} não aparece explicitamente no código. Ao invés disso, a multiplicação já aparece simplificada, nas linhas de número 16 a 24.

Por se tratar de um algoritmo executado em períodos de amostragem relativamente pequenos - na ordem de microsegundos -, se faz necessário um código eficiente, da maneira mais otimizada possível a fim de evitar processamento desnecessário. Assim, procura-se empregar cálculos simplificados e que exijam o menor processamento possível sem, no entanto, pecar na clareza do código.

Além disso, pode-se verificar que o bloco de código compreendido entre as linhas 27 e 32, assim como o que está entre as linhas 16 e 24, poderia ser escrito de maneira mais compacta através da utilização de laço *for*. Neste ponto, cabe uma ponderação interessante. Se esses blocos fossem escritos da maneira com que foi conjecturada, tornaria-se o código mais conciso, especialmente para o caso de um conversor mais complexo. No entanto, o tempo de processamento do algoritmo seria maior, pois o processador necessitaria realizar um laço de repetição e, possivelmente, operações adicionais introduzidas pelo programador para poder compactar as linhas de código em um bloco menor.

Desse ponto de vista, quanto mais explícitas as linhas de código forem, melhor será o desempenho do dispositivo de processamento. Por outro lado, ao adotar essa prática, o tamanho do algoritmo fica maior, podendo impactar significativamente na memória de programa, havendo a possibilidade de não conseguir carregar o código para o DSP. Dessa forma, o programador deve encontrar um equilíbrio entre desempenho e tamanho do algoritmo.

Por fim, salienta-se as linhas de código de número 51 a 56. Essas são estruturas de dados criadas como interface para envio da sequência de comutação para o modulador vetorial - FPGA (ver Apêndice A). As duas primeiras linhas referem-se às razões cíclicas respectivas a cada vetor da sequência. O primeiro campo, *EsvmRegs*, referencia os registradores do modulador. O segundo, *DATA* informa que se trata de um registrador de dado e não de controle. O terceiro, *DWT*, informa que esse dado é de *dwell time*, ou seja, de razão cíclica. O último, *DW0* e *DW1*, informam a qual vetor o valor de razão cíclica se refere.

Os valores de razão cíclica, por sua vez, são informados através da função *EsvmDwPerc*, para a qual se tem maiores detalhes no Apêndice A. Como entrada dessa função, se tem o valor de razão cíclica informado em formato percentual, acompanhado do segundo

parâmetro que representa o índice do vetor correspondente da sequência. Ainda se observa que apenas duas razões cíclicas são informadas. Isso porque não há a necessidade de informar ao modulador a terceira, uma vez que o somatório delas deve ser igual a um. Como o modulador é pré-configurado na função principal - função *main* -, já são esperados três vetores e, portanto, a terceira razão cíclica é consequência das duas primeiras.

Já os vetores em si - contendo o estado dos interruptores do conversor - são informados utilizando-se a instância VEC, precedida daquela referente ao vetor desejado, W0, W1 e W2. Esses vetores são informados por meio da função *EsvmSequence*, para a qual se tem maiores detalhes no Apêndice A, tendo como parâmetro um número com formato binário representando o estado de cada interruptor. Dessa forma, se, por exemplo, for escrito:

```
1 EsvmSequence(10)
```

O interruptor S1 da Figura 3.1 assumirá valor lógico 1 (fechado); e S2, 0 (aberto). No entanto, apesar de ter o formato de um número binário, a função o considera como um inteiro. Por isso, o valor do primeiro interruptor é multiplicado por 10 e somado ao do segundo.

3.5 RESULTADOS EXPERIMENTAIS: FUNÇÃO CUSTO 1

Nesta seção, serão mostrados os resultados experimentais obtidos considerando-se o inversor da Figura 3.1 e uma função custo que utiliza a norma um da diferença entre o último vetor e o candidato sob teste, tal como apresentada no fluxograma da Figura 3.3.

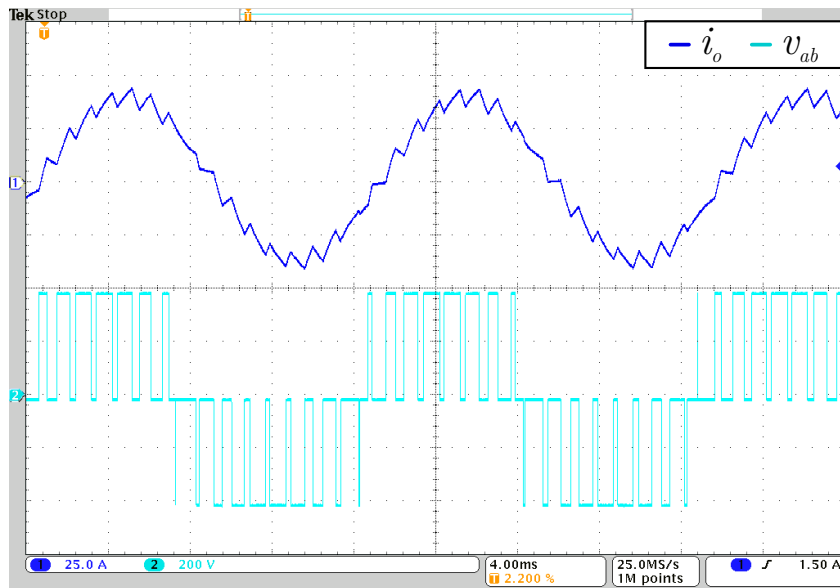
Para fins de experimentação, considera-se o DSP TEXAS TMS320F28335 em conjunto com a FPGA Xilinx Spartan 3E-500, tendo o Typhoon HIL402 como emulador do conversor, conforme Apêndice A. Para a simulação, é considerada, na saída do conversor da Figura 4.1, uma carga trifásica RL, consumindo uma potência de 5 kW, com $FP=0,92$, barramento CC de 400 V e referência senoidal com índice de modulação igual à $m_i = 0,8$. As formas de onda de saída encontram-se conforme as Figuras 3.4 e 3.5 para uma frequência de amostragem de 1 kHz.

3.5.1 Discussão dos Resultados

A Figura 3.4 mostra, na parte de cima, a forma de onda da corrente de saída para a carga RL considerada. Vê-se que a modulação cumpre o papel de gerar uma tensão de saída média senoidal, observando que o valor médio da tensão de saída encontra-se descolado da forma de onda de corrente em função do fator de potência.

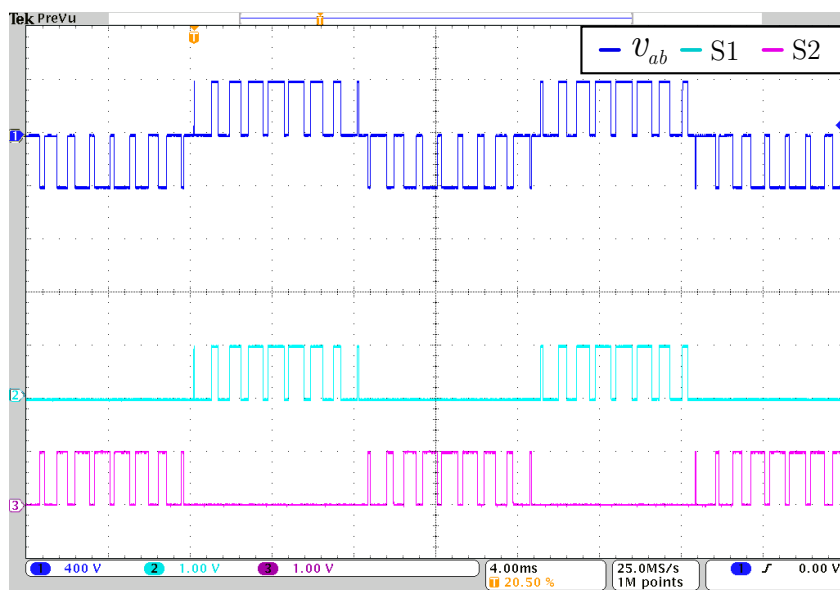
A modulação proposta gerou um padrão PWM de saída bem definido e simétrico

Figura 3.4 – Corrente de saída e tensão de linha considerando a sequência de comutação (3.5) e a função custo (3.8).



Fonte: Autor.

Figura 3.5 – Tensão de linha de saída e a distribuição dos pulsos sobre interruptores.



Fonte: Autor.

em relação à um quarto da onda, o que é de se esperar pela definição também simétrica da sequência de comutação (3.5).

Já a Figura 3.5 mostra os pulsos nos interruptores superiores do conversor. É possível observar que, como a função custo (3.8) favorece o menor número de comutações, percebe-se que os interruptores não comutam por metade de um ciclo. Dessa maneira, deseja-se uma melhor distribuição dos pulsos nos interruptores, culminando numa função

custo incrementada como mostra o tópico seguinte.

3.6 RESULTADOS EXPERIMENTAIS: FUNÇÃO CUSTO 2

Como se pôde perceber do resultado do padrão PWM da Figura 3.5, utilizando-se como função custo apenas a norma um da diferença entre o último vetor e o candidato sob teste, os vetores foram escolhidos de modo que se tivesse a menor mudança no estado dos interruptores. Porém, ocorreu que os interruptores ficaram sem comutar por meio ciclo. Dessa maneira, a fim de se ter uma distribuição mais homogênea dos pulsos entre os interruptores, cogita-se uma função custo que leve em conta esse critério. Dessa forma, a função custo da seção anterior é modificada, conforme Equação 3.10.

$$g_i = \|\mathbf{s}_L - \mathbf{s}_{2j+i}\|_1 + k \sum_{n=0}^1 |\text{floor}(|\mathbf{s}_{L_n} - \mathbf{s}_{n,2j+i}| - 0,5)| (\mathbf{LSC}_n + 1) \quad (3.10)$$

Onde os termos \mathbf{s}_{L_n} e $\mathbf{s}_{n,2j+i}$ denotam, respectivamente, o n -ésimo elemento do vetor coluna \mathbf{s}_L e o n -ésimo elemento do vetor da coluna $2j + i$. Em outras palavras, essa é uma função que manipula elemento por elemento de cada vetor no espaço dos interruptores. Cabe observar que n varia de 0 a 1, uma vez que se tem apenas dois interruptores com acionamento independente no caso do conversor ponte completa. Ainda nessa mesma linha, o termo \mathbf{LSC}_n denota o n -ésimo elemento do vetor auxiliar \mathbf{LSC} , o qual estende sua nomenclatura para *Last Switching Counting*. Para fins de clareza, essa função custo é datalhada a seguir.

A fim de evitar que um interruptor permaneça sem comutar por um tempo significativo, para cada troca de vetor ω da sequência de comutação, é feita uma contagem de quantas vezes o estado de um interruptor não é alterado. Isto é, para cada um dos dois interruptores envolvidos na troca dos vetores ω , verifica-se se houve comutação. Em caso negativo, incrementa-se o elemento correspondente do vetor auxiliar \mathbf{LSC} . Ou seja, \mathbf{LSC} é um vetor coluna de dimensão 2×1 , onde o primeiro termo refere-se a quantas vezes S1 não comutou consecutivamente; e, da mesma forma, o segundo elemento, à S2. No entanto, se houver comutação, o elemento correspondente de \mathbf{LSC} é zerado. É importante observar que essa função é realizada para cada termo do somatório da equação (3.10). No entanto, o valor que entra na função custo é aquele referente à soma de todos os termos de \mathbf{LSC} .

Na Figura 3.6, um exemplo é realizado para demonstrar a lógica do segundo termo da função custo apresentada na equação (3.10), considerando uma sequência qualquer, sem se prender ao seu significado físico, ou seja, apenas para explicitar a lógica em questão.

Uma vez selecionado o vetor da sequência, o vetor auxiliar \mathbf{LSC} é redefinido, de

Figura 3.6 – Demonstração da lógica do segundo termo da função custo da equação (3.10).

$$\begin{array}{cccccccccccc}
 & \omega_1 & \omega_2 & \omega_3 & \omega_1 & \omega_2 & \omega_3 & \omega_1 & \omega_2 & \omega_3 & \omega_1 & \omega_2 & \omega_3 \\
 \mathbf{S} = & \left\{ \begin{array}{cccccccccccc}
 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\
 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0
 \end{array} \right\} \begin{array}{l} \rightarrow \mathbf{S1} \\ \rightarrow \mathbf{S2} \end{array} \\
 & & & & & & & & & & & & \downarrow \\
 \mathbf{LSC} = & \left\{ \begin{array}{cccccccccccc}
 0 & 1 & 2 & 3 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 0 & 0 & 0
 \end{array} \right\} \\
 & & & & & & & & & & & & \downarrow \\
 \sum_{n=0}^1 |\text{floor}(|\mathbf{SL}_n - \mathbf{S}_{n,2j+i}| - 0,5)| (\mathbf{LSC}_n + 1) = & 0 & 1 & 2 & 3 & 0 & 1 & 1 & 2 & 3 & 0 & 0 & 0
 \end{array}$$

Fonte: Autor.

modo a guardar a informação do número de comutações consecutivas envolvidas efetivamente no último vetor aplicado. Após, então, esse vetor é utilizado na seleção do próximo vetor de comutação e assim sucessivamente. Observa-se, ainda, que o segundo termo da equação (3.10) é ponderado por um fator k . Esse parâmetro serve para ajustar o impacto de cada termo na função custo como um todo. Tal procedimento fica melhor explicitado quando observado o fluxograma da Figura 3.7, onde, para fins de simplicidade, se detém apenas na seção onde se realiza a seleção dos vetores de comutação por meio da função custo (3.10). Isto é, se realiza a reformulação do Bloco 5 do fluxograma da Figura 3.3.

Em termos de código de programação, o fluxograma da Figura 3.7 pode ser escrito como:

```

1 //Seleciona os vetores baseando-se na função custo 2
2 for (j=0; j<=2; j++){
3   gmin = 10000000;
4   for (i=0; i<=1; i++){
5     LSC_temp[0] = abs(floor((abs(SL[0]-S[0][2*j+i])-0.5))) *
6                 (LSC[0] + 1);
7     LSC_temp[1] = abs(floor((abs(SL[1]-S[1][2*j+i])-0.5))) *
8                 (LSC[1] + 1);
9     g=abs(SL[0]-S[0][2*j+i])+abs(SL[1]-S[1][2*j+i])+
10        k*(LSC_temp[0]+LSC_temp[1]);
11    if(g<gmin){
12      gmin = g;
13      m = i;
14      LSC_temp[0];
15      LSC[1]=LSC_temp[1];
16    }
17  }
18

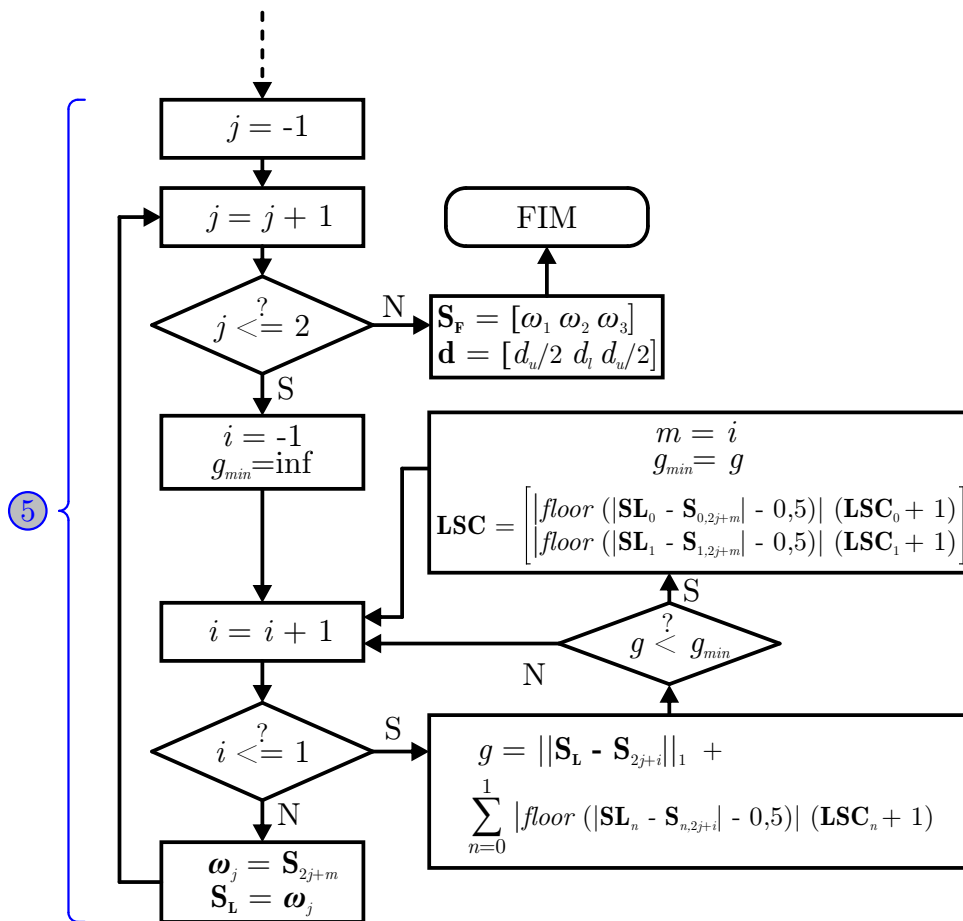
```

```

19 LSC[0]=abs(floor((abs(SL[0]-S[0][2*j+m])-0.5)) *
20 (LSC[0] + 1));
21 LSC[1]=abs(floor((abs(SL[1]-S[1][2*j+m])-0.5)) *
22 (LSC[1] + 1));
23
24 //Monta a sequência de comutação a ser aplicada ao conversor
25 SF[0][j] = S[0][2*j+m]; SF[1][j] = S[1][2*j+m];
26 SL[0]=SF[0][j]; SL[1]=SF[1][j];
27 }

```

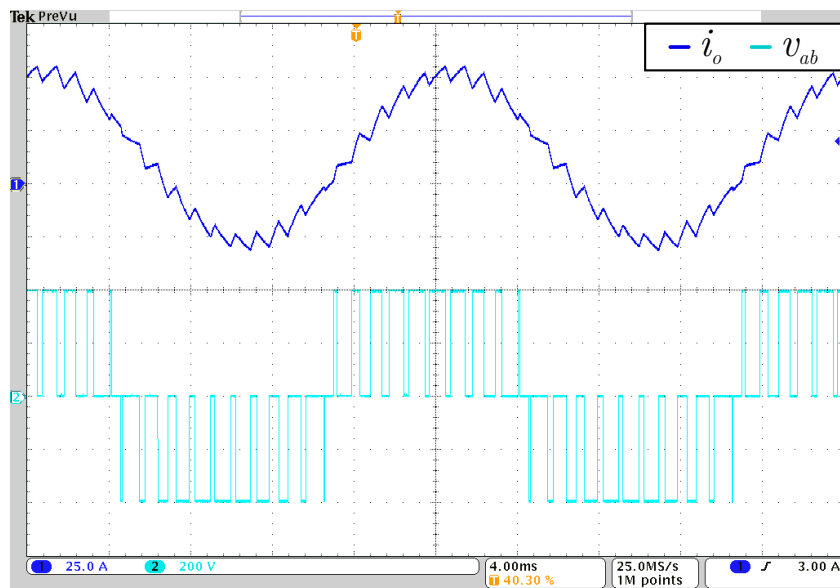
Figura 3.7 – Seção 5 do fluxograma da Figura 3.3 reformulada para contemplar a função custo (3.10).



Fonte: Autor.

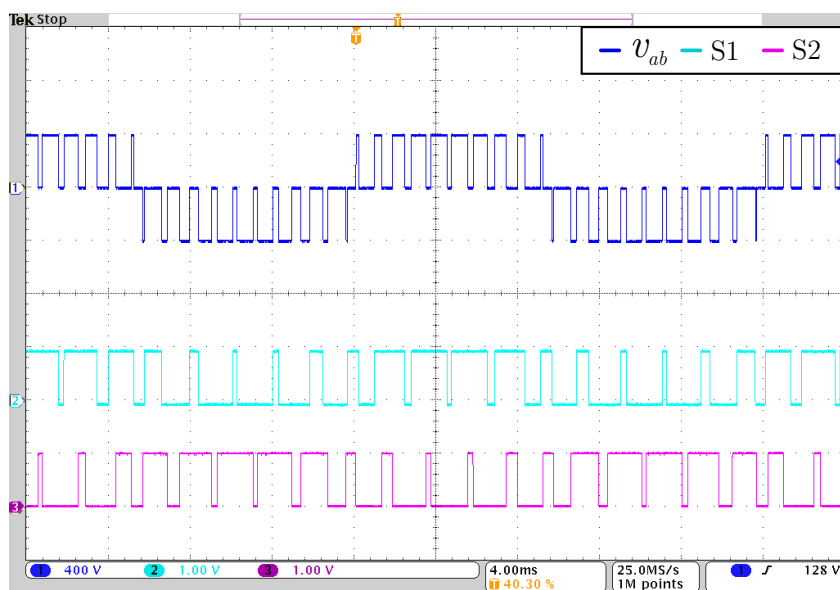
As formas de onda decorrentes da nova função custo estão mostradas na Figuras 3.8 e 3.9 para um fator $k = 0, 1$ na função custo (3.10).

Figura 3.8 – Corrente de saída e tensão de linha considerando a sequência de comutação (3.5) e a função custo (3.10).



Fonte: Autor.

Figura 3.9 – Tensão de linha de saída e a distribuição dos pulsos sobre interruptores.



Fonte: Autor.

3.6.1 Discussão dos Resultados

Observa-se que, em relação à tensão de linha não houve diferenças, uma vez que a sequência de comutação (3.5) permaneceu inalterada. Conseqüentemente, a forma de onda da corrente também permanece a mesma.

No entanto, a distribuição dos pulsos se alterou de modo a torná-la homogênea ao

longo de um ciclo de 50 Hz. Com isso, a temperatura do semicondutor permanece mais constante e a comutação não fica mais concentrada em períodos de meia onda, evitando, então, que nesse período, um par de interruptores fique sobrecarregado.

3.7 CONCLUSÃO

Neste capítulo, a modulação *space vector* ótima foi desenvolvida para o caso do inversor monofásico ponte completa. Observa-se a simplicidade do conversor, sendo seus setores compostos por apenas dois vetores de linha. Dessa forma, vê-se que a técnica apresenta maior potencial para conversores mais complexos. Além disso, vê-se a liberdade de escolha tanto na sequência de comutação como na função custo. Assim, a abordagem exposta consiste em uma metodologia genérica para se resolver um problema de otimização que incide na forma com que os vetores são dispostos.

4 MODULAÇÃO VETORIAL ÓTIMA PARA CONVERSOR TRIFÁSICO DOIS NÍVEIS

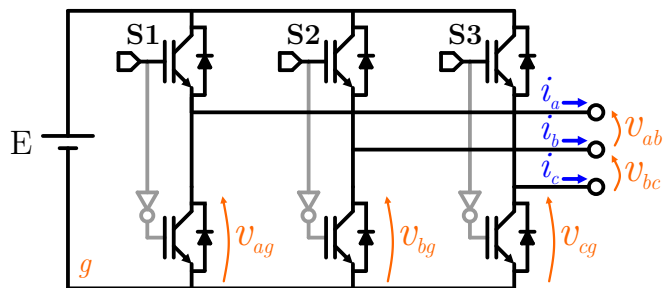
4.1 INTRODUÇÃO

Nesta seção, a modulação proposta no capítulo anterior será estendida para o caso do conversor trifásico dois níveis. As mesmas estratégias computacionais serão levadas em conta, porém de modo mais amplo para incluir as diferenças existentes no conversor trifásico. Por fim, são apresentados os resultados experimentais bem como a discussão dos mesmos, encerrando o capítulo com a conclusão do exposto.

4.2 INVERSOR TRIFÁSICO DOIS NÍVEIS

O conversor considerado neste Capítulo está representado na Figura 4.1.

Figura 4.1 – Inversor Trifásico Dois Níveis.



Fonte: Autor.

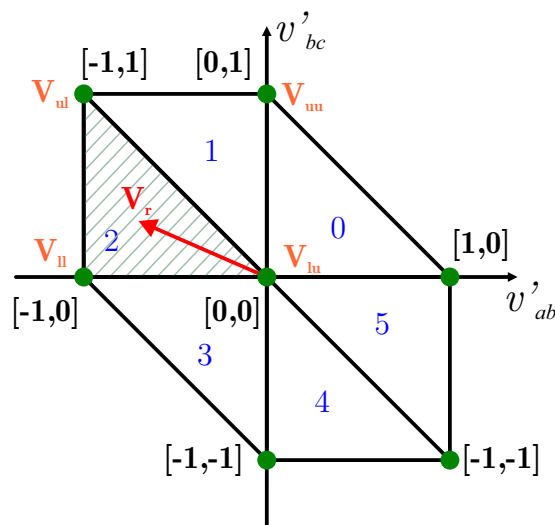
Esse conversor é composto por três braços contendo dois interruptores que atuam de maneira complementar, gerando, em cada um dos braços dois possíveis níveis de tensão. Caso o interruptor de cima do braço esteja fechado, a tensão de fase correspondente será 0 V enquanto que, do contrário, a tensão valerá a mesma do barramento CC, isto é, valerá E .

Tal topologia foi escolhida por introduzir, gradativamente, um grau de complexidade maior à estratégia proposta. Além disso, essa é uma das topologias mais largamente usadas na indústria em diversas aplicações. Dessa forma, a estratégia de modulação proposta nesta dissertação é explorada nas seções seguintes.

4.3 ESTRATÉGIA DE MODULAÇÃO PROPOSTA

Nesta seção, a estratégia de modulação vetorial ótima é desenvolvida. Para tanto, os vetores de comutação para esse conversor podem ser representados no espaço abc das tensões conforme o diagrama da Figura 4.2, onde se indica um vetor de referência \mathbf{v}_r a sintetizar, estando hachurado o setor onde ele se encontra. Cada setor na figura é denominado pelo número em seu interior.

Figura 4.2 – Diagrama vetorial em coordenadas abc para o conversor trifásico dois níveis.



Fonte: Autor.

Para fins de sistematização, esta seção traz o desenvolvimento da modulação proposta, estando organizada, respectivamente, nos tópicos de seleção dos três vetores mais próximos, na escolha da sequência de comutação, no mapeamento dos três vetores mais próximos no espaço das tensões de fase e no dos interruptores e na função custo. Posteriormente, na seção subjacente, é apresentado um fluxograma sumarizando todos os passos apresentados neste Capítulo.

4.3.1 Detecção dos três vetores mais próximos

Da mesma forma com que a modulação foi feita para o caso do conversor monofásico, necessita-se, primeiramente, de um artifício computacional para identificar o setor em que o vetor de referência \mathbf{v}_r se encontra. Primeiramente, percebe-se um aumento na complexidade do diagrama. O espaço vetorial das tensões de linha de saída, denominado de \mathbf{v}_o possui dimensão $\dim(\mathbf{v}_o) = n$, com $n = 2$ e corpo \mathbb{R}^2 . Conseqüentemente, cada setor é delimitado por $n + 1$ vetores de comutação que englobam a referência. Dessa maneira, é necessário encontrar os três vetores mais próximos que circundam o vetor \mathbf{v}_r . Para

tanto, a metodologia utilizada é a mesma abordada em (CELANOVICK; BOROYEVICK, 1999).

Inicialmente, o vetor de referência é escrito como $\mathbf{V}_r = [v'_{ab} \ v'_{bc}]^T$, onde v'_{ab} e v'_{bc} são os vetores de linha normalizados com respeito ao barramento CC, assumindo, portanto, a faixa de valores de 0 a 1. Posteriormente, cada componente do vetor é arredondado utilizando-se as funções $\text{floor}(\cdot)$ e $\text{ceil}(\cdot)$, já definidas no capítulo anterior. Como consequência, quatro vetores mais próximos são obtidos. Denotando por l (*lower*) os escalares do corpo \mathbf{v}_o resultantes do arredondamento $\text{floor}(\cdot)$ e por u (*upper*), aqueles resultantes do $\text{ceil}(\cdot)$, pode-se denominar os quatro vetores mais próximos obtidos por $\mathbf{v}_{xy} = [x, y]$, $(x, y) \in \{(l, u) \times (l, u)\}$, onde o símbolo " \times " denota o produto cartesiano.

O primeiro subíndice da notação \mathbf{v}_{xy} refere-se à posição vertical no diagrama e, o segundo, à posição horizontal. Em outras palavras, o subíndice x informa se o vetor obtido está acima ou abaixo de \mathbf{v}_r , enquanto que y indica se ele se encontra à esquerda ou à direita de \mathbf{v}_r , conforme Figura 4.2. De modo específico, pode-se obter as relações contidas em (4.1).

$$\begin{aligned} \mathbf{v}_{ll} &= [\text{floor}(v'_{ab}) \quad \text{floor}(v'_{bc})]^T \\ \mathbf{v}_{lu} &= [\text{ceil}(v'_{ab}) \quad \text{floor}(v'_{bc})]^T \\ \mathbf{v}_{ul} &= [\text{floor}(v'_{ab}) \quad \text{ceil}(v'_{bc})]^T \\ \mathbf{v}_{uu} &= [\text{ceil}(v'_{ab}) \quad \text{ceil}(v'_{bc})]^T \end{aligned} \quad (4.1)$$

No entanto, o setor para o caso desse conversor é constituído por três vetores, sendo necessário, portanto, excluir um dos vetores de comutação obtidos pelo arredondamento das componentes de \mathbf{v}_r . A fim de selecionar apenas os três vetores mais próximos, pode-se observar, da Figura 4.2, que sempre \mathbf{v}_{ul} e \mathbf{v}_{lu} farão parte do setor ao qual \mathbf{v}_r pertence. O problema reduz-se, então, à decidir entre qual dos vetores \mathbf{v}_{uu} e \mathbf{v}_{ll} o algoritmo deve selecionar. Dessa maneira, pode-se definir a relação (4.2), baseada na equação da reta.

$$R = v'_{ab} + v'_{bc} - (\text{ceil}(v'_{ab}) + \text{floor}(v'_{bc})) \quad (4.2)$$

Se $R > 0$, então \mathbf{v}_r encontra-se acima da reta que divide a região delimitada pelos quatro vetores mais próximos em dois triângulos e, então, o terceiro vetor é escolhido como sendo \mathbf{v}_{uu} . Caso $R < 0$, \mathbf{v}_r encontra-se no setor delimitado pelo triângulo inferior e, portanto, \mathbf{v}_{ll} é escolhido. Caso $R = 0$, \mathbf{v}_r encontra-se exatamente sobre a reta que divide os dois setores e, portanto, a escolha entre \mathbf{v}_{ll} ou \mathbf{v}_{uu} é indiferente.

Deforma resumida, denominando por \mathbf{v}_{rm} o vetor remanescente selecionado (podendo ser \mathbf{v}_{ll} ou \mathbf{v}_{uu}), cada setor é formado pelos três seguintes vetores: \mathbf{v}_{lu} , \mathbf{v}_{ul} e \mathbf{v}_{rm} . Após, é necessário calcular a parcela de tempo que cada vetor de comutação deve aparecer em um período de amostragem. Para tanto, pode-se observar que o vetor médio de saída

$\bar{\mathbf{v}} = [\bar{v}'_{ab} \ \bar{v}'_{bc}]^T$ pode ser escrito como na Equação 4.3.

$$\begin{bmatrix} \bar{\mathbf{v}} \\ 1 \end{bmatrix}_{3 \times 1} = \begin{bmatrix} \mathbf{v}_{lu} & \mathbf{v}_{ul} & \mathbf{v}_{rm} \\ 1 & 1 & 1 \end{bmatrix}_{3 \times 3} \begin{bmatrix} d_{lu} \\ d_{ul} \\ d_{rm} \end{bmatrix}_{3 \times 1} \quad (4.3)$$

Onde d_{lu} , d_{ul} e d_{rm} referem-se às razões cíclicas associadas à, respectivamente, \mathbf{v}_{lu} , \mathbf{v}_{ul} e \mathbf{v}_{rm} . Nota-se que, na equação (4.3), uma matriz linha unitária foi concatenada ao lado esquerdo e à primeira matriz do lado direito. Tal manipulação deve ser realizada para que a última matriz mencionada seja inversível. Isso implica em $d_1 + d_2 + d_3 = 1$. Portanto, de (4.3), pode-se obter equação (4.4), a qual é utilizada para o cálculo das razões cíclicas referentes à cada vetor, também referidas como *dwell time*.

$$\begin{bmatrix} d_{lu} \\ d_{ul} \\ d_{rm} \end{bmatrix}_{3 \times 1} = \begin{bmatrix} \mathbf{v}_{lu} & \mathbf{v}_{ul} & \mathbf{v}_{rm} \\ 1 & 1 & 1 \end{bmatrix}_{3 \times 3}^{-1} \begin{bmatrix} \bar{\mathbf{v}} \\ 1 \end{bmatrix}_{3 \times 1} \quad (4.4)$$

No entanto, baseando-se na metodologia presente em (DE CASTRO; CORREA; JACOBINA, 2013), as razões cíclicas podem ser calculadas de maneira mais concisa como

$$\begin{bmatrix} d_{lu} \\ d_{ul} \end{bmatrix} = \begin{cases} \begin{bmatrix} v'_{ab} - \text{floor}(v'_{ab}) \\ v'_{bc} - \text{floor}(v'_{bc}) \end{bmatrix}, & R < 0 \\ \begin{bmatrix} -(v'_{bc} - \text{ceil}(v'_{bc})) \\ -(v'_{ab} - \text{ceil}(v'_{ab})) \end{bmatrix}, & R \geq 0 \end{cases} \quad (4.5)$$

$$d_{rm} = 1 - d_{lu} - d_{ul}$$

4.3.2 Escolha da sequência de comutação

Posteriormente à seleção dos três vetores mais próximos, deve-se definir uma sequência de comutação a fim de poder implementar a técnica de modulação proposta. Primeiramente, essa pode ser definida por:

$$\mathbf{s} = \mathbf{v}_1 \ \mathbf{v}_2 \ \mathbf{v}_3 \quad (4.6)$$

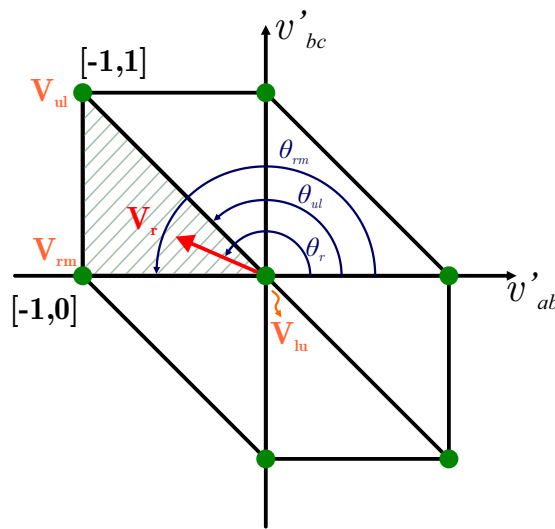
Onde \mathbf{v}_1 denota o vetor de comutação que apresenta, no espaço das tensões de fase correspondente (chamado de espaço Ω), a cardinalidade mais alta. Isto é, aquele que tem maior número de vetores redundantes nesse espaço, sendo, portanto, o vetor nulo. Deve ser lembrado que, para cada vetor no espaço das tensões de linha existe um ou mais vetores no espaço das tensões de fase que gera o referido vetor. É importante observar que, no caso do conversor trifásico dois níveis, a cardinalidade dos vetores de comutação

no espaço das tensões de fase é o mesmo em relação ao dos estados dos interruptores.

Ainda, resta definir \mathbf{v}_2 e \mathbf{v}_3 . Como os demais vetores além do nulo têm o mesmo número de vetores de fase, o critério escolhido para seleção consiste em definir \mathbf{v}_2 e \mathbf{v}_3 de tal modo que expressem uma trajetória no sentido anti-horário. Por exemplo, para o vetor de referência \mathbf{v}_r , na Figura 4.2, $\mathbf{v}_1 = [0, 0]$, $\mathbf{v}_2 = [-1, 1]$ e $\mathbf{v}_3 = [-1, 0]$.

A fim de se ter uma metodologia para a seleção de \mathbf{v}_2 e \mathbf{v}_3 por meio do critério exposto, observe a Figura 4.3. Nota-se que, o vetor \mathbf{v}_2 corresponderá àquele onde a diferença entre o ângulo do vetor de referência e do vetor de comutação é positiva. Neste caso, observa que essa condição ocorrerá para θ_{ul} , uma vez que:

Figura 4.3 – Detecção dos vetores que formam uma sequência anti-horária



Fonte: Autor.

$$\theta_r - \theta_{ul} \geq 0 \quad (4.7)$$

É fácil observar que a desigualdade (4.7) pode ser reescrita conforme a (4.8).

$$\operatorname{tg}^{-1} \left(\frac{\begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{v}_r}{\begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{v}_r} \right) - \operatorname{tg}^{-1} \left(\frac{\begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{v}_{ul}}{\begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{v}_{ul}} \right) \geq 0 \quad (4.8)$$

Onde a multiplicação pela matrizes linha $\begin{bmatrix} 0 & 1 \end{bmatrix}$ e $\begin{bmatrix} 1 & 0 \end{bmatrix}$ tem o propósito de selecionar, respectivamente, a ordenada e a abscissa dos vetores envolvidos. Ou, ainda, de (4.8), pode-se escrever:

$$\frac{\begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{v}_r}{\begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{v}_r} \geq \frac{\begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{v}_{ul}}{\begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{v}_{ul}} \quad (4.9)$$

Como consequência, \mathbf{v}_3 é o vetor remanescente dos três que constituem o setor onde \mathbf{v}_r se encontra. Não menos importante, deve-se observar a correspondência dos vetores

de comutação às suas respectivas razões cíclicas. Por isso, se faz necessário definir d_1 , d_2 e d_3 associados à \mathbf{v}_1 , \mathbf{v}_2 e \mathbf{v}_3 , respectivamente, os quais provém de (4.5).

Deve-se observar que é possível, ainda, que um vetor no espaço das tensões de fase apresentem um ou mais vetores correspondentes no espaço dos estados de condução dos interruptores. No caso do inversor trifásico dois níveis essa correspondência é de um para um. Isto é, um vetor de fase possui um único estado de condução relacionado. Portanto, aquele vetor de linha que tiver maior número de correspondentes no espaço das tensões de fase, também o terá no espaço dos estados dos interruptores.

Dessa forma, com base no exposto e do raciocínio sintetizado na inigualdade (4.9) para o caso particular representado na Figura 4.3, pode-se definir, de maneira formal, \mathbf{v}_1 , \mathbf{v}_2 e \mathbf{v}_3 e d_1 , d_2 e d_3 como mostrado no conjunto de relações (4.10).

$$\begin{aligned}
 \mathbf{v}_1 = \mathbf{v}_{xy}, d_1 = d_{xy} : & \quad |\Omega(\mathbf{v}_{xy})| = \max \{ |\Omega(\mathbf{v}_{ab})| : (a, b) \in \{(l, u) \times (l, u)\} \\
 & \quad | (x, y) \in \{(l, u) \times (l, u)\} \\
 \mathbf{v}_2 = \mathbf{v}_{xy}, d_2 = d_{xy} : & \quad \frac{\mathbf{v}_r \begin{bmatrix} 0 & 1 \end{bmatrix}^T}{\mathbf{v}_r \begin{bmatrix} 1 & 0 \end{bmatrix}^T} \geq \frac{\mathbf{v}_{xy} \begin{bmatrix} 0 & 1 \end{bmatrix}^T}{\mathbf{v}_{xy} \begin{bmatrix} 1 & 0 \end{bmatrix}^T} \\
 & \quad | (x, y) \in \{(l, u), (u, l)\} \cup \{(a, b) : \mathbf{v}_{ab} = \mathbf{v}_{rm}\} \setminus \{(a, b) : \mathbf{v}_{ab} = \mathbf{v}_1\} \\
 \mathbf{v}_3 = \mathbf{v}_{xy}, d_3 = d_{xy} : & \quad (x, y) \in \{(l, u), (u, l)\} \cup \{(a, b) : \mathbf{v}_{ab} = \mathbf{v}_{rm}\} \\
 & \quad \setminus \{(a, b) \cup (c, d) : \mathbf{v}_{ab} = \mathbf{v}_1 \wedge \mathbf{v}_{cd} = \mathbf{v}_2\}
 \end{aligned} \tag{4.10}$$

Onde a notação $|\cdot|$ representa a cardinalidade do conjunto e $\Omega(\cdot)$ é denominado o operador que mapeia os vetores no espaço das tensões de linha nos vetores correspondentes no espaço das tensões de fase.

4.3.3 Mapeamento no espaço das tensões de fase e no dos interruptores

No tópico anterior, foi introduzido o operador $\Omega(\cdot)$, o qual mapeia as tensões de linha nas tensões de fase. Este pode ser matematicamente definido por:

$$\Omega(\mathbf{v}_x) = feasible \left(\left[\begin{array}{c} floor \left(\mathbf{M} \begin{bmatrix} \mathbf{v}_x \\ 3 \end{bmatrix} \right) \\ ceil \left(\mathbf{M} \begin{bmatrix} \mathbf{v}_x \\ 3 \end{bmatrix} \right) \end{array} \right] \right) \tag{4.11}$$

Onde cada coluna de Ω representa um vetor de fase, formado por v'_a , v'_b e v'_c , respectivamente, de modo a corresponder ao vetor coluna \mathbf{v}_x , um vetor de tensão de linha

qualquer. A matriz auxiliar \mathbf{M} é dada pela equação (4.12);

$$\mathbf{M} = \frac{0,99}{3} \begin{bmatrix} 2 & 1 & 1 \\ -1 & 1 & 1 \\ -1 & -2 & 1 \end{bmatrix} \quad (4.12)$$

Em (4.11), é possível que valores retornados sejam inválidos. Se, em uma coluna, tiver um elemento de valor diferente de 0 ou 1, a mesma deve ser excluída, pois isso significa que o valor de tensão exigida em um dos braços do conversor é infactível. Por essa razão, define-se a função *feasible* (\cdot), a qual elimina os vetores coluna que contenham no mínimo uma tensão de fase irrealista. Portanto, essa função pode ser definida por

$$feasible(\mathbf{U}_{3 \times 2}) = \begin{bmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \\ z_{31} & z_{32} \end{bmatrix} \quad z_{ij} = \begin{cases} u_{ij}, & u_{ij} = 0 \vee u_{ij} = 1 \quad \forall i|j \in \{1, 2\} \\ \emptyset, & u_{ij} \neq 0 \wedge u_{ij} \neq 1 \quad \forall i|j \in \{1, 2\} \end{cases} \quad (4.13)$$

É importante observar que como o vetor \mathbf{V}_x é assumido ser normalizado e o mapeamento do espaço Ω para \mathbf{S} , os vetores no espaço dos estados dos interruptores, se dá na relação de um para um:

$$\mathbf{S}(\mathbf{v}_x) = \Omega(\mathbf{v}_x) \quad (4.14)$$

Neste ponto, se tem os vetores de comutação definidos, bem como os correspondentes estados de condução dos interruptores e as razões cíclicas associadas. Para fins de clareza, a Tabela 4.1 sumariza os resultados das equações (4.1), (4.10) e (4.14) de acordo com o setor em que o vetor de referência \mathbf{v}_r se encontra.

4.3.4 Função custo

Definida a sequência (4.6), se faz necessária uma função custo para selecionar, dos conjuntos \mathbf{s}_x , onde o subíndice x refere-se à qualquer um dos vetores da sequência, o vetor redundante a ser empregado. Essa função pode ser dada por (4.15).

$$g = \|(\mathbf{s}_L - \mathbf{s}_i)\|_1, \quad i \in \{0..1\} \quad (4.15)$$

Onde \mathbf{s}_L é o vetor no espaço dos interruptores imediatamente anterior dentro sequência de comutação aplicado ao conversor, \mathbf{s}_i é o i -ésimo vetor da sequência sob teste e $\|\cdot\|_1$ é a norma um do argumento, definida em (3.9).

Tabela 4.1 – Mapeamento dos vetores no espaço das tensões de linha para o espaço dos interruptores

<i>Setor</i>	\mathbf{v}_1	\mathbf{v}_2	\mathbf{v}_3	\mathbf{s}_1	\mathbf{s}_2	\mathbf{s}_3
0	[0,0]	[1,0]	[0,1]	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$
1	[0,0]	[0,1]	[-1,1]	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$
2	[0,0]	[-1,1]	[-1,0]	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$
3	[0,0]	[-1,0]	[0,-1]	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$
4	[0,0]	[0,-1]	[1,-1]	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$
5	[0,0]	[1,-1]	[1,0]	$\begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

4.4 FLUXOGRAMA

Dados os passos anteriores, se faz importante um fluxograma sumarizando as etapas a serem realizadas pela rotina de modulação. Esse fluxograma encontra-se na Figura 4.4. Para fins de clareza, cada bloco denotado no fluxograma pelos símbolos de chave serão explicados.

BLOCO 1: O algoritmo de modulação ótima recebe o vetor de referência, bem como o último vetor de comutação no espaço dos interruptores aplicado no período de amostragem anterior. Primeiramente, é verificado se o vetor de referência, dado no espaço das tensões de linha, está dentro do diagrama vetorial da Figura 4.2. Para tanto, o

Figura 4.4 – Fluxograma com os passos da rotina de modulação para o inversor trifásico dois níveis.

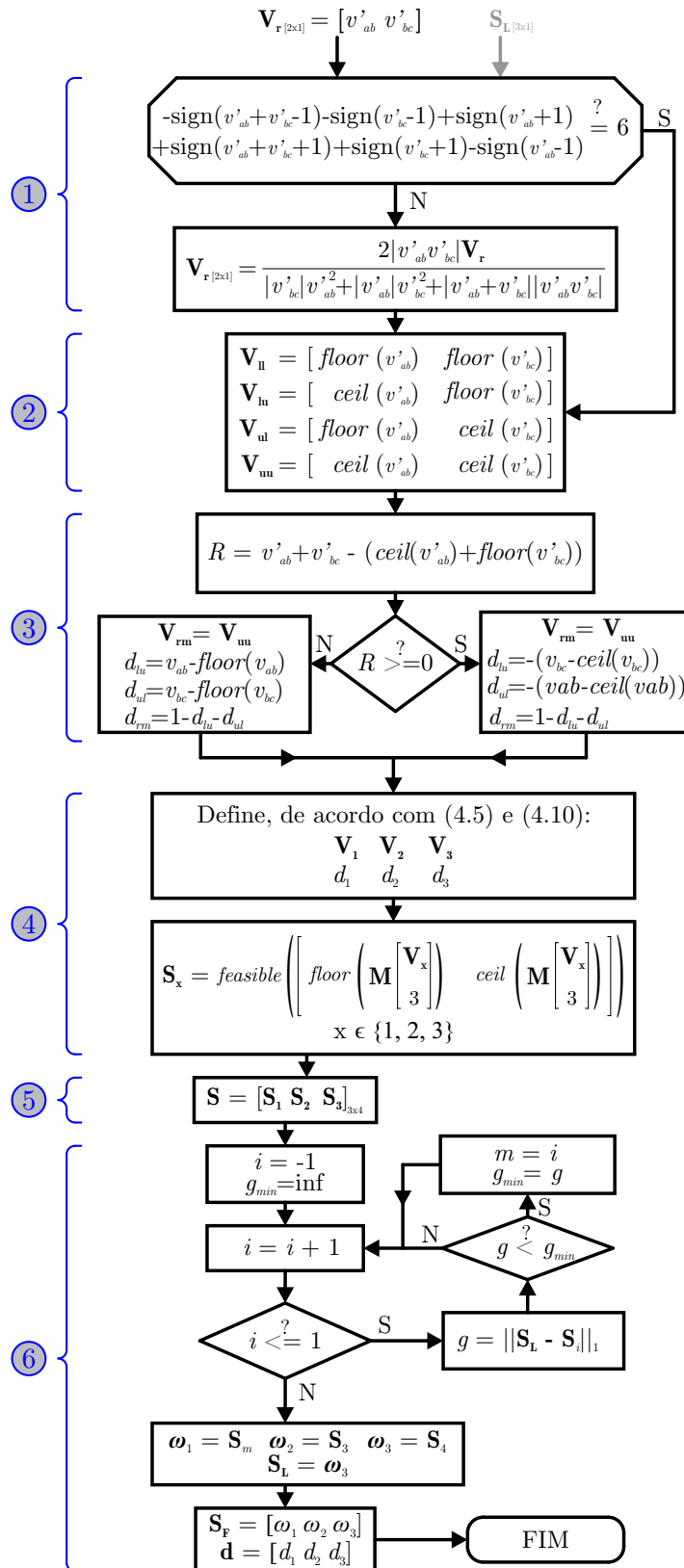


diagrama vetorial da Figura 4.2 é pensado como sendo delimitado pelas retas

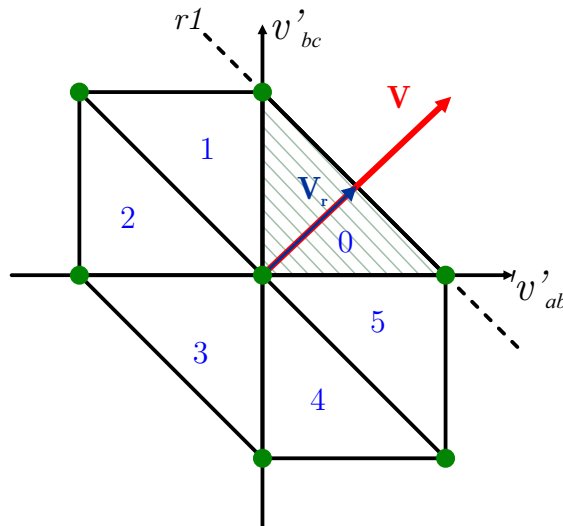
$$\begin{aligned}
 v_{ab} + v_{bc} - 1 &= 0 \\
 v_{bc} - 1 &= 0 \\
 v_{ab} + 1 &= 0 \\
 v_{ab} + v_{bc} + 1 &= 0 \\
 v_{bc} + 1 &= 0 \\
 v_{ab} - 1 &= 0
 \end{aligned} \tag{4.16}$$

Dessa maneira, se \mathbf{v}_r estiver acima da primeira reta, o resultado de $v'_{ab} + v'_{bc} - 1$ será positivo. Analisando o diagrama vetorial do conversor, é fácil perceber que, para um vetor de referência poder ser sintetizado, ele deve estar na região delimitada por todas as retas em (4.16). Essa condição pode ser escrita como

$$\begin{aligned}
 -\text{sign}(v_{ab}+v_{bc}-1) - \text{sign}(v_{bc}-1) + \text{sign}(v_{ab}+1) + \\
 \text{sign}(v_{ab}+v_{bc}+1) + \text{sign}(v_{bc}+1) - \text{sign}(v_{ab}-1) = 6
 \end{aligned} \tag{4.17}$$

Caso essa condição ocorra, é necessário, ainda, saturar o módulo do vetor, de forma ao vetor sintetizado estar dentro do diagrama vetorial. Para isso, observe a Figura 4.5.

Figura 4.5 – Saturação em módulo para um vetor de referência fora dos limites do diagrama vetorial.



Fonte: Autor.

O vetor de referência inicial encontra-se no setor 0 e excede os limites do diagrama. Dessa maneira, seu módulo deve ser saturado. Para tanto, deve-se observar que a reta $r1$ pode ser descrita por

$$r1 : v'_{ab} + v'_{bc} = 1 \tag{4.18}$$

Dessa maneira, o módulo de $\mathbf{v}_{2 \times 1}$, o vetor de referência inicial, deve ser reduzido até tocar $r1$. Então, considerando o vetor unitário $\mathbf{v}_n = \frac{\mathbf{v}}{\|\mathbf{v}\|}$, pode-se encontrar o fator multiplicativo k , tal que

$$\frac{k}{\|\mathbf{v}\|} \begin{bmatrix} 1 & 1 \end{bmatrix} \mathbf{v} = 1 \quad (4.19)$$

Ou seja, k é o fator que, multiplicando o vetor unitário v_n , toca $r1$. Após manipulação algébrica, é fácil perceber o vetor de referência saturado, para o setor 0, é dado por

$$\mathbf{v}_r = \frac{\mathbf{v}}{\begin{bmatrix} 1 & 1 \end{bmatrix} \mathbf{v}} \quad (4.20)$$

O mesmo raciocínio pode ser repetido para os outros setores. Como consequência em caso de saturação de $\mathbf{v} = [v'_{ab} \ v'_{bc}]^T$, de modo compacto, \mathbf{v}_r pode ser escrito como

$$\mathbf{v}_r = \frac{2|v'_{ab}v'_{bc}|}{|v'_{bc}|v'^2_{ab} + |v'_{ab}|v'^2_{bc} + |v'_{ab} + v'_{bc}||v'_{ab}v'_{bc}|} \mathbf{v} \quad (4.21)$$

Assim, garante-se que existirão vetores de comutação para sintetizar a referência.

BLOCO 2: Executa-se a rotina para encontrar os quatro vetores mais próximos que englobam a referência, conforme (4.1).

BLOCO 3: Seleciona-se, dentre os quatro vetores encontrados na etapa anterior, aqueles que compõem o setor de três vetores que englobam a referência.

BLOCO 4: Define-se os vetores \mathbf{v}_1 , \mathbf{v}_2 e \mathbf{v}_3 , bem como as razões cíclicas d_1 , d_2 e d_3 de acordo com (4.10) e (4.5). Uma vez definidos os vetores de linha anteriores, se faz necessário mapeá-los no espaço dos estados dos interruptores. Para tanto, é aplicada a relação (4.11), levando-se em conta (4.14).

BLOCO 5: Cria-se a sequência de comutação desejada, podendo ser qualquer uma, uma vez que a sequência de comutação é um grau de liberdade que a modulação vetorial oferece, desde que as razões cíclicas sejam corretamente distribuídas entre os pulsos.

BLOCO 6: Executa-se a rotina de seleção dos vetores no espaço dos estados dos interruptores, sendo que essa pode incorporar todos os objetivos do projetista. Nesse caso, a função custo é aquela referente ao menor número de comutações de um vetor para o outro na ordem em que aparecem na sequência.

Em termos de código, o fluxograma da Figura 4.4 pode ser escrito da seguinte forma, tomando por base a linguagem C.

```

1 float Vr[2], V1[2], V2[2], V3[2], Vrm[2], v1l[2], v1u[2], vul
   [2], vuu[2], du, dl, gmin, g, kr, R, d1, d2, d3, dl, du, drm
   , s1, s2, s3;
2 int i, j, m, Vu, V1, SL[3], Su[2][2], S1[2][2], S[3][4],
3 SF[2][3];

```

```
4
5 //Testa se a vetor está fora do Diagrama Vetorial e, em caso
   afirmativo, satura-o.
6
7 if(-sign(Vr[0]+Vr[1]-1) - sign(Vr[1]-1) + sign(Vr[0]+1) + sign(
   Vr[0]+Vr[1]+1) + sign(Vr[1]+1) - sign(Vr[0]-1) != 6 ){
8
9 kr = 2*abs(Vr[0]*Vr[1])/(abs(Vr[1])*Vr[0]*Vr[0] + abs(Vr[0])*Vr
   [1]*Vr[1] + abs(Vr[0]+Vr[1])abs(Vr[0]*Vr[1]))
10
11 Vr[0]=kr*Vr[0];
12 Vr[1]=kr*Vr[1];
13 }
14
15 //Encontra os vetores Vll, Vlu, Vul, Vuu
16 vll[0] = floor(vab); vll[1] = floor(vbc);
17 vlu[0] = ceil(vab); vlu[1] = floor(vbc);
18 vul[0] = floor(vab); vul[1] = ceil(vbc);
19 vuu[0] = ceil(vab); vuu[1] = ceil(vbc);
20
21 //Seleciona o vetor Vrm, o terceiro vetor do setor
22 R = vab + vbc - (ceil(vab)+floor(vbc));
23 if(R>=0){
24   Vrm[0] = Vuu[0]; Vrm[1] = Vuu[1];
25   dlu = -(vbc-ceil(vbc));
26   dul = -(vab-ceil(vab));
27 }
28 else{
29   Vrm[0] = Vll[0]; Vrm[1] = Vll[1];
30   dlu = (vab-floor(vab));
31   dul = (vbc-floor(vbc));
32 }
33 drm = 1 - dlu - dul;
34
35 // Define V1, V2 e V3 e calcula d1, d2 e d3
36 if(Vlu[0]==0 && Vlu[1]==0){ V1[0] = Vlu[0]; V1[1] = Vlu[1]; d1=
   dlu; } else
37 if(Vul[0]==0 && Vul[1]==0){ V1[0] = Vul[0]; V1[1] = Vul[1]; d1=
   dul; } else
```

```

38 {V1[0] = Vrm[0]; V1[1] = Vrm[1]; d1=drm;}
39
40 if(Vlu[0]!=0 && Vlu[1]!=0 && Vr[1]*Vlu[0]>=Vr[0]*Vlu[1]){ V2[0]
    = Vlu[0]; V2[1] = Vlu[1]; d2=dlu; } else
41 if(Vul[0]!=0 && Vul[1]!=0 && Vr[1]*Vul[0]>=Vr[0]*Vul[1]){ V2[0]
    = Vul[0]; V2[1] = Vul[1]; d2=dul;} else
42 {V2[0] = Vrm[0]; V2[1] = Vrm[1]; d2=drm;}
43
44 if(Vlu[0]!=V1[0] && Vlu[1]!=V1[1] && Vlu[0]!=V2[0] && Vlu[1]!=
    V2[1]){ V3[0] = Vlu[0]; V3[1] = Vlu[1]; d3=dlu; } else
45 if(Vul[0]!=V1[0] && Vul[1]!=V1[1] && Vul[0]!=V2[0] && Vul[1]!=
    V2[1]){ V3[0] = Vul[0]; V3[1] = Vul[1]; d3=dul;} else
46 {V3[0] = Vrm[0]; V3[1] = Vrm[1]; d3=drm; }
47
48
49
50 //Mapeia V1, V2 e V3 em S1, S2 e S3
51 s1 = 0.6667*V1[0] + 0.3333*V1[1]+1;
52 s2 = 0.6667*V1[0] + 0.3333*V1[1]+1;
53 s3 = 0.6667*V1[0] + 0.3333*V1[1]+1;
54 S[0][0] = floor(s1); S[0][1] = ceil(s1);
55 S[0][0] = floor(s2); S[0][1] = ceil(s2);
56 S[0][0] = floor(s3); S[0][1] = ceil(s3);
57
58 s1 = 0.6667*V2[0] + 0.3333*V2[1]+1;
59 s2 = 0.6667*V2[0] + 0.3333*V2[1]+1;
60 s3 = 0.6667*V2[0] + 0.3333*V2[1]+1;
61 S[1][0] = floor(s1);
62 S[1][0] = floor(s2);
63 S[1][0] = floor(s3);
64
65 s1 = 0.6667*V3[0] + 0.3333*V3[1]+1;
66 s2 = 0.6667*V3[0] + 0.3333*V3[1]+1;
67 s3 = 0.6667*V3[0] + 0.3333*V3[1]+1;
68 S[2][0] = floor(s1);
69 S[2][0] = floor(s2);
70 S[2][0] = floor(s3);
71
72 //Seleciona um vetor ótimo dentre os redundantes do vetor nullo

```

```

73 gmin=100000000;
74 for(i=0; i<1; i++){
75     g = abs(SL[0]-S[i][0]) + abs(SL[1]-S[i][1]) +
                                                abs(SL[2]-S[i][2]);
76     if(g<gmin){
77         gmin = g;
78         m = i;
79     }
80 }
81
82 //Envia a sequência para o modulador vetorial (FPGA)
83 EsvmRegs.DATA.DWT.DW0 = EsvmDwPerc(d1*100.0, 0);
84 EsvmRegs.DATA.DWT.DW1 = EsvmDwPerc(d2*100.0, 1);
85
86 EsvmRegs.DATA.VEC.W0 = EsvmSequence(S[2][m]*100+S[1][m]*10+S
    [0][m]);
87 EsvmRegs.DATA.VEC.W1 = EsvmSequence(S[2][2]*100+S[1][2]*10+S
    [0][2]);
88 EsvmRegs.DATA.VEC.W2 = EsvmSequence(S[2][3]*100+S[1][3]*10+S
    [0][3]);

```

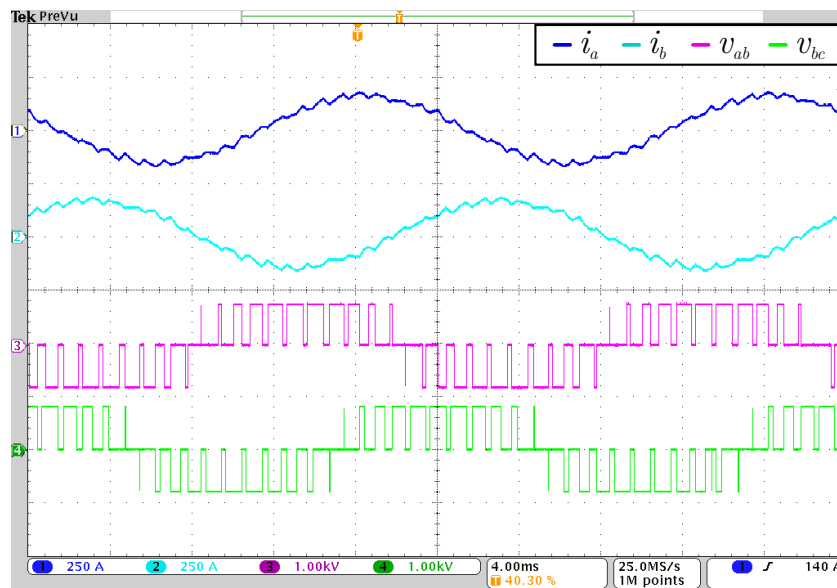
Observa-se que a metodologia empregada para o conversor trifásico é a mesma aplicada ao monofásico. No entanto, sua complexidade é maior devido ao maior número de vetores de comutação.

4.5 RESULTADOS EXPERIMENTAIS: SEQUÊNCIA 1

Nesta seção, serão apresentados os resultados da modulação ótima desenvolvida neste capítulo, com a definição dos vetores dada em (4.10). Isto é, o primeiro vetor da sequência corresponde ao vetor nulo e, os demais, são dados no sentido anti-horário.

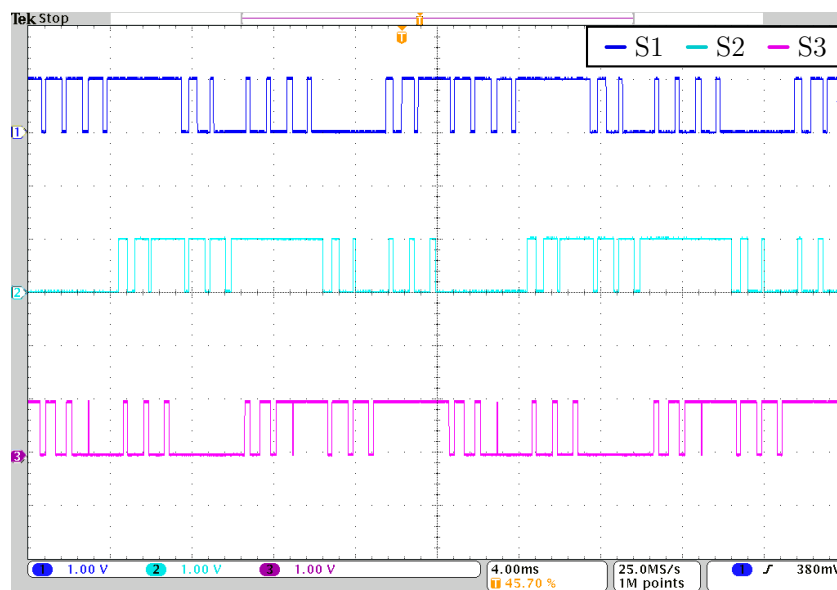
Para fins de experimentação, considera-se o DSP TEXAS TMS320F28335 em conjunto com a FPGA Xilinx Spartan 3E-500, tendo o Typhoon HIL402 como emulador do conversor, conforme Apêndice A. Para a simulação, é considerada, na saída do conversor da Figura 4.1, uma carga trifásica RL, consumindo uma potência de 100 kW, com $FP = 0,92$, barramento CC de 800 V e referência senoidal com índice de modulação igual à $m_i = 0,8$. As formas de onda de saída encontram-se conforme as Figuras 4.6 e 4.7 para uma frequência de amostragem de 1 kHz.

Figura 4.6 – Correntes de saída e tensões de linha considerando a sequência de comutação (4.6) e a função custo (4.15).



Fonte: Autor.

Figura 4.7 – Distribuição dos pulsos entre os interruptores.



Fonte: Autor.

4.5.1 Discussão dos Resultados

Na Figura 4.6, se tem as correntes de linha de saída do conversor da Figura 4.1, mostrando que a modulação está gerando uma tensão de saída média conforme a referência senoidal. Ainda, na mesma imagem, se tem, na parte de baixo, as tensões de linha v_{ab} e v_{bc} . Nota-se a defasagem entre o valor médio dos pulsos de saída e a corrente sintetizada

em função do fator de potência.

Já na Figura 4.7, se tem os pulsos referentes a cada um dos interruptores do conversor. Como se pode observar, há períodos em que eles permanecem sem comutar. Isso é uma clara consequência da função custo (4.15), que escolhe os vetores justamente de modo a minimizar o número de comutações.

No entanto, devido à rigidez decorrente da forma com que foram definidos os vetores de comutação do setor, isto é, devendo estarem em sentido anti-horário, observa-se que uma distribuição não uniforme dos pulsos. Assim, torna-se interessante flexibilizar a função custo (4.15), de maneira a dar mais liberdade à escolha dos vetores por parte do algoritmo. Dessa maneira, uma segunda estratégia para a composição da sequência de comutação é apresentada na seção a seguir.

4.6 RESULTADOS EXPERIMENTAIS: SEQUÊNCIA 2

Nesta seção, uma alteração é feita na forma com que os vetores são dispostos na sequência. Igualmente ao caso anterior, o primeiro vetor da sequência corresponde ao vetor nulo. Porém, o vetor adjacente é aquele, dentre os dois remanescentes, apresenta a menor distância em termos de norma um. Isto é, aquele vetor que apresenta a menor norma um da diferença do vetor candidato com o anterior \mathbf{sL} .

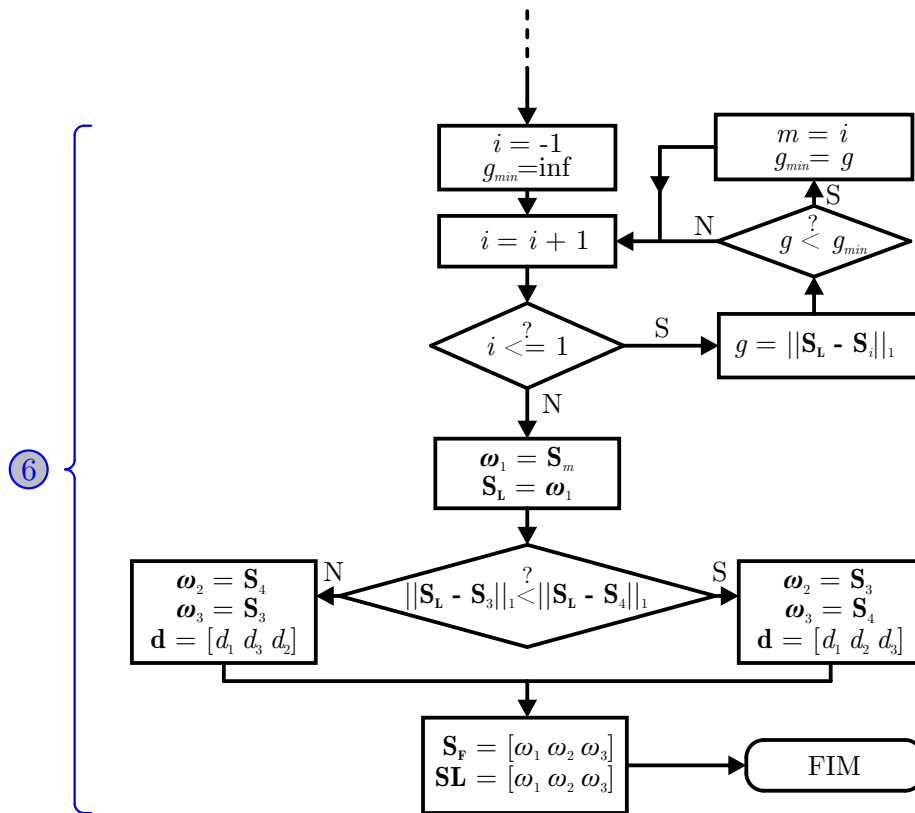
Ao contrário do caso anterior, a escolha dos vetores é feita de modo a otimizar as comutações, no sentido de diminuí-las. Dessa maneira, o fluxograma da Figura 4.4 pode ser atualizado, conforme Figura 4.8. Essa mudança é refletida nas formas de onda de saída conforme as Figuras 4.9 e 4.10.

4.6.1 Discussão dos Resultados

Quanto à síntese da tensão média de saída da Figura 4.9, pode-se perceber que, conforme esperado, não houve diferença. O que se observa, no entanto, é que o valor instantâneo da corrente de saída não apresenta a mesma forma. Além disso, o padrão dos pulsos de saída não ficaram simétricos como no caso anterior.

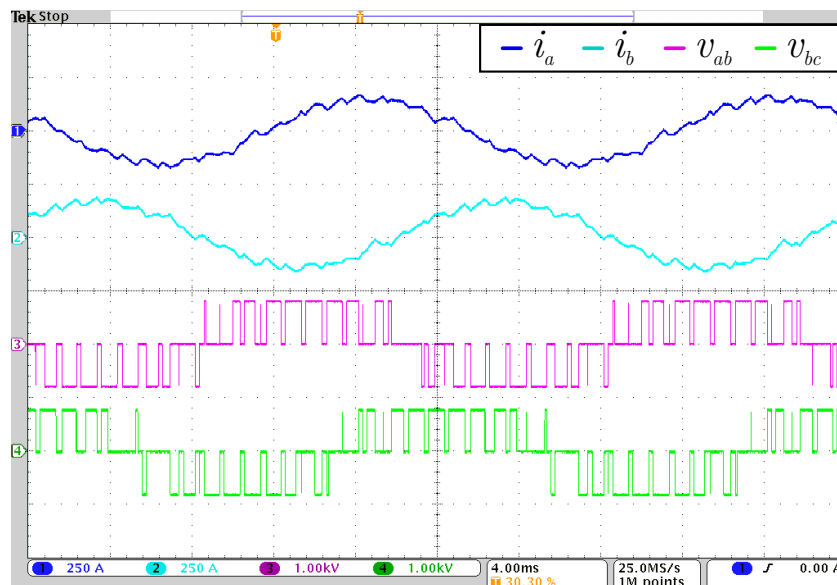
No entanto, os pulsos nos interruptores, conforme mostrados na Figura 4.10 apresentaram uma distribuição mais homogênea. Uma distribuição mais bem distribuída entre os interruptores é interessante em termos de aquecimento e vida útil dos componentes.

Figura 4.8 – Atualização do Bloco 6 do fluxograma da Figura 4.4 com o novo critério de seleção dos vetores.



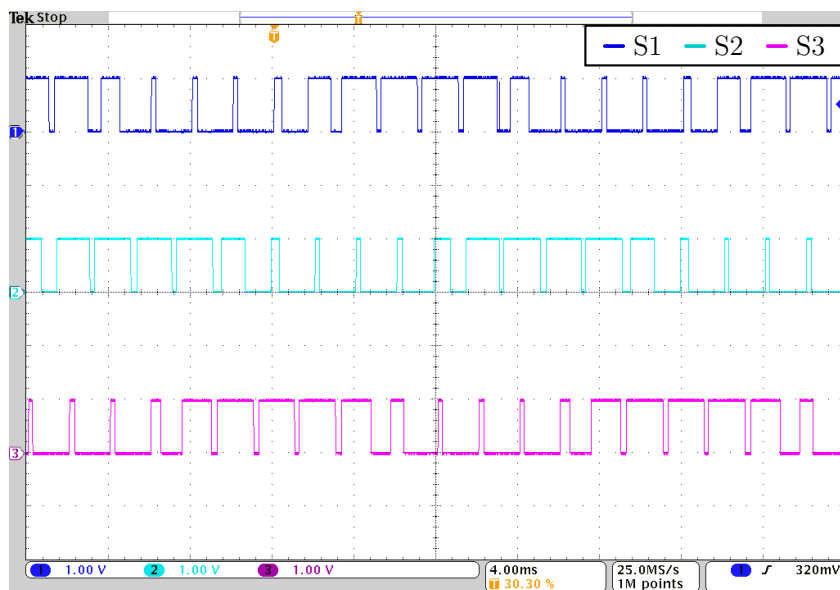
Fonte: Autor.

Figura 4.9 – Correntes de saída e tensões de linha considerando o fluxograma da Figura 4.8.



Fonte: Autor.

Figura 4.10 – Distribuição dos pulsos entre os interruptores.



Fonte: Autor.

4.7 CONCLUSÃO

Neste capítulo, a modulação *space vector* ótima foi desenvolvida para o caso do inversor trifásico dois níveis. Neste caso, a complexidade do conversor é maior e o problema de como a sequência de comutação bem como a função custo são escolhidas se intensifica. Portanto, duas formas de função custo são implementadas evidenciando as diferenças entre as duas através dos resultados experimentais. Por fim, observa-se que não há variáveis internas a controlar pelo fato da topologia ser relativamente simples. Assim, cabe explorar a metodologia de modulação ao caso do inversor trifásico NPC, que se encontra no capítulo seguinte.

5 MODULAÇÃO VETORIAL ÓTIMA PARA CONVERSOR TRIFÁSICO NPC

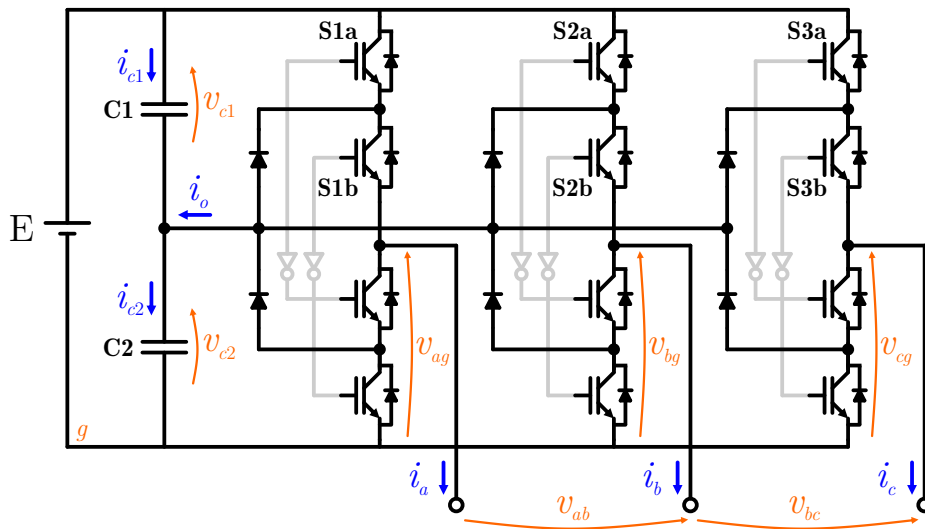
5.1 INTRODUÇÃO

Neste capítulo, a modulação ótima desenvolvida para o conversor monofásico e trifásico dois níveis será aplicada ao conversor trifásico NPC. Primeiramente, encontra-se uma breve descrição da topologia bem como de suas especificidades. Após, a metodologia de modulação proposta nesta dissertação é desenvolvida para a topologia descrita. Por fim, são apresentados os resultados experimentais bem como a discussão dos mesmos, encerrando o capítulo com a conclusão do exposto.

5.2 INVERSOR TRIFÁSICO NPC

O conversor considerado neste Capítulo está representado na Figura 5.1.

Figura 5.1 – Inversor trifásico NPC.



Fonte: Autor.

Esse conversor, trifásico, é composto por três braços, sendo que cada um contém quatro interruptores: dois superiores e dois inferiores. Os inferiores trabalham de modo complementar aos superiores, conforme indicado pelo circuito lógico em cinza na Figura 5.1. Nota-se que existe a presença de diodos, os quais garantem que a tensão sobre os interruptores não excederá a metade do barramento CC, possibilitando a utilização de semicondutores de mais baixa especificação de tensão (DAI; WONG; HAN, 2006). Além

disso, há dois capacitores, onde cada um deve estar carregado com metade da tensão do barramento, possibilitando a geração de três níveis de saída para cada braço.

A tensão para cada braço, em função dos estados dos interruptores, estão explicitados na Tabela 5.1, onde v_{xg} é a tensão de fase para o braço x , $x \in \{1..3\}$ normalizada com respeito à metade do barramento CC. Assim, $v_{xg} = 1$ corresponde à $E/2$, e $v_{xg} = 2$, à E . Essa normalização foi escolhida por conter apenas números inteiros, o que torna mais fácil de se trabalhar com os truncamentos $\text{floor}(\cdot)$ e $\text{ceil}(\cdot)$ e, conseqüentemente, mais conveniente para o algoritmo de otimização.

Da Tabela 5.1, vê-se que não está presente a combinação de interruptores [1 0]. Isto ocorre porque, nessa configuração, a tensão de fase depende do sentido da corrente e, por tal dependência, esse estado não é desejado. Isso implica em um fato semelhante ao conversor trifásico dois níveis: o número de vetores no espaço das tensões de fase é o mesmo de vetores no espaço dos estados dos interruptores. Essa implicação incorre em relativa simplicidade ao código no sentido de que o mapeamento dos vetores de linha para os vetores referentes ao estado dos interruptores pode ser realizado de maneira direta.

Observe a Tabela 5.2 que relaciona a tensão de fase em cada braço com corrente i_o que entra no divisor capacitivo da Figura 5.1. É possível verificar, observando o circuito da topologia, que um braço irá interferir na corrente do divisor capacitivo quando a sua tensão de saída for igual a 1. Porém, por se tratar de um sistema a três fios, tem-se que $i_a + i_b + i_c = 0$. Por tanto, com uma breve análise, pode-se concluir que, quando as três fases estiverem com tensão de saída igual a 1, a corrente no divisor capacitivo será igual a zero. De modo semelhante, quando apenas dois braços estiverem sintetizando tensão igual a 1, a corrente no divisor capacitivo será aquela do braço remanescente, porém com sinal positivo.

Tabela 5.1 – Tensão de fase em função dos estados de s_{xa} e s_{xb} , $x \in \{1..3\}$.

s_{xa}	s_{xb}	v_{xg}
0	0	0
0	1	1
1	1	2

Como se pode observar da Tabela 5.2, há vetores redundantes de fase quando observados os vetores de linha. Por exemplo, para o vetor $[v'_{ab} \ v'_{bc}] = [1 \ 0]$, tem-se os vetores de fase de índice 10 e 23. No entanto, para o primeiro, a corrente no divisor capacitivo vale $-i_a$ enquanto, no segundo, i_a . Dessa maneira, cabe, à modulação, escolher o vetor mais adequado que, além de sintetizar a tensão desejada, realize o equilíbrio da tensão nos capacitores. De modo prático, isso implicará em um termo adicional na função

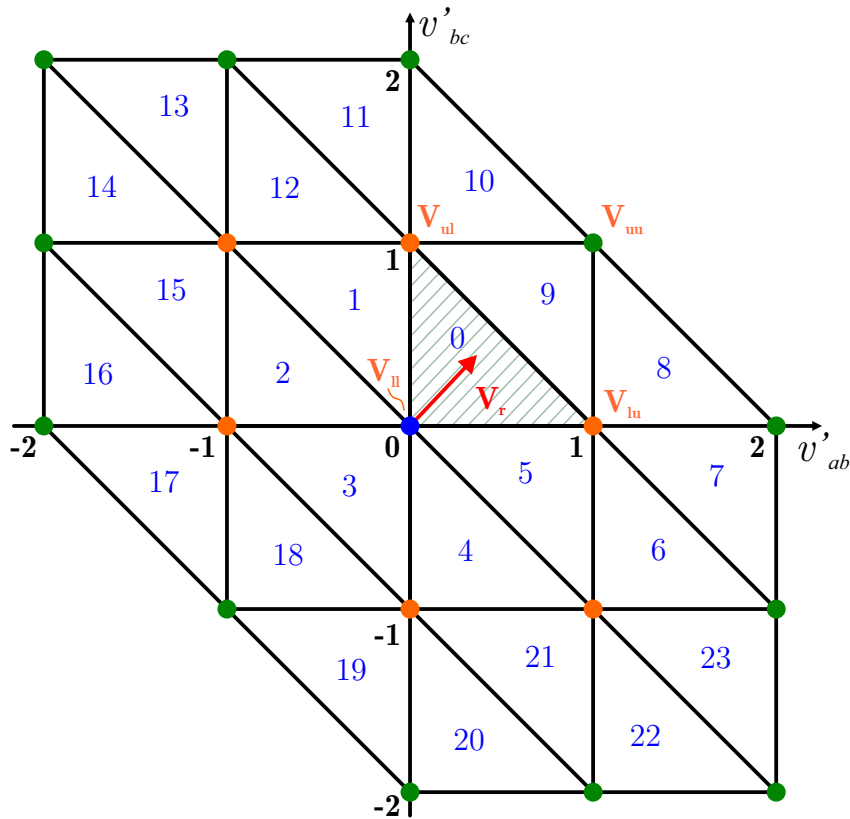
Tabela 5.2 – Mapeamento dos vetores no espaço das tensões de linha para o espaço dos interruptores

Índice	v'_{ag}	v'_{bg}	v'_{cg}	v'_{ab}	v'_{bc}	i_o
1	0	0	0	0	0	0
2	0	0	1	0	-1	$-i_c$
3	0	0	2	0	-2	0
4	0	1	0	-1	1	$-i_b$
5	0	1	1	-1	0	i_a
6	0	1	2	-1	-1	$-i_b$
7	0	2	0	-2	2	0
8	0	2	1	-2	1	$-i_c$
9	0	2	2	-2	0	0
10	1	0	0	1	0	$-i_a$
11	1	0	1	1	-1	i_b
12	1	0	2	1	-2	$-i_a$
13	1	1	0	0	1	i_c
14	1	1	1	0	0	0
15	1	1	2	0	-1	i_c
16	1	2	0	-1	2	$-i_a$
17	1	2	1	-1	1	i_b
18	1	2	2	-1	0	$-i_a$
19	2	0	0	2	0	0
20	2	0	1	2	-1	$-i_c$
21	2	0	2	2	-2	0
22	2	1	0	1	1	$-i_b$
23	2	1	1	1	0	i_a
24	2	1	2	1	-1	$-i_b$
25	2	2	0	0	2	0
26	2	2	1	0	1	$-i_c$
27	2	2	2	0	0	0

custo.

Como consequência, observe que, no diagrama vetorial da Figura 5.2, os vetores mais internos, simbolizados pela cor laranja, apresentam dois vetores redundantes, enquanto que os externos, na cor verde, apresentam apenas um vetor nas tensões de fase. Já o vetor nulo, em azul na origem, apresenta três vetores de fase redundantes.

Figura 5.2 – Diagrama vetorial para o inversor trifásico NPC.



Fonte: Autor.

5.3 ESTRATÉGIA DE MODULAÇÃO PROPOSTA

Em primeiro lugar, nota-se uma diferença crucial em relação aos conversores apresentados no Capítulo 3 e 4. Há a presença de dois capacitores, os quais devem permanecer carregados com a metade do barramento CC. Isto é, além da tensão de saída sintetizada, deve-se levar em conta o balanço nas tensões dos capacitores. Conseqüentemente, a função custo deve contemplar esse quesito na escolha dos vetores. Para tanto, deve-se prever, para cada vetor candidato à escolha, o impacto na tensão dos capacitores a fim de valorar a função custo.

Neste sentido, observa-se a estreita relação da estratégia de modulação proposta com a consolidada MPC aplicada à eletrônica de potência. De modo particular, quanto ao equilíbrio dos capacitores, pode-se dizer que se tem a aplicação da técnica OSV-MPC, descrita no Capítulo 2.

No entanto, deve-se salientar que, apesar das semelhanças entre a técnica de modulação proposta e a técnica MPC serem muitas, trata-se de duas metologias diferentes. A técnica MPC padrão toma por base o modelo físico de um sistema e prevê, para cada vetor, uma variável de interesse no final de um período de amostragem T_s , a exemplo da

corrente de saída, aplicando o resultado em uma função custo. Aquele vetor que minimiza essa função é escolhido para ser aplicado ao conversor.

No entanto, a técnica proposta se concentra apenas em sintetizar uma tensão de saída para uma de referência aos moldes da modulação vetorial, utilizando-se da filosofia de otimização intrincada na técnica MPC por meio de uma função custo. Assim, escolhe-se sempre os três vetores mais próximos no diagrama vetorial, e, dentre esses, escolhe-se aqueles vetores de modo a otimizar a função custo.

Dessa maneira, essa seção traz o desenvolvimento da modulação proposta, estando organizada, respectivamente, nos tópicos de seleção dos três vetores mais próximos, na escolha da sequência de comutação, no mapeamento dos três vetores mais próximos no espaço das tensões de fase e no dos interruptores e na função custo. Posteriormente, na seção subjacente, é apresentado um fluxograma sumarizando todos os passos apresentados neste Capítulo.

5.3.1 Detecção dos três vetores mais próximos

Da mesma maneira com que foi feito nos capítulos anteriores, primeiramente, dado um vetor de referência $\mathbf{v}_r = [v'_{ab} \ v'_{bc}]^T$, arredonda-se suas componentes por meio das funções $\text{floor}(\cdot)$ e $\text{ceil}(\cdot)$, de modo a obter os vetores \mathbf{v}_{ll} , \mathbf{v}_{lu} , \mathbf{v}_{ul} e \mathbf{v}_{uu} , conforme definidos em (4.1). No entanto, pelo fato de os vetores de comutação variarem, em módulo, de 0 a 2, é fácil perceber que o vetor de referência, desde que dado com índice de modulação com a excursão de 0 a 1, necessita ser multiplicado por 2. Dessa maneira, as definições apresentadas em (4.1) podem ser atualizadas para

$$\begin{aligned} \mathbf{v}_{ll} &= [2\text{floor}(v'_{ab}) \quad 2\text{floor}(v'_{bc})]^T \\ \mathbf{v}_{lu} &= [2\text{ceil}(v'_{ab}) \quad 2\text{floor}(v'_{bc})]^T \\ \mathbf{v}_{ul} &= [2\text{floor}(v'_{ab}) \quad 2\text{ceil}(v'_{bc})]^T \\ \mathbf{v}_{uu} &= [2\text{ceil}(v'_{ab}) \quad 2\text{ceil}(v'_{bc})]^T \end{aligned} \quad (5.1)$$

Por se tratar de um conversor trifásico, observa-se que, igualmente ao semelhante dois níveis, deve-se selecionar os três vetores mais próximos que englobam a referência. Portanto, deve-se excluir um dos quatro vetores mais próximos, utilizando, para tanto, o mesmo parâmetro de (4.2). Ou seja, os vetores \mathbf{v}_{lu} e \mathbf{v}_{ul} sempre farão parte dos três vetores. No entanto, caso $R \geq 0$, \mathbf{v}_{rm} , o vetor remanescente, será igual à \mathbf{v}_{uu} . Do contrário, \mathbf{v}_{rm} será igual à \mathbf{v}_{ll} .

Quanto às razões cíclicas, as mesmas empregadas em (4.5) podem ser empregadas

para o caso do NPC, porém multiplicando a referência por 2. Dessa forma, obtém-se

$$\begin{aligned} \begin{bmatrix} d_{lu} \\ d_{ul} \end{bmatrix} &= \begin{cases} \begin{bmatrix} 2v'_{ab} - \text{floor}(2v'_{ab}) \\ 2v'_{bc} - \text{floor}(2v'_{bc}) \end{bmatrix}, & R < 0 \\ \begin{bmatrix} -(2v'_{bc} - \text{ceil}(2v'_{bc})) \\ -(2v'_{ab} - \text{ceil}(2v'_{ab})) \end{bmatrix}, & R \geq 0 \end{cases} \quad (5.2) \\ d_{rm} &= 1 - d_{lu} - d_{ul} \end{aligned}$$

onde

5.3.2 Escolha da sequência de comutação

Após, necessita-se escolher uma sequência de comutação. Para tanto, pode-se escolher a sequência

$$\mathbf{S} = \mathbf{v}_1 \mathbf{v}_2 \mathbf{v}_3 \quad (5.3)$$

Deve-se salientar que a função custo pode ser qualquer uma e que reflete os objetivos do projetista. Neste caso, é adotado que os vetores referidos em (5.3) não assumem uma definição fixa, mas são selecionados pela função custo. Isto é, dado o último vetor aplicado \mathbf{s}_L , haverão três conjuntos (que definem o setor) de vetores de fase onde a função custo será aplicada a cada vetor dentro deles. Aquele vetor que resultar no menor custo, será o escolhido, e o conjunto ao qual pertence (chamado de \mathbf{v}_1) será excluído do universo de busca para o próximo vetor. O procedimento é, então, repetido para os dois conjuntos remanescentes. Dessa maneira \mathbf{v}_1 , \mathbf{v}_2 e \mathbf{v}_3 na sequência (5.3) são totalmente flexíveis, ficando sua definição implícita por meio da função custo.

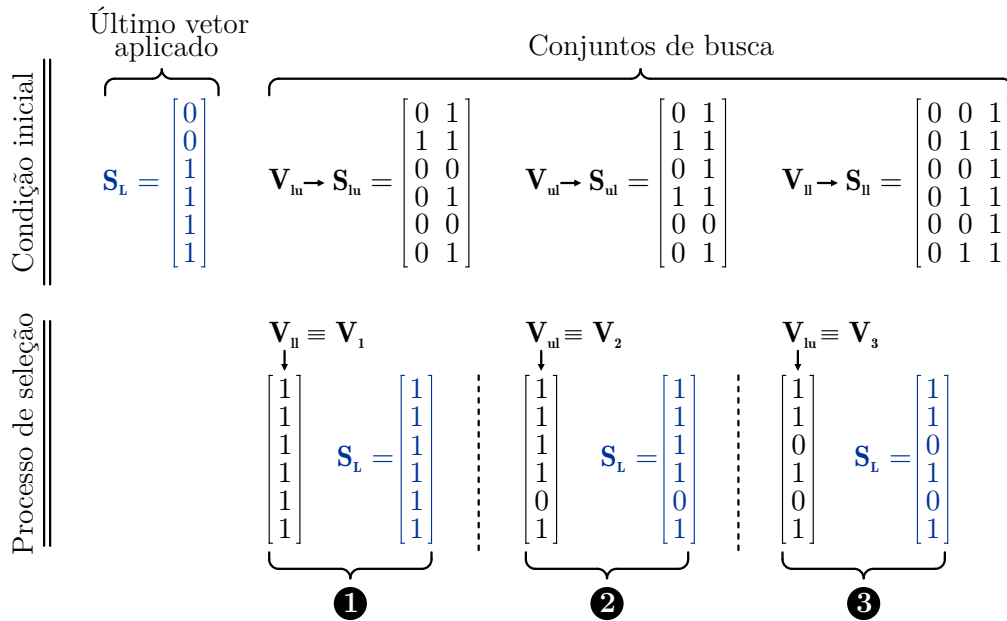
Para fins de clareza, essa lógica é explicada por meio do seguinte exemplo, onde a função custo é dada apenas por

$$g = \|(\mathbf{s}_L - \mathbf{s}_i)\|_1 \quad (5.4)$$

Onde $\mathbf{S}_i = [S_{1a} \ S_{1b} \ S_{2a} \ S_{2b} \ S_{3a} \ S_{3b}]^T$, sendo i um subíndice que enumera os vetores no espaço dos estados dos interruptores para o vetor de linha considerado. A Figura 5.3 indica o fluxo de seleção realizado pelo algoritmo, considerando o último vetor \mathbf{s}_L indicado e um vetor de referência qualquer presente no Setor 0;

Observe que na primeira escolha, na Figura 5.3, o vetor \mathbf{s}_L foi comparado com todos os vetores de todos os conjuntos, em termos da função custo (5.4). Aquele que retornou o menor custo foi o terceiro vetor de \mathbf{s}_{11} , uma vez que, nesse caso, $\|[0 \ 0 \ 1 \ 1 \ 1 \ 1]^T - [1 \ 1 \ 1 \ 1 \ 1 \ 1]^T\|_1 = 2$ retorna o menor valor possível dentre todos os outros vetores. Assim, esse vetor é selecionado e o conjunto \mathbf{s}_{11} está excluído da próxima busca. Para efeitos da

Figura 5.3 – Exemplo para a metodologia de escolha dos vetores que compõem a sequência de comutação.



Fonte: Autor.

escolha do próximo vetor, agora $\mathbf{s}_L = [1 \ 1 \ 1 \ 1 \ 1 \ 1]^T$.

Na sequência, o próximo vetor escolhido é $[1 \ 1 \ 1 \ 1 \ 0 \ 1]^T$, o conjunto \mathbf{s}_{ul} é excluído da busca, e assim por diante. Observe que, para esse caso, $\mathbf{v}_1 = \mathbf{v}_{ll}$, $\mathbf{v}_2 = \mathbf{v}_{ul}$ e $\mathbf{v}_3 = \mathbf{v}_{lu}$. Este cenário poderia ser completamente diferente caso o vetor \mathbf{s}_L , na condição inicial, fosse diferente.

5.3.3 Mapeamento no espaço das tensões de fase e no dos interruptores

Uma vez criada uma estratégia computacional para selecionar os três vetores que compõem o setor, com o cálculo das razões cíclicas e definição da sequência de comutação, deve-se, neste momento, criar uma estratégia para mapear \mathbf{v}_{xy} , $xy \in (l, u) \times (l, u)$ no espaço das tensões de fase Ω e dessa para os vetores no espaço dos estados de comutação dos interruptores \mathbf{s}_x .

Para realizar o mapeamento de \mathbf{v}_x em Ω_x , deve-se observar que existe uma diferença em relação aos conversores monofásico e trifásico dois níveis. Diferentemente dos últimos, o NPC apresenta três vetores redundantes para o vetor nulo. Assim, é impossível se ter uma relação fechada como da forma presente no conjunto de equações (3.6) através dos

truncamentos $\text{floor}(\cdot)$ e $\text{ceil}(\cdot)$. Dessa maneira, deve-se observar que

$$\begin{bmatrix} v_{ab} \\ v_{bc} \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \quad (5.5)$$

Como a matriz que pré-multiplica o lado direito da equação não é quadrada, nota-se que ela não admite matriz inversa. Neste caso específico, isso significa que há infinitas soluções possíveis para v_a , v_b e v_c que satisfazem a equação (5.5) para dado v_{ab} e v_{bc} . Assim, define-se o vetor \mathbf{w} que consiste naquele com a solução para as tensões de fase v_a , v_b e v_c com os menores valores possíveis dentro do conjunto $\{0, 1, 2\}$. Para achar esse vetor, deve-se fazer um laço iterativo, como mostrado abaixo:

```

1 for (va=0; va<=2; va++) {
2   vb=va-vx[0];
3   vc=va-vx[0]-vx[1];
4   if (vb>=0 && vb<=2 && vc>=0 && vc<=2)
5     break;
6 }

```

Onde \mathbf{v}_x é o vetor de linha qualquer que se está sendo mapeado para o espaço das tensões fase. Posteriormente, observa-se que as demais soluções podem ser achadas somando-se consecutivamente à \mathbf{W} vetores com todos elementos igual a 1. Ou seja

$$\mathbf{\Omega}_x = \begin{bmatrix} \mathbf{w} & \mathbf{w} + \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} & \mathbf{w} + \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix} \end{bmatrix} \quad (5.6)$$

Onde x representa o subíndice do vetor de linha qualquer \mathbf{v}_x . Já a função $\text{feasible}(\cdot)$ apresentada em (4.13) para o caso do NPC deve ser expandida para

$$\text{feasible}(\mathbf{U}_{3 \times 2}) = \begin{bmatrix} z_{11} & z_{12} & z_{13} \\ z_{21} & z_{22} & z_{23} \\ z_{31} & z_{32} & z_{33} \end{bmatrix} \quad z_{ij} = \begin{cases} u_{ij}, & u_{ij} = 0 \vee u_{ij} = 1 \vee u_{ij} = 2 \quad \forall i|j \in \{1, 2, 3\} \\ \emptyset, & u_{ij} \neq 0 \wedge u_{ij} \neq 1 \wedge u_{ij} \neq 2 \quad \forall i|j \in \{1, 2, 3\} \end{cases} \quad (5.7)$$

É importante observar que de (5.5), qualquer vetor coluna somado à \mathbf{w} , desde que tenha todos os seus elementos iguais, ainda produzirá as mesmas tensões de linha. No entanto, deve-se verificar se o conversor é capaz de gerar tais níveis de fase. Essa visão permite facilmente expandir o mapeamento para conversores com maior número de níveis.

Uma vez com as tensões de fase, é necessário mapeá-las no espaço dos estados dos

interruptores. Para tanto, a relação 5.8 pode ser usada.

$$\mathbf{s}_{x,i}(\boldsymbol{\Omega}_{x,i}) = \mathbf{M}_s \begin{bmatrix} \text{floor}\left(\frac{1}{2}\boldsymbol{\Omega}_{x,i}\right) \\ \text{ceil}\left(\frac{1}{4}\boldsymbol{\Omega}_{x,i}\right) \end{bmatrix} \quad (5.8)$$

Onde $\boldsymbol{\Omega}_{x,i}$ é um vetor de fase qualquer do conjunto $\boldsymbol{\Omega}_x$ associado ao vetor de linha \mathbf{v}_x , e $\mathbf{s}_{x,i}$ é o vetor no espaço dos estados dos interruptores correspondentes. A matriz auxiliar \mathbf{M}_s é dada por

$$\mathbf{M}_s = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.9)$$

5.3.4 Função custo

Na sequência, é necessário definir uma função custo para nortear a escolha dos vetores de comutação. É necessário, nesta definição, levar em conta a tensão nos capacitores. Portanto, em um primeiro momento, é sugerida a função custo (5.10).

$$g = \|(\mathbf{s}_L - \mathbf{s}_i)\|_1 + \gamma\varepsilon^2, \quad i \in \{0..1\} \quad (5.10)$$

Onde γ é um fator de ponderação e ε é a diferença das tensões v_{C1} e v_{C2} nos capacitores da Figura 5.1 prevista para o i -ésimo vetor candidato \mathbf{S}_i durante seu tempo de aplicação. Quando $v_{C1} - v_{C2} = 0$, significa que as duas tensões encontram-se em equilíbrio. Assim, na função custo (5.10), se para o vetor \mathbf{s}_i , ε^2 for grande, esse vetor terá um custo associado também significativo, tendendo a não ser selecionado frente aos outros vetores.

A fim de avaliar ε , é necessário um modelo do sistema. Esse pode ser facilmente obtido, observando na Figura 5.1 que

$$i_o = i_{C2} - i_{C1} \quad (5.11)$$

Considerando $C_1 = C_2 = C$,

$$i_{C1} = C \frac{dv_{C1}}{dt} \quad i_{C2} = C \frac{dv_{C2}}{dt} \quad (5.12)$$

Fazendo

$$\varepsilon = v_{C2} - v_{C1} \quad (5.13)$$

e, discretizando (5.13) por Euler, tem-se

$$\varepsilon(k_s + 1) = \varepsilon(k_s) + \frac{dT_s}{C} i_o \quad (5.14)$$

Onde k_s se refere aos instantes dentro da janela de amostragem T_s onde se tem as trocas dos vetores de comutação da sequência. Pelo fato de que o tempo da janela associada à k_s depende da razão cíclica d , o horizonte de predição é dada pelo fator dT_s presente na Equação (5.14).

5.4 FLUXOGRAMA

Nesta seção, o fluxograma da Figura 5.4 é apresentado com os passos específicos que o algoritmo de modulação vetorial ótima deve realizar para o caso do conversor NPC. Esse, por sua vez, é dividido em blocos os quais serão explicados detalhadamente na sequência.

BLOCO 1: Primeiramente, é testado se o vetor de referência não satura o conversor. Esse procedimento é análogo para o caso do conversor trifásico. Um detalhe muito importante deve ser esclarecido: apesar dos eixos do diagrama vetorial da Figura 5.2 estar sobre o intervalo de -2 a 2 , o vetor de referência é esperado ter suas componentes entre o intervalo -1 e 1 . Portanto, espera-se uma referência normalizada com respeito ao barramento CC. Essa convenção foi feita para não se perder o significado do índice de modulação. Isto é, uma vez que o vetor de referência seja dado por

$$\mathbf{v}_r = [m_i \sin(\omega t) \quad m_i \sin(\omega t - 2/3\pi)]^T \quad (5.15)$$

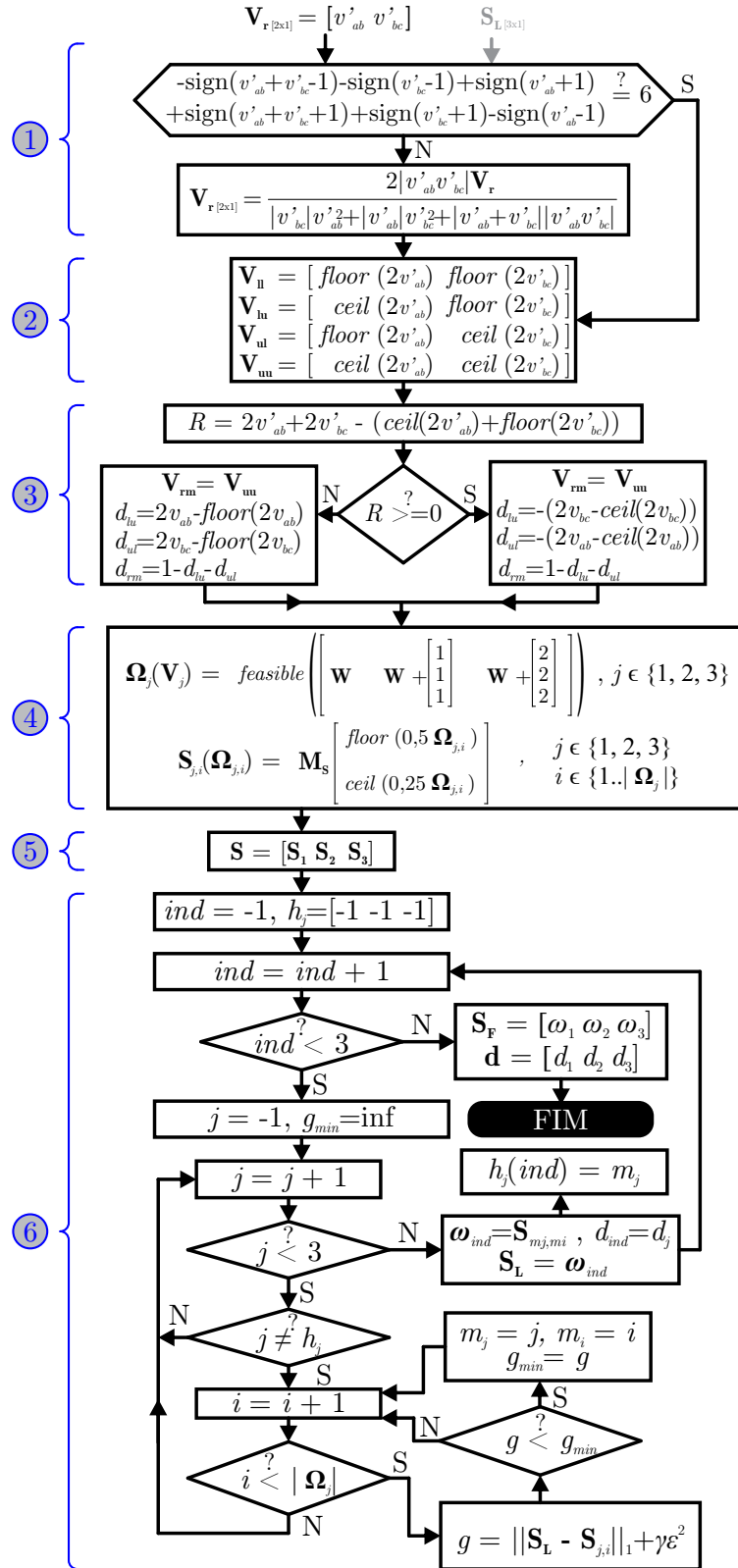
sendo ω a frequência angular e m_i o índice de modulação, sabe-se que $m_i = 1$ implica em 100% da capacidade de síntese para uma tensão senoidal. Dessa maneira, evita-se confusão quanto ao vetor de referência.

Dado o exposto, se o vetor de referência estiver fora dos limites possíveis do conversor sintetizar, o mesmo é saturado de modo que o seu módulo é reduzido até o vetor de referência tocar a borda externa do diagrama da Figura 5.2.

BLOCO 2: Na sequência, os quatro vetores mais próximos são encontrados através dos truncamentos por meio das funções $\text{floor}(\cdot)$ e $\text{ceil}(\cdot)$. No entanto, nota-se que esse truncamento é multiplicado por dois. Neste ponto, fica claro do porquê da decisão de normalizar as tensões de saída com respeito à metade do barramento CC. Uma vez feito dessa forma, os níveis são representados por números inteiros, compatíveis com as técnicas de truncamento.

BLOCO 3: Por se tratar de um espaço vetorial das tensões de saída de dimensão $n = 2$, necessita-se de apenas $n + 1$ vetores de comutação. Portanto, uma vez encontrados

Figura 5.4 – Fluxograma com os passos da rotina de modulação para o inversor trifásico NPC.



Fonte: Autor.

os quatro vetores mais próximos que englobam a referência, deve-se escolher apenas três deles para comporem os vetores de comutação que irão formar a sequência.

Uma rápida inspeção no diagrama da Figura 5.2 mostra que os vetores \mathbf{v}_{lu} e \mathbf{v}_{ul} irão sempre fazer parte dos três vetores mais próximos. Já o terceiro, chamado de \mathbf{v}_{rm} será constituído por \mathbf{v}_{uu} ou \mathbf{v}_{ll} . Para tanto, verifica-se o parâmetro R , expressão da equação da reta que divide a região dos quatro vetores mais próximos em dois triângulos. Se R for positivo, então \mathbf{v}_{r} está acima da reta e é feito $\mathbf{v}_{\text{rm}} = \mathbf{v}_{\text{uu}}$. Do contrário, $\mathbf{v}_{\text{rm}} = \mathbf{v}_{\text{ll}}$.

BLOCO 4: Nesta posição do fluxograma, os três vetores mais próximos já estão definidos. O próximo passo consiste, então, em mapeá-los no espaço das tensões de fase e, após, no espaço dos estados dos interruptores. Para tanto, as funções (5.5) e (5.8) são utilizadas.

Como uma importante observação, deve-se notar que os vetores Ω_x são matrizes em que cada coluna representa um vetor de fase. Dessa forma, o número de colunas pode ser de até três, dependendo do número de redundâncias que o vetor \mathbf{v}_x apresentar. Portanto, as matrizes Ω_x , chamadas também de conjuntos de vetores pelo significado que expressam, não possuem cardinalidade fixa - quando referido por conjuntos, é tido por elemento cada vetor coluna da referida matriz.

Deve-se explicitar que o mapeamento em \mathbf{S} se dá vetor a vetor, e não sobre todo o conjunto. Assim, $\mathbf{s}_{x,i}$ representa o mapeamento no espaço dos interruptores para a tensão de fase referente ao vetor de subíndice x para o i -ésimo vetor do conjunto Ω_x .

Uma das implicações dos vetores Ω_x não apresentarem cardinalidade fixa, é que os vetores em \mathbf{s}_x correspondentes, nos estados dos interruptores, também assim não apresentarão. Por isso, no fluxograma no Bloco 4, o subíndice i assume os valores de 1 até a cardinalidade do conjunto Ω_j .

BLOCO 5: Ocorre a definição da sequência de comutação, reiterando que os vetores \mathbf{s}_1 , \mathbf{s}_2 e \mathbf{s}_3 não assumem uma ordem fixa com respeito ao diagrama vetorial. Do contrário, todos os vetores são definidos pela aplicação da função custo (5.10). Nesse sentido, a sequência de comutação assume o papel de salientar quantos vetores serão aplicados em um período de amostragem.

BLOCO 6: Implementação do algoritmo de seleção dos vetores \mathbf{s}_1 , \mathbf{s}_2 e \mathbf{s}_3 . Para tanto, há três laços de repetição concatenados. Isto é, para cada vetor coluna i do conjunto \mathbf{s}_j , aplica-se a função custo (5.10). Aquele que retornar o menor custo, é escolhido como \mathbf{s}_1 e esse procedimento está associado com a primeira repetição do laço referente à $\text{ind} = 0$. Nesse estágio, é necessário, também, associar à \mathbf{s}_1 a razão cíclica referente ao conjunto de subíndice j .

Em seguida, todos os conjuntos são percorridos novamente, exceto aquele de onde o vetor \mathbf{s}_1 foi escolhido. Por isso, existe o vetor auxiliar h_j , o qual armazena o índice j dos conjuntos \mathbf{s}_j dos quais os vetores ótimos já foram escolhidos, cabendo ao algoritmo desprezar da busca aqueles conjuntos que tenham seus índices registados nesse vetor.

Começa-se com $h_j = [-1 \ -1 \ -1]$ justamente para garantir que, na primeira iteração de *ind*, todos os conjuntos de índice *j* sejam percorridos.

Por último, a sequência final \mathbf{s}_F a aplicar para o conversor trifásico NPC é montada. Em conjunto com ela, se tem um vetor \mathbf{d} com as razões cíclicas associadas. Essa sequência é, então, enviada ao modulador vetorial, o qual implementa os sinais de tensão referentes aos estados dos interruptores de acordo com as razões cíclicas.

Em termos de código, a técnica de modulação pode ser apresentada como exposta a seguir.

```

1 //obtem as medidas das correntes de saída Ia, Ib além das
   tensões dos capacitores vc1 e vc2
2
3 ia = 50.*(0.000732421875*((AdcRegs.ADCRESULT6 >> 4)+10.))-1.5)
   *3.333333333333333;
4 ib = 50.*(0.000732421875*((AdcRegs.ADCRESULT7 >> 4)+25.))-1.5)
   *3.333333333333333;
5
6 vc1 = 200.*(0.000732421875*((AdcRegs.ADCRESULT4 >> 4)+20.))-1.5)
   *3.333333333333333;
7 vc2 = 200.*(0.000732421875*((AdcRegs.ADCRESULT5 >> 4)+10.))-1.5)
   *3.333333333333333;
8
9 //Erro entre os capacitores
10 erro = vc2-vc1;
11
12 ic=-ia-ib;
13
14 //Dicionário da corrente que entra no divisor capacitivo de
   acordo com o índice obtido em função do estado dos
   interruptores
15 Io[0]=0.; Io[1]=-ic; Io[2]=-ib; Io[3]=ia; Io[4]=-ia; Io[5]=ib;
   Io[6]=ic; Io[7]=0.;
16
17 //Geração da referência
18 theta = theta + 2*freq*Ts*pi;
19 if(theta>2*pi)
20     theta = theta - 2*pi;
21
22 vr[0] = indmod*sin(theta);
23 vr[1] = indmod*sin(theta-2./3.*pi);

```

```

24
25 //Encontra os quatro vetores mais próximos
26 vll[0] = floor(2*vr[0]); vll[1] = floor(2*vr[1]);
27 vlu[0] = ceil(2*vr[0]); vlu[1] = floor(2*vr[1]);
28 vul[0] = floor(2*vr[0]); vul[1] = ceil(2*vr[1]);
29 vuu[0] = ceil(2*vr[0]); vuu[1] = ceil(2*vr[1]);
30
31 //Decide, entre os quatro vetores mais próximos, os três que
    definem o setor onde o vetor de referência se encontra
32 R=2*vr[0]+2*vr[1]-(ceil(2*vr[0])+floor(2*vr[1]));
33
34 if(R>=0) {
35     dlu=-(2*vr[1]-ceil(2*vr[1]));
36     dul=-(2*vr[0]-ceil(2*vr[0]));
37     vrm[0]=vuu[0]; vrm[1]=vuu[1];
38 }else{
39     dlu=2*vr[0]-floor(2*vr[0]);
40     dul=2*vr[1]-floor(2*vr[1]);
41     vrm[0]=vll[0]; vrm[1]=vll[1];
42 }
43 dr=1-dlu-dul;
44
45 //Mapeia o vetor vlu, que está no espaço de linha para o espaço
46 //de fases de fase (apenas para o vetor de fase com
47 //os menores valores possíveis)
48 for (va=0; va<=2; va++){
49     vb=va-vlu[0];
50     vc=va-vlu[0]-vlu[1];
51     if(vb>=0 && vb<=2 && vc>=0 && vc<=2)
52         break;
53 }
54
55 W[0][0]=va;
56 W[0][1]=vb;
57 W[0][2]=vc;
58
59 //Mapeia o vetor vul, que está no espaço de linha para o espaço
60 //de fases de fase (apenas para o vetor de fase com
61 //os menores valores possíveis)

```



```
62 for (va=0; va<=2; va++){
63     vb=va-vul[0];
64     vc=va-vul[0]-vul[1];
65     if(vb>=0 && vb<=2 && vc>=0 && vc<=2)
66         break;
67 }
68
69 W[1][0]=va;
70 W[1][1]=vb;
71 W[1][2]=vc;
72
73 //Mapeia o vetor vrm, que está no espaço de linha para o espaço
74 //das tensões de fase (apenas para o vetor de fase com
75 //os menores valores possíveis)
76 for (va=0; va<=2; va++){
77     vb=va-vrm[0];
78     vc=va-vrm[0]-vrm[1];
79     if(vb>=0 && vb<=2 && vc>=0 && vc<=2)
80         break;
81 }
82
83 W[2][0]=va;
84 W[2][1]=vb;
85 W[2][2]=vc;
86
87 //Mapeia todas as tensões de fase possíveis para todos
88 //os vetores de linha
89 Omega[0][0][0]=W[0][0];    Omega[0][1][0]=W[0][0]+1;
90 Omega[0][2][0]=W[0][0]+2;
91 Omega[0][0][1]=W[0][1];    Omega[0][1][1]=W[0][1]+1;
92 Omega[0][2][1]=W[0][1]+2;
93 Omega[0][0][2]=W[0][2];    Omega[0][1][2]=W[0][2]+1;
94 Omega[0][2][2]=W[0][2]+2;
95
96 Omega[1][0][0]=W[1][0];    Omega[1][1][0]=W[1][0]+1;
97 Omega[1][2][0]=W[1][0]+2;
98 Omega[1][0][1]=W[1][1];    Omega[1][1][1]=W[1][1]+1;
99 Omega[1][2][1]=W[1][1]+2;
100 Omega[1][0][2]=W[1][2];    Omega[1][1][2]=W[1][2]+1;
```

```

101 Omega[1][2][2]=W[1][2]+2;
102
103 Omega[2][0][0]=W[2][0];    Omega[2][1][0]=W[2][0]+1;
104 Omega[2][2][0]=W[2][0]+2;
105 Omega[2][0][1]=W[2][1];    Omega[2][1][1]=W[2][1]+1;
106 Omega[2][2][1]=W[2][1]+2;
107 Omega[2][0][2]=W[2][2];    Omega[2][1][2]=W[2][2]+1;
108 Omega[2][2][2]=W[2][2]+2;
109
110 d[0]=dlu; d[1]=dul; d[2]=drm;
111
112 //Mapeia as tensões de fase nos estados dos interruptores
113 for(j=0; j<=2; j++){ //for each vector from sector j
114     for(i=0; i<=2; i++){ //for each leg voltage
115         if(Omega[j][i][0]<=2 && Omega[j][i][1]<=2 &&
116                                     Omega[j][i][2]<=2){
117             S[j][i][0]=floor(0.5*Omega[j][i][0]);
118             S[j][i][1]=ceil(0.25*Omega[j][i][0]);
119             S[j][i][2]=floor(0.5*Omega[j][i][1]);
120             S[j][i][3]=ceil(0.25*Omega[j][i][1]);
121             S[j][i][4]=floor(0.5*Omega[j][i][2]);
122             S[j][i][5]=ceil(0.25*Omega[j][i][2]);
123         }
124         else{
125             //Assegura uma condição impossível para a chave caso
126             //o vetor não seja factível
127             S[j][i][0]=3;
128         }
129     }
130 }
131 //Seleciona os vetores de acordo com o BLOCO 6 do Fluxograma
132 hj[0]=-1; hj[1]=-1; hj[2]=-1;
133 for(ind=0; ind<=2; ind++){
134     gmin=100000;
135     for(j=0; j<=2; j++){
136         if(j!=hj[0] && j!=hj[1] && j!=hj[2]){
137             for(i=0; i<=2; i++){
138                 if(S[j][i][0] != 3){

```

```

139     index=4*(S[j][i][1]-S[j][i][0]) +
140     2*(S[j][i][3]-S[j][i][2]) +
141     (S[j][i][5]-S[j][i][4]);
142     erro_t=erro + d[j]*Ts/C*Io[index];
143     g = (abs(sL[0]-S[j][i][0])+abs(sL[1]-S[j][i][1])+
144     abs(sL[2]-S[j][i][2])+abs(sL[3]-S[j][i][3])+
145     abs(sL[4]-S[j][i][4])+abs(sL[5]-S[j][i][5])) +
146     gama*erro_t*erro_t;
147     if(g<gmin){
148         gmin=g;
149         mj=j; mi= i;
150     }
151 }
152 }
153 }
154 }
155 SF[ind][0]=S[mj][mi][0]; SF[ind][1]=S[mj][mi][1];
156 SF[ind][2]=S[mj][mi][2]; SF[ind][3]=S[mj][mi][3];
157 SF[ind][4]=S[mj][mi][4]; SF[ind][5]=S[mj][mi][5];
158 sL[0]=SF[ind][0]; sL[1]=SF[ind][1]; sL[2]=SF[ind][2];
159 sL[3]=SF[ind][3]; sL[4]=SF[ind][4]; sL[5]=SF[ind][5];
160 df[ind]=d[mj];
161 hj[ind]=mj;
162 index=4*(S[mj][mi][1]-S[mj][mi][0]) +
163 2*(S[mj][mi][3]-S[mj][mi][2]) + (S[mj][mi][5]-S[mj][mi][4]);
164 erro=erro+d[mj]*Ts/C*Io[index];
165 }
166
167 // Aplica a sequência de comutação ao modulador
168 vec1 = 100000*SF[0][5] + 10000*SF[0][4] + 1000*SF[0][3] +
169     100*SF[0][2] + 10*SF[0][1] + SF[0][0];
170 vec2 = 100000*SF[1][5] + 10000*SF[1][4] + 1000*SF[1][3] +
171     100*SF[1][2] + 10*SF[1][1] + SF[1][0];
172 vec3 = 100000*SF[2][5] + 10000*SF[2][4] + 1000*SF[2][3] +
173     100*SF[2][2] + 10*SF[2][1] + SF[2][0];
174 EsvmRegs.DATA.DWT.DW0 = EsvmDwPerc(df[0]*100, 0);
175 EsvmRegs.DATA.DWT.DW1 = EsvmDwPerc(df[1]*100, 1);

```

```

175
176 EsvmRegs.DATA.VEC.W0 = EsvmSequence(vec1);
177 EsvmRegs.DATA.VEC.W1 = EsvmSequence(vec2);
178 EsvmRegs.DATA.VEC.W2 = EsvmSequence(vec3);

```

Como se pode perceber, as linhas 3 à 7 é realizada a medida das variáveis de interesse. Em 10, o erro referente às tensões dos capacitores é calculado. Em 15, é realizado um dicionário para se relacionar a corrente que entra no divisor capacitivo de acordo com a Tabela 5.2. Essa associação é feita considerando a Tabela 5.1, de onde observa que a expressão $|s_{xa} - s_{xb}|$ será igual a um somente para a combinação [0 1]. Nos demais casos, ela será igual a zero.

A relação anterior é uma forma de filtrar aquela combinação que retorna a tensão de saída 1 no braço em questão, uma vez que é nesse caso que se interfere na corrente do divisor capacitivo. Assim, o conjunto da relação mencionada para os três braços do conversor é convertido em um índice que, aplicado ao dicionário da linha 15, obtém-se o valor da corrente que entra no ponto central dos capacitores.

Após, a implementação do fluxograma da Figura 5.4 é implementado de uma maneira direta, aplicando a função custo (5.10) para todos os vetores de todos os conjuntos, excluindo do universo de busca aqueles conjuntos de vetores dos quais os ótimos foram selecionados.

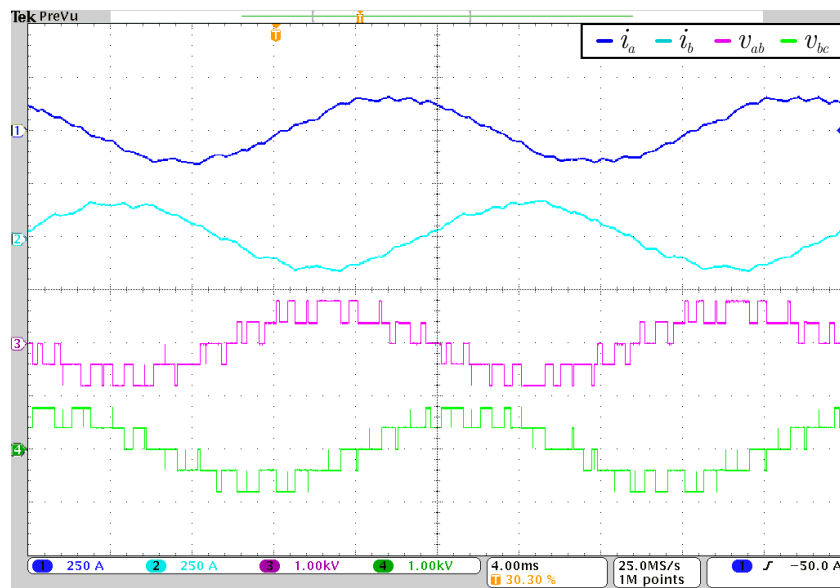
5.5 RESULTADOS EXPERIMENTAIS

Nesta seção, serão mostrados os resultados experimentais para a topologia NPC. Para a obtenção dos resultados, foi utilizado o DSP TEXAS TMS320F28335 em conjunto com a FPGA Xilinx Spartan 3E-500, tendo o Typhoon HIL402 como emulador do conversor, conforme Apêndice A. Para fins de simulação, é considerada, na saída do conversor da Figura 5.1, uma carga trifásica RL, consumindo uma potência de 100 kW, com FP=0,92, barramento CC de 800 V, referência senoidal com índice de modulação igual à $m_i = 0,8$ e capacitores de 47000 μF . As formas de onda de saída encontram-se conforme da Figura 5.5 à 5.8 para uma frequência de amostragem de 1 kHz.

5.5.1 Discussão dos Resultados

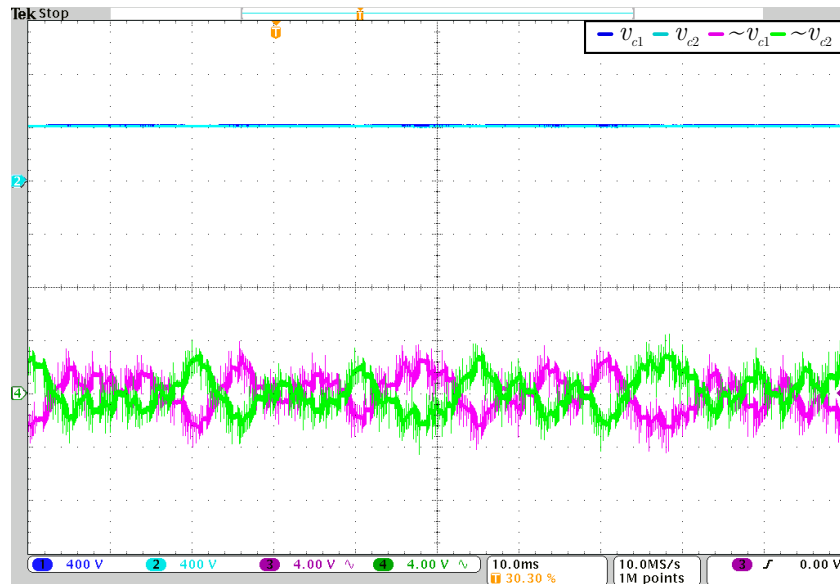
A Figura 5.5 mostra as correntes de linha i_a e i_b da Figura 5.1 para a carga RL, bem como as tensões de linha na saída do conversor. Vê-se que a modulação cumpre o papel de sintetizar uma tensão de saída senoidal, com a geração de cinco níveis. No entanto, se tem a presença de pulsos espúrios nas tensões de saída. Esses pulsos são *glitches* devido à

Figura 5.5 – Correntes de saída e tensões de linha considerando a sequência de comutação (5.3) e a função custo (5.10).



Fonte: Autor.

Figura 5.6 – Tensões sobre os capacitores internos, tanto em acoplamento CC como em CA para evidenciar a pequena oscilação de tensão.

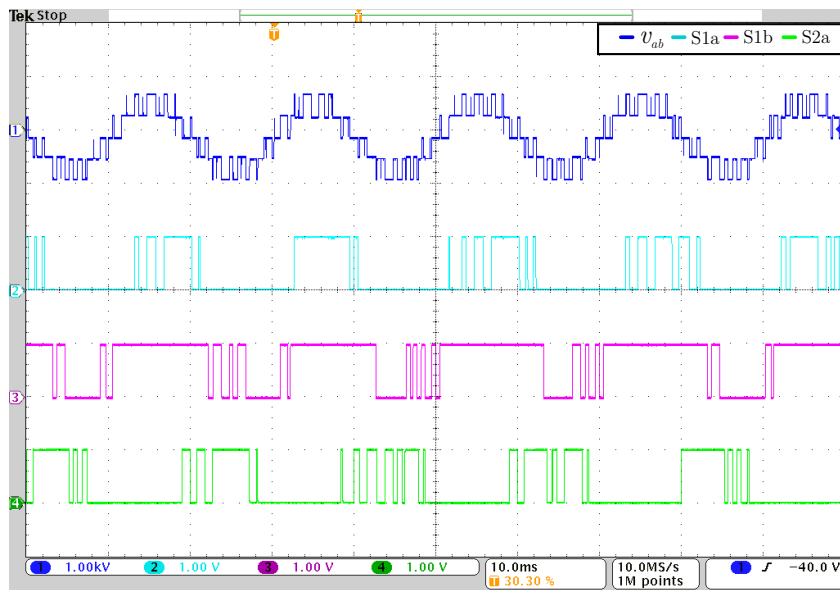


Fonte: Autor.

circuitaria do conjunto DSP, FPGA e TyphoonHIL402.

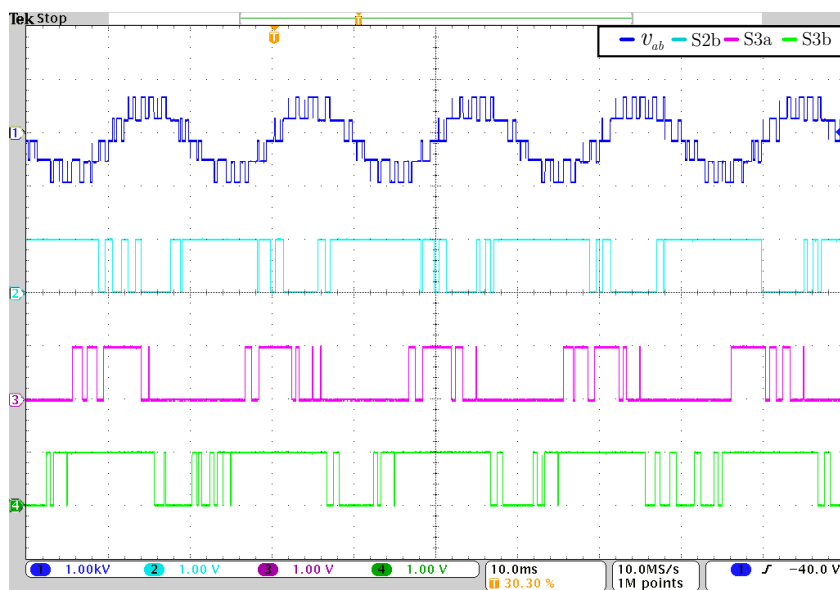
Já a Figura 5.6 mostra as tensões sobre os capacitores. Na parte de cima, elas são mostradas no modo de acoplamento CC enquanto que, na parte de baixo, com acoplamento CA, de mono a evidenciar as conseqüentes flutuações. Desse resultado, depreende-se que cada capacitor permanece carregado com metade (400 V) do barramento CC, com uma

Figura 5.7 – Tensão de linha de saída e distribuição dos pulsos entre os interruptores.



Fonte: Autor.

Figura 5.8 – Tensão de linha de saída e distribuição dos pulsos entre os interruptores.



Fonte: Autor.

ondulação de 4 V.

O padrão dos pulsos nos interruptores estão mostrados nas Figuras 5.7 e 5.8, juntamente com a tensão de linha v_{ab} na parte de cima, de modo a servir como referência de análise para o leitor. Nesse ponto, cabem observações.

A forma com que os pulsos nos interruptores são distribuídos não permanecem fixa no tempo como ocorre no caso dos conversores monofásico e trifásico. Isso acontece em função do processo de regulação das tensões nos capacitores e, ainda, da seleção flexível

dos vetores da sequência de comutação 5.3. Ainda, vê-se que há períodos em que os interruptores permanecem sem comutar. Isso pode ser explicado pelo fato da função custo não incidir sobre a distribuição das comutações dos interruptores.

Por fim, deve-se mencionar a carga computacional envolvida na modulação. Para executar a rotina, o DSP consome um tempo de $700 \mu\text{s}$. Se ainda considerar a execução da rotina de controle, esse tempo pode aumentar. Assim, uma maneira otimizada para implementação deve ser investigada, abrindo possibilidades para trabalhos futuros.

5.6 CONCLUSÃO

Neste capítulo, a modulação *space vector* ótima foi desenvolvida para o caso do inversor trifásico NPC. Como grande diferença em relação aos casos anteriores, encontra-se a necessidade do balanço das tensões dos capacitores da topologia. Neste ponto, a técnica FCS-MPC torna-se mais evidente, fazendo uso de previsões futuras para encontrar o vetor de comutação que tenderá ao equilíbrio das tensões nos capacitores. No entanto, este controle é apenas uma parte da função custo, que seleciona a forma com que os vetores são dispostos na sequência.

6 CONSIDERAÇÕES FINAIS

Esta dissertação começa com um apanhado geral sobre as técnicas de comutação já consolidadas na literatura de modo a estabelecer a problemática e introduzir o propósito deste trabalho. Posteriormente, é explicada a técnica MPC, sobre a qual se baseia a técnica de modulação vetorial proposta, desenvolvendo-a nos capítulos subsequentes para o caso de três conversores considerados.

Em primeiro lugar, pode-se perceber a grande liberdade que traz a modulação proposta, tanto em termos de *hardware* como em questões conceituais. Com ela, toda a implementação ocorre de maneira *online* e é possível otimizar várias questões de operação do conversor, como número de comutações, distribuição dos pulsos, equilíbrio nas tensões dos capacitores, etc.

Trata-se de uma modulação com um princípio de modulação simples. Não são necessários métodos de seleção de vetores e *look-up tables* a serem computadas de modo *offline*. Além disso, a técnica consiste em uma metodologia que pode ser generalizada de modo direto para qualquer outro conversor, praticamente de modo linear. Isto é, a abordagem proposta pode ser facilmente expandida para qualquer conversor.

Além disso, fica evidente que, ao utilizar uma função custo, torna-se relativamente simples a formulação de otimizações multiobjetivo, convertendo, ainda, problemas complexos em outros mais simples, a exemplo de conversores com muitos níveis de saída. Neste quesito, deve-se salientar a grande importância do *hardware* descrito no Apêndice A, que permite ao projetista facilidade e independência na hora de implementar as sequências de comutação desejadas.

No entanto, nota-se que as soluções apresentadas neste documento não são soluções fechadas, mas apenas demonstram uma frente de possibilidades quanto à formulação da sequência de comutação. Isto é, há uma problemática de como escolhê-la. Aquelas apresentadas ao longo do trabalho foram uma primeira abordagem de modo a explorar o tema.

Não somente sobre a forma com que os vetores são dispostos na sequência de comutação se torna intrigante, mas como sobre a maneira com que a função custo é selecionada. É, como qualquer problema de otimização, um desafio para o projetista acomodar todos os objetivos em uma função custo, sendo que trabalhos futuros podem explorar essa questão.

Como se tem o propósito de desenvolver a modulação em si, os conversores não foram considerados no que tange ao controle de variáveis de saída. Assim, assume-se que a tensão de referência vem de um controlador que foge o escopo do tema abordado. Porém, uma consideração quanto a esse tópico deve ser feita no que concerne ao tempo de execução da rotina.

É inegável que as rotinas apresentadas são computacionalmente caras, isto é, de-

mandam grande poder de processamento, resultando em um grande tempo de execução. Se ainda a rotina de controle for considerada, pode ser que não seja viável a implementação de tal técnica de modulação. Assim, uma lacuna encontra-se aberta quanto a métodos otimizados para encontrar vetores que minimizam a função custo.

Outra questão é o custo quanto ao uso de componentes. Para se tornar possível tal modulação, foi necessária uma FPGA além de um DSP. Assim, fica evidenciado que, para conversores de baixa potência, tal técnica talvez não seja economicamente competitiva. No entanto, para conversores de maior potência, onde o preço dos componentes de controle diluem-se frente aos demais custos, a metodologia proposta se torna conveniente, uma vez que se pode atuar de modo direto nas comutações, abrindo a possibilidade de diminuir as perdas associadas a elas e, conseqüentemente, impactar na eficiência dos conversores.

Neste sentido, é possível vislumbrar trabalhos onde se possa quantificar e analisar o impacto do padrão PWM na eficiência do conversor, ou, ainda, como escolher uma seqüência e uma função custo de modo a diminuir a THD da corrente de saída. Ou seja, há a possibilidade da criação de vários fatores de mérito para qualificar os resultados da modulação e antecipar uma seqüência e uma função custo de modo a atingir os objetivos desejados.

Quanto à implementação em *hardware*, observa-se que novas funcionalidades podem ser adicionadas ao modulador, como uma *trip zone* de modo a reagir a uma falta ou para fixar os sinais PWM em valores específicos, como é feito nos módulos ePWM do DSP TMS320F28335. Ainda, opções adicionais de interrupção também podem ser adicionadas. Por exemplo, para gerar uma interrupção toda vez que o contador principal reiniciar determinada quantidade de vezes pré determinadas.

Em suma, contrapõe-se as vantagens de utilizar a metodologia proposta com as desvantagens inerentes da complexidade computacional decorrente. Assim, para o caso de implementação em conversores comerciais, deve-se avaliar o benefício adicional oferecido com os custos associados, de modo a se ter vantagens em relação aos métodos convencionais de modulação. Neste ponto, reitera-se o ponto de vista de que a abordagem proposta se destina a conversores de potência considerável.

REFERÊNCIAS

- A, N.; AGARWAL, P. Nearest and non nearest three vector modulations of npc using two level space vector diagram a novel approach. *IEEE Transactions on Industry Applications*, v. 54, n. 3, p. 2400–2415, May 2018. ISSN 0093-9994.
- ABDULVELEEV, I.; KHRAMSHIN, T.; KORNILOV, G. Space-vector pulse-width modulation of a three-level npc-inverter at low switching frequency. In: **2016 IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (EIconRusNW)**. [S.l.: s.n.], 2016. p. 476–481.
- AHMED, A.; KOH, B.; LEE, Y. A comparison of finite control set and continuous control set model predictive control schemes for speed control of induction motors. *IEEE Transactions on Industrial Informatics*, v. 14, n. 4, p. 1334–1346, April 2018. ISSN 1551-3203.
- BABKRANI, Y. et al. Simulation of cascaded h - bridge multilevel inverter with several multicarrier waveforms and implemented with pd, pod and apod techniques. In: **2017 International Renewable and Sustainable Energy Conference (IRSEC)**. [S.l.: s.n.], 2017. p. 1–6. ISSN 2380-7393.
- BABU, N.; AGARWAL, P. Space vector modulation for three-level npc inverter using two-level space vector diagram. In: **2016 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)**. [S.l.: s.n.], 2016. p. 1–6.
- BENNETT, W. New results in the calculation of modulation products. *The Bell System Technical Journal*, v. 12, n. 2, p. 228–243, April 1933. ISSN 0005-8580.
- BORDONS, C.; MONTERO, C. Basic principles of mpc for power converters: Bridging the gap between theory and practice. *IEEE Industrial Electronics Magazine*, v. 9, n. 3, p. 31–43, Sep. 2015. ISSN 1932-4529.
- BROECK, H.; SKUDELNY, H.; STANKE, G. Analysis and realization of a pulsewidth modulator based on voltage space vectors. *IEEE Transactions on Industry Applications*, v. 24, n. 1, p. 142–150, Jan 1988. ISSN 0093-9994.
- CARRARA, G. et al. A new multilevel pwm method: a theoretical analysis. *IEEE Transactions on Power Electronics*, v. 7, n. 3, p. 497–505, July 1992. ISSN 0885-8993.
- CELANOVICK, N.; BOROYEVICK, D. A fast space vector modulation algorithm for multilevel three-phase converters. In: **Conference Record of the 1999 IEEE Industry Applications Conference. Thirty-Forth IAS Annual Meeting (Cat. No.99CH36370)**. [S.l.: s.n.], 1999. v. 2, p. 1173–1177 vol.2. ISSN 0197-2618.
- CHAMARTHI, P.; CHHETRI, P.; AGARWAL, V. Simplified implementation scheme for space vector pulse width modulation of n-level inverter with online computation of optimal

switching pulse durations. **IEEE Transactions on Industrial Electronics**, v. 63, n. 11, p. 6695–6704, Nov 2016. ISSN 0278-0046.

CHEN, W. K. **The Circuits and Filters Handbook**. [S.l.]: CRC Press, 1995. ISBN 0849383412.

DAI, N.; WONG, M.; HAN, Y. Application of a three-level npc inverter as a three-phase four-wire power quality compensator by generalized 3dsvm. **IEEE Transactions on Power Electronics**, v. 21, n. 2, p. 440–449, March 2006. ISSN 0885-8993.

DE ALMEIDA, R.; LOPES, J.; BARREIROS, J. Improving power system dynamic behavior through doubly fed induction machines controlled by static converter using fuzzy control. **IEEE Transactions on Power Systems**, v. 19, n. 4, p. 1942–1950, Nov 2004. ISSN 0885-8950.

DE CASTRO, L.; CORREA, M.; JACOBINA, C. A fast space vector algorithm for common mode voltage elimination in multilevel converters. In: **2013 Brazilian Power Electronics Conference**. [S.l.: s.n.], 2013. p. 243–247. ISSN 2165-0454.

DESAI, E.; SHAH, N. Comparative analysis of multilevel inverter topologies for photovoltaic system. In: **2013 Nirma University International Conference on Engineering (NUiCONE)**. [S.l.: s.n.], 2013. p. 1–6. ISSN 2375-1282.

DONG, D. et al. Design of hybrid ac-dc-ac topology for uninterruptible power supply. In: **2015 IEEE 2nd International Future Energy Electronics Conference (IFEEC)**. [S.l.: s.n.], 2015. p. 1–5.

DONOSO, F. et al. Finite-set model-predictive control strategies for a 3l-npc inverter operating with fixed switching frequency. **IEEE Transactions on Industrial Electronics**, v. 65, n. 5, p. 3954–3965, May 2018. ISSN 0278-0046.

GEYER, T.; AGUILERA, R.; QUEVEDO, D. On the stability and robustness of model predictive direct current control. In: **2013 IEEE International Conference on Industrial Technology (ICIT)**. [S.l.: s.n.], 2013. p. 374–379.

Geyer, T.; Papafotiou, G.; MORARI, M. Model predictive direct torque control x2014 part i concept algorithm and analysis. **IEEE Transactions on Industrial Electronics**, v. 56, n. 6, p. 1894–1905, June 2009. ISSN 0278-0046.

GEYER, T.; QUEVEDO, D. Multistep finite control set model predictive control for power electronics. **IEEE Transactions on Power Electronics**, v. 29, n. 12, p. 6836–6846, Dec 2014. ISSN 0885-8993.

_____. _____. **IEEE Transactions on Power Electronics**, v. 29, n. 12, p. 6836–6846, Dec 2014. ISSN 0885-8993.

GOMES, A. T. **Telecomunicacoes Transmissao e Recepcao**. 21. ed. Sao Paulo: Editora Erica Ltda, 2008.

GUBIA, E. et al. Ground currents in single-phase transformerless photovoltaic systems. **Progress in Photovoltaics: Research and Applications**, v. 15, n. 7, p. 629–650, 2007. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/pip.761>>.

KARAARSLAN, A. Hysteresis control of power factor correction with a new approach of sampling technique. In: **2008 IEEE 25th Convention of Electrical and Electronics Engineers in Israel**. [S.l.: s.n.], 2008. p. 765–769.

KIEFERNDORF, F. et al. Model predictive control of the internal voltages of a five-level active neutral point clamped converter. In: **2012 IEEE Energy Conversion Congress and Exposition (ECCE)**. [S.l.: s.n.], 2012. p. 1676–1683. ISSN 2329-3721.

KNECHT, O.; BORTIS, D.; KOLAR, J. Comparative evaluation of a triangular current mode (tcm) and clamp-switch tcm dc-dc boost converter. In: **2016 IEEE Energy Conversion Congress and Exposition (ECCE)**. [S.l.: s.n.], 2016. p. 1–8.

KUMAR, A.; CHATTERJEE, D. A survey on space vector pulse width modulation technique for a two-level inverter. In: **2017 National Power Electronics Conference (NPEC)**. [S.l.: s.n.], 2017. p. 78–83.

LARRINAGA, S. et al. Predictive control strategy for dc/ac converters based on direct power control. **IEEE Transactions on Industrial Electronics**, v. 54, n. 3, p. 1261–1271, June 2007. ISSN 0278-0046.

LEE, J. H. Model predictive control: Review of the three decades of development. **International Journal of Control, Automation and Systems**, v. 9, n. 3, p. 415, Jun 2011. ISSN 2005-4092. Disponível em: <<https://doi.org/10.1007/s12555-011-0300-6>>.

MARXGUT, C.; BIELA, J.; KOLAR, J. Interleaved triangular current mode (tcm) resonant transition, single phase pfc rectifier with high efficiency and high power density. In: **The 2010 International Power Electronics Conference - ECCE ASIA -**. [S.l.: s.n.], 2010. p. 1725–1732.

MARXGUT, C. et al. Ultraflat interleaved triangular current mode (tcm) single-phase pfc rectifier. **IEEE Transactions on Power Electronics**, v. 29, n. 2, p. 873–882, Feb 2014. ISSN 0885-8993.

MCGRATH, B.; HOLMES, D. Multicarrier pwm strategies for multilevel inverters. **IEEE Transactions on Industrial Electronics**, v. 49, n. 4, p. 858–867, Aug 2002. ISSN 0278-0046.

_____. _____. **IEEE Transactions on Industrial Electronics**, v. 49, n. 4, p. 858–867, Aug 2002. ISSN 0278-0046.

MIN, B. et al. A novel topology with high efficiency with high efficiency for grid connected photovoltaic pcs. In: **2007 European Conference on Power Electronics and Applications**. [S.l.: s.n.], 2007. p. 1–8.

- MOHAMADIAN, M. et al. Implementation of the first commercial medium power active front end transformerless uninterruptible power supply made in iran. In: **2016 7th Power Electronics and Drive Systems Technologies Conference (PEDSTC)**. [S.l.: s.n.], 2016. p. 217–221.
- MONGE, S. et al. The nearest three virtual space vector pwm - a modulation for the comprehensive neutral-point balancing in the three-level npc inverter. **IEEE Power Electronics Letters**, v. 2, n. 1, p. 11–15, Março 2004. ISSN 1540-7985.
- MOON, H. et al. Mpc-svm method with subdivision strategy for current ripples reduction and neutral-point voltage balance in three-level inverter. In: **2017 IEEE Energy Conversion Congress and Exposition (ECCE)**. [S.l.: s.n.], 2017. p. 191–196.
- MORA, A. et al. Computationally efficient cascaded optimal switching sequence mpc for grid-connected three-level npc converters. **IEEE Transactions on Power Electronics**, p. 1–1, 2019. ISSN 0885-8993.
- MUHAMAD, N.; AZIZ, J. Simulation of power electronic converters with sliding mode control using pspice. In: **2008 IEEE 2nd International Power and Energy Conference**. [S.l.: s.n.], 2008. p. 231–236.
- NAMBOODIRI, A.; WANI, H. Unipolar and bipolar pwm inverter. In: **International Journal for Innovative Research in Science and Technology**. [S.l.: s.n.], 2014. v. 1, n. 7, p. 237–243.
- NETO, D. et al. Investment risk analysis for eolic power plants in the free contracting environment. In: **2015 IEEE International Conference on Industrial Technology (ICIT)**. [S.l.: s.n.], 2015. p. 2783–2788.
- NEUMAYR, D. et al. Novel efficiency-optimal frequency modulation for high power density dc/ac converter systems. In: **2017 IEEE 3rd International Future Energy Electronics Conference and ECCE Asia (IFEEC 2017 - ECCE Asia)**. [S.l.: s.n.], 2017. p. 834–839.
- OGATA, K. **Engenharia de Controle Moderno**. 4. ed. Sao Paulo: Pearson Prentice Hall, 2003. ISBN 9788587918239.
- PANTEN, N.; HOFFMANN, N.; FUCHS, F. Finite control set model predictive current control for grid-connected voltage-source converters with lcl filters: A study based on different state feedbacks. **IEEE Transactions on Power Electronics**, v. 31, n. 7, p. 5189–5200, July 2016. ISSN 0885-8993.
- PAWAR, S.; DESHPANDE, A.; MURALI, M. Modelling and simulation of dstatcom for power quality improvement in distribution system using matlab simulink tool. In: **2015 International Conference on Energy Systems and Applications**. [S.l.: s.n.], 2015. p. 224–227.

QUEVEDO, D.; GOODWIN, G.; DE DONA, J. Finite constraint set receding horizon quadratic control. **International Journal of Robust and Nonlinear Control**, v. 14, n. 4, p. 355–377. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.887>>.

RODRIGUEZ, J. et al. State of the art of finite control set model predictive control in power electronics. **IEEE Transactions on Industrial Informatics**, v. 9, n. 2, p. 1003–1016, May 2013. ISSN 1551-3203.

_____. Predictive current control of a voltage source inverter. **IEEE Transactions on Industrial Electronics**, v. 54, n. 1, p. 495–503, Feb 2007. ISSN 0278-0046.

RYAN, M.; LORENZ, R.; DE DONCKER, R. Modeling of multileg sine-wave inverters: a geometric approach. **IEEE Transactions on Industrial Electronics**, v. 46, n. 6, p. 1183–1191, Dec 1999. ISSN 0278-0046.

SCHNICK, O. et al. New modulation strategy for controlled power electronic applications with low switching frequencies. In: **2008 International Conference on Signals and Electronic Systems**. [S.l.: s.n.], 2008. p. 265–268.

STOJILOVIC, M.; PEJOVIC, P. Performance analysis of space-vector modulated two- and three-level inverters. In: **2012 20th Telecommunications Forum (TELFOR)**. [S.l.: s.n.], 2012. p. 1076–1076.

TAKAHASHI, I.; NOGUCHI, T. A new quick response and high efficiency control strategy of an induction motor. **IEEE Transactions on Industry Applications**, IA-22, n. 5, p. 820–827, Sept 1986. ISSN 0093-9994.

Texas Instruments. **SPRU949D TMS320x2833x 2823x DSC External Interface XINTF**. [S.l.], 2010.

THOMAS, C. et al. A study of energy conversion at the jebba hydroelectric power station. In: **2017 IEEE 3rd International Conference on Electro-Technology for National Development (NIGERCON)**. [S.l.: s.n.], 2017. p. 828–833. ISSN 2377-2697.

TISCHER, C.; SCHERER, L.; DE CAMARGO, R. Voltage and frequency regulation of induction generator based system applying proportional-resonant controller. In: **2015 IEEE 13th Brazilian Power Electronics Conference and 1st Southern Power Electronics Conference (COBEP/SPEC)**. [S.l.: s.n.], 2015. p. 1–6.

VAZQUEZ, S. et al. Predictive optimal switching sequence direct power control for grid-connected power converters. **IEEE Transactions on Industrial Electronics**, v. 62, n. 4, p. 2010–2020, April 2015. ISSN 0278-0046.

_____. Model predictive control for power converters and drives advances and trends. **IEEE Transactions on Industrial Electronics**, v. 64, n. 2, p. 935–947, Feb 2017. ISSN 0278-0046.

WEIBIN, C.; YANRU, Z.; SHUN, J. Combined modulation and harmonic suppression. In: **2006 CES/IEEE 5th International Power Electronics and Motion Control Conference**. [S.l.: s.n.], 2006. v. 1, p. 1–5.

XIE, S. et al. Optimal switching sequence model predictive control for three-phase vienna rectifiers. **IET Electric Power Applications**, v. 12, n. 7, p. 1006–1013, 2018. ISSN 1751-8660.

YEN, K.; TSAI, C.; BAI, M. Performance evaluation of large tilt angle photovoltaic systems in taiwan. In: **2009 34th IEEE Photovoltaic Specialists Conference (PVSC)**. [S.l.: s.n.], 2009. p. 000237–000240. ISSN 0160-8371.

ZHANG, Y.; ZHANG, Q. Relationship between finite control set model predictive control and direct current control for three-phase voltage source converters. In: **2014 International Power Electronics and Application Conference and Exposition**. [S.l.: s.n.], 2014. p. 831–836.

ZHANG, Z. et al. Multiple-vector model predictive power control for grid-tied wind turbine system with enhanced steady-state control performance. **IEEE Transactions on Industrial Electronics**, v. 64, n. 8, p. 6287–6298, Aug 2017. ISSN 0278-0046.

ZHOU, K.; WANG, D. Relationship between space-vector modulation and three-phase carrier-based pwm: a comprehensive analysis [three-phase inverters]. **IEEE Transactions on Industrial Electronics**, v. 49, n. 1, p. 186–196, Feb 2002. ISSN 0278-0046.

APÊNDICES

Apêndice A – MODULADOR VETORIAL - FPGA

Neste apêndice, o modulador vetorial implementado na FPGA Xilinx Spartan 3E-500 será descrito. Este tem por função receber uma sequência de vetores do DSP TEXAS TMS320F28335 em conjunto com as razões cíclicas e gerar os pulsos de saída para os interruptores. Tal modulador, implementado em unidade externa ao DSP, torna possível e simples a implementação de qualquer sequência de comutação, o que por vezes se torna complicado através do uso de *look-up tables* com os módulos ePWM que o DSP fornece.

O DSP TEXAS TMS320F28335 conta com seis módulos ePWM, estendendo por *Enhanced Pulse Width Modulator*. Tais módulos são periféricos que tem por função gerar pulsos PWM através da lógica de comparação de portadoras triangulares ou dente de serra com um sinal modulante. Por se tratarem de periféricos, utilizam o mínimo de carga computacional da CPU (*Central Processing Unit*), realizando operações lógicas de maneira independente, sem interferir na rotina principal.

Através dos módulos ePWM, pode-se implementar rotinas através da modulação vetorial. No entanto, essa tarefa, na maioria das vezes, reduz-se a montar *look-up tables*, tabelas-dicionário onde se associa comparadores (sinais modulantes) para cada setor onde o vetor de referência se encontre. Obviamente, essa é uma tarefa complicada para conversores mais complexos, como os multiníveis. Por tanto, um modulador que implemente qualquer sequência de comutação se torna não só interessante mas como necessário para estender a modulação ótima para qualquer conversor.

Tal modulador, implementado na FPGA, se comunica com o DSP de modo paralelo, modo a se comportar como periférico do último. Para se tornar clara a função do modulador, no DSP, pode-se conseguir a implementação da sequência de comutação através das seguintes linhas de código.

```

1   EsvmRegs.CTLREG.bit.TBPRD = EsvmSampFreq(10000);
2   EsvmRegs.CTLREG.bit.NW = 5;
3
4   EsvmRegs.DATA.DWT.DW0 = EsvmDwPerc(20., 0);
5   EsvmRegs.DATA.DWT.DW1 = EsvmDwPerc(30., 1);
6   EsvmRegs.DATA.DWT.DW2 = EsvmDwPerc(10., 2);
7   EsvmRegs.DATA.DWT.DW3 = EsvmDwPerc(30., 3);
8
9   EsvmRegs.DATA.VEC.W0 = EsvmSequence(101010);
10  EsvmRegs.DATA.VEC.W1 = EsvmSequence(101011);
11  EsvmRegs.DATA.VEC.W2 = EsvmSequence(101111);
12  EsvmRegs.DATA.VEC.W3 = EsvmSequence(101011);
13  EsvmRegs.DATA.VEC.W4 = EsvmSequence(101010);

```

Esse é um exemplo da lógica utilizada para o modulador, do ponto de vista do DSP, para uma sequência simbólica. As duas primeiras linhas são de configuração. A primeira define a frequência de atualização, a segunda, o número de vetores a serem aplicados no intervalo de amostragem, ou seja, o tamanho da sequência de comutação.

Das linhas 4 à 7, se tem a configuração de *dwell times*. Isto é, a parcela de tempo em que se mantém cada um dos vetores especificados nas linhas de número 9 à 13. Esses tempos são dados em percentuais referentes ao período de amostragem. Assim, o vetor W0 permanecerá por 20% do intervalo de amostragem, seguido do vetor W1, o qual permanecerá 30% e assim por diante. Observa-se que não tem, explicitamente, o tempo para o vetor W4. Ocorre que, como a soma das razões cíclicas necessita igualar a um, de uma sequência com N vetores, necessita-se especificar apenas $N - 1$ razões cíclicas, a n -ésima fica definida por consequência.

Logo se observa, do exemplo exposto, a estrutura de dados criada. O registrador Esvm, o qual estende por *Enhanced Space Vector Modulator* indica que será escrito um dado referente ao modulador. Na sequência, pode haver dois campos: CTLREG ou DATA. O primeiro se refere a dados de controle e, o segundo, de informação, tanto referente a razões cíclicas como a vetores.

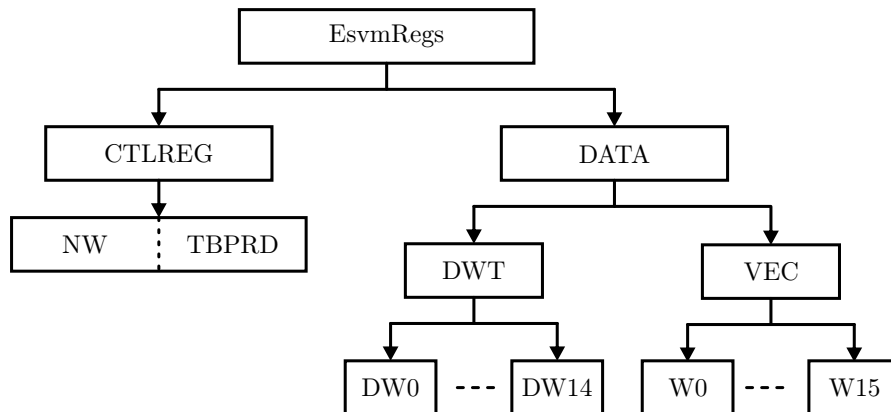
Caso se insira o campo CTLREG, pode-se configurar, da maneira como mostrada no código exemplo, tanto o período de amostragem através do campo TBPRD, como o número de vetores na sequência, por meio do campo NW. Para o primeiro, utiliza-se a função criada EsvmSampFreq(\cdot). Essa função é necessária para o usuário entrar com uma frequência em unidades de Hertz, onde tal número é convertido, através dela, em um dado de comparador para a FPGA.

Por outro lado, se inserir o campo DATA, abre-se duas possibilidades: ou insere-se um *dwell time* DWT ou um vetor VEC. Se for escolhido o primeiro, pode-se configurar as razões cíclicas de DW0 a DW14. Caso se escolha o campo VEC, pode-se configurar os vetores de W0 a W15. Note que a faixa de tempos *dwell time* configuráveis não correspondem ao número de vetores pela dependência decorrente do somatório deles dever ser igual a um. A Figura A.1 sumariza tal estrutura.

Neste momento, cabe ilustrar a arquitetura do modulador, a qual é exposta na Figura A.2. Como se pode observar, há um contador dente de serra (*Mode UP*) de 16 bits que conta até o valor TBPRD. Esse contador apresenta um registrador *Shadow*, o qual garante a atualização de tal valor apenas sob o pulso ESVM_INT. Com essa metodologia, assegura um comportamento síncrono do modulador, evitando que se haja escrita do registrador em momentos inoportunos. Dessa maneira, ao contador atingir o limite determinado, o valor de TBPRD shadow é transferido para o registrador TBPRD active, o qual é de fato utilizado para efeito de comparação do contador.

Além disso, uma vez que o contador atinja o valor de TBPRD, um sinal de interrupção ESVM_INT é gerado, a fim de poder informar um dispositivo externo sobre tal

Figura A.1 – Estrutura de dados para o modulador vetorial.



Fonte: Autor.

estado, com o objetivo de se obter sincronia entre a FPGA e o DSP.

O mesmo sinal do contador é utilizado nos submódulos comparadores de *dwell times*, os quais são responsáveis por gerar sinais que determinarão quanto tempo cada vetor permanecerá no intervalo de amostragem. Os sinais de saída desses comparadores são aplicados ao submódulo Action Qualifier (AQA). Esse submódulo interpreta os sinais dos submódulos comparadores de *dwell times* e os converte em sinais PWM, de acordo com os valores dos registradores ω^0 a ω^3 e com o número de registradores NW informados.

A fim de clarificar o funcionamento do modulador Esvm, o diagrama de tempo da Figura A.3 é apresentado. Do diagrama de tempo, é possível verificar que cada vez que um comparador de *dwell time* DW x , onde x denota o índice de um comparador qualquer dentre os existentes, é atingido pelo contador, um bit do registrador AQA é definido para 1, na ordem com que os comparadores se igualam ao contador. Através do registrador AQA é, então, indexado o vetor ω^x correspondente, onde x denota um dentre os 31 vetores de comutação possíveis.

Dessa maneira, é possível realizar a escrita de um código VHDL para a FPGA, o qual encontra-se abaixo.

```

1 library IEEE;
2 use IEEE.STD_LOGIC_1164.ALL;
3 use IEEE.NUMERIC_STD.all;
4 use ieee.std_logic_misc.all;
5
6 entity main is
7 generic (NBP:integer := 16; -- Número de bits da palavra
8 NR:integer := 63; -- Número total de Registradores
9 NBE:integer := 6; -- Número de bits de endereço
10 NRC:integer := 2; -- Número de registradores de controle

```

```

11   NMW:integer := 31); -- Número máximo de vetores
12
13   -- Barramento de dados
14   Port ( D : in  STD_LOGIC_VECTOR (NBP-1 downto 0);
15         -- Sinal XWE0 do DSP (Habilitação para gravar)
16         E : in  STD_LOGIC;
17         -- Barramento de endereço
18         ADR: in  STD_LOGIC_VECTOR (NBE-1 downto 0);
19         -- Sinal de clock
20         clk: in  STD_LOGIC;
21         -- Sinal de interrupção gerado ao DSP
22         ESVM_INT: out STD_LOGIC;
23         -- Vetor PWM resultante
24         PWM: out  STD_LOGIC_VECTOR(NBP-1 downto 0)
25     );
26   end main;
27
28   architecture Behavioral of main is
29       -- Memória contendo todos os registradores
30       signal M : STD_LOGIC_VECTOR (NR*NBP-1 downto 0);
31       -- Contador
32       signal count: unsigned(NBP-1 downto 0);
33       -- Regist. NOFW: número de vetores definido pelo usuário
34       signal NOFW: unsigned(NBP-1 downto 0);
35       -- Regist. TBPRDs (shadow)
36       signal TBPRDs: unsigned(NBP-1 downto 0);
37       -- Regist. TBPRDa (active): valor em que o contador
38       --reinicia
39       signal TBPRDa: unsigned(NBP-1 downto 0);
40       -- Registradores DWs (shadow)
41       signal DWs: UNSIGNED((NMW-1)*NBP-1 downto 0);
42       -- Registradores DWs (active): comparadores com count
43       signal DWa: UNSIGNED((NMW-1)*NBP-1 downto 0);
44       -- Registradores Ws (shadow)
45       signal Ws: STD_LOGIC_VECTOR(NMW*NBP-1 downto 0);
46       -- Registradores Wa (active): vetores PWM
47       signal Wa: STD_LOGIC_VECTOR(NMW*NBP-1 downto 0);
48       -- Resultado da comparação de DWa com DWs
49       signal AQA: STD_LOGIC_VECTOR ((NMW-1)-1 downto 0);

```

```

50
51 begin
52
53 -- Carrega os valores provenientes do DSP nos registradores
54 --      (conforme mapa de memória)
55
56 TBPRDs <= unsigned(M(0*NBP+(NBP-1) downto 0*NBP));
57 NOFW   <= unsigned(M(1*NBP+(NBP-1) downto 1*NBP));
58
59 DWs((NMW-1)*NBP-1 downto 0) <= unsigned(M((NMW-1+NRC)*NBP-1
60                                           downto NRC*NBP));
61 Ws(NMW*NBP-1 downto 0) <= M((2*NMW+NRC-1)*NBP-1
62                               downto (NMW-1+NRC)*NBP);
63
64 -- Contador e comparadores. O resultado dos comparadores de
65 -- count com DWa é armazenado em AQA.
66 -- Através de AQA, se tem o endereço de memória que apontará
67 -- para um determinado vetor PWM (Wa)
68 process (clk)
69 begin
70 -- contador com máximo valor em TBPRDa. Quando chega em TBPRDa
71 -- é reiniciado gerando a
72 -- interrupção ESVM_INT é gerada, carregando os valores dos
73 -- registradores "SHADOW" para aqueles "ACTIVE"
74 if clk'event and clk = '1' then
75   if(count<TBPRDa) then
76     count <= count + 1;
77     ESVM_INT <= '0';
78   else
79     count <= to_unsigned(0,NBP);
80     ESVM_INT <= '1';
81     TBPRDa <= TBPRDs;
82     DWa <= DWs;
83     Wa <= Ws;
84   end if;
85
86 -- Realiza a gravação quando E (sinal XWE0 do DSP) vai para
87 -- zero (ou seja, se está
88 -- realizando um acesso de escrita)

```

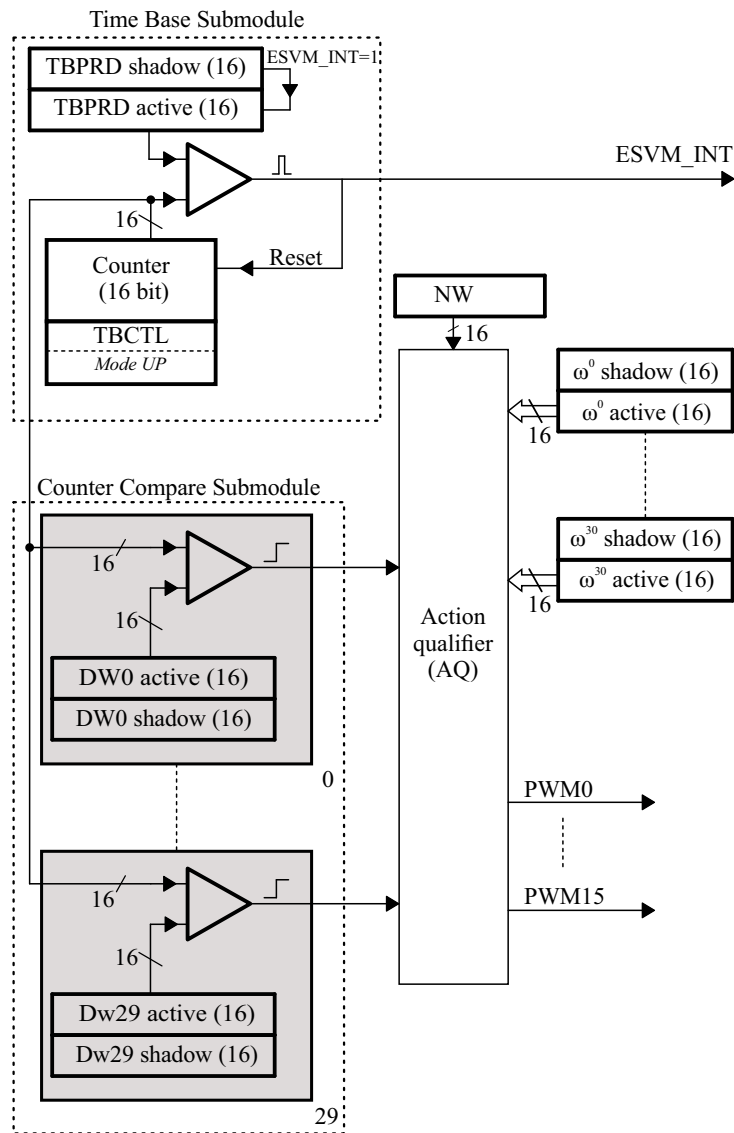
```

82  if E = '0' then
83      M(to_integer(unsigned( ADR ))*NBP+NBP-1 downto
          to_integer(unsigned( ADR ))*NBP) <= D;
84  end if;
85
86  -- Resultado dos comparadores de DWxa com o contador circular
87  for i in 0 to NMW-2 loop
88      if count > DWa(i*NBP+(NBP-1) downto i*NBP) then
89          AQA(i) <= '1';
90      else
91          AQA(i) <= '0';
92      end if;
93  end loop;
94 end if;
95 end process;
96
97 -- Carrega PWM na saída de acordo com AQA - Action Qualifier
98 process (clk)
99 variable nbits1: integer;
100 begin
101     if clk'event and clk = '1' then
102         nbits1 := 0;
103         -- Conta o número de bits '1' no vetor AQA, armazenando-o
104         na variável nbits1
105         for i in 0 to (NMW-2) loop
106             nbits1 := nbits1 + to_integer(unsigned('0' & AQA(i)));
107         end loop;
108         -- De acordo com o valor de nbits1, indexa-se o vetor de
109         comutação a ser aplicado na saída no vetor 'PWM'
110         PWM <= Wa(nbits1*NBP+NBP-1 downto nbits1*NBP);
111     end if;
112 end process;
113
114 end Behavioral;

```

É possível observar, do código exposto, que há uma entidade, a qual representa o modulador com suas entradas e saídas e de uma arquitetura com o comportamento dessa entidade. Como entradas, se tem o barramento de dados, o qual tem 16 bits, e o barramento de endereço, de 6 bits. Em termos de circuitos, cada bit é um sinal elétrico de um barramento, constituindo o conjunto, por tanto, de 22 sinais elétricos. Dessa maneira,

Figura A.2 – Arquitetura interna do modulador vetorial.



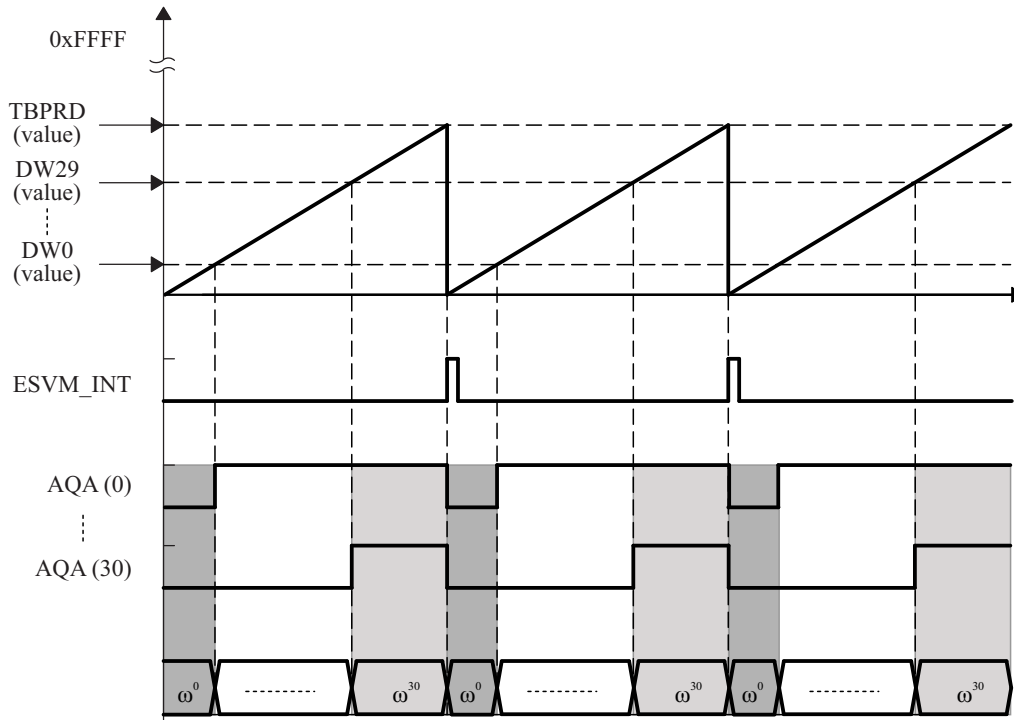
Fonte: Autor.

é possível endereçar $2^6 = 64$ registradores. Dos 64 registradores possíveis, deve-se dividi-los entre registradores de controle e de dados. Dessa maneira, o mapa de memória da Figura A.4, para FPGA, foi considerado.

É importante observar que o mapa da Figura A.4 é implementado através de um vetor na FPGA de 1008 bits, sendo dividido em blocos de bits para corresponder ao mapa de memória. Cada bloco recebe, dessa maneira, uma designação a qual corresponde à posição de memória. Quando o usuário do DSP desejar escrever um dado nessa memória, através do periférico XINTF do DSP (explicado posteriormente) disponibilizará o dado juntamente com o endereço de memória, e assim, a FPGA atualizará os bits no bloco de bits (posição de memória) corretamente.

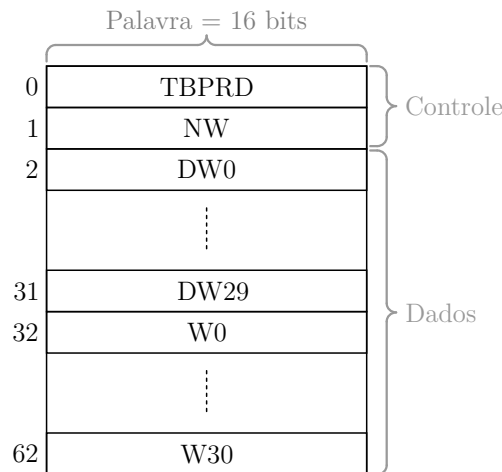
O vetor responsável por desempenhar o papel de memória é o vetor M, o qual é

Figura A.3 – Diagrama de tempo dos sinais envolvidos na geração dos pulsos PWM.



Fonte: Autor.

Figura A.4 – Mapa de memória contendo os registradores do modulador vetorial.



Fonte: Autor.

atualizado apenas quando o bit "E" for igual a zero (linhas 82 a 84). Esse bit, o qual contém o dado do sinal XWE0 do DSP é um bit de habilitação que é jogado para o pino externo toda vez que o DSP estiver escrevendo um dado em uma memória externa. Esse sinal é necessário a fim de evitar que o dispositivo alvo interprete um sinal qualquer do barramento como dados. No entanto, nota-se que essa gravação não é síncrona com o reinício do contador, podendo ocorrer em qualquer momento da contagem, sendo por isso

necessário o uso de registradores *shadow*.

Os registradores *shadow* podem ser observados entre as linhas 56 e 60, onde, de modo concorrente, recebem os valores correspondentes da memória M. Por estarem dispostos dessa forma, recebem os valores de M assim que atualizado. Observando as linhas de número 69 a 78, observa-se um contador que se encontra dentro de um processo sensível ao *clock* (linha 64), executando, portanto, atribuições de modo sequencial. O contador é reiniciado quando atinge TBPRDa, e, quando isso ocorre, não só um sinal de interrupção é gerado - `ESVM_INT = '1'` como também os valores dos registradores *shadow* são transferidos para àqueles *active*, tornando-os síncronos com o início do período de amostragem.

Dentro do mesmo processo, nas linhas de número 87 a 93, realiza-se a comparação do valor do contador com os valores referentes aos *Dwell Times*, conforme o diagrama da Figura A.2. De modo específico, cada vez que um valor de contagem de *Dwell Time* é maior que o valor de contagem, um bit de AQA é levado para 1 e isso é feito para cada um deles, conforme digrama de tempo da Figura A.3. Ao final da comparação de todos os *Dwell Times*, se obterá um vetor AQA com um número de bits "1" igual à ordem do vetor de comutação a aplicar.

Para fins de exemplificação, se `AQA=00001`, aqui representado apenas em cinco bits para fins de simplicidade, significa que se aplicará o primeiro vetor de comutação. Caso `AQA=00011`, será aplicado o segundo vetor de comutação, e assim por diante. Tal processo pode ser pensado como tendo em AQA o endereço de memória do vetor a ser aplicado, estando esse, entretanto, implícito na quantidade de bits '1' a qual ele porta.

No entanto, a ação de traduzir a informação de AQA no vetor de comutação em si a ser aplicado - Módulo *Action Qualifier*, Figura A.2 - dentro do processo presente entre as linhas de número 98 e 110. Nesse processo, sensível ao *clock*, conta-se a quantidade de bits '1' em AQA e armazena-se o resultado na variável `nbits1`. Tal variável, portanto, contém o endereço do vetor de comutação a ser aplicado na saída, o que é feito atribuindo-se tal vetor à porta PWM de 16 bits.

Uma vez com o modulador e a descrição de hardware realizada para o modulador implementado na FPGA, torna-se necessário explicar a respeito do DSP. O TEXAS TMS320F28335 possui um periférico XINTF o qual é responsável pela interface com uma memória externa. Dessa maneira, os sinais de endereço e de dados são, portanto, disponibilizados em seus pinos externos. O XINTF é mapeado em três zonas de memória, chamadas XINTF Zone 0, XINTF Zone 6 e XINTF Zone 7. Isto de modo simples, significa que cada dado escrito pelo processador nessas zonas pode ser acessado externamente.

A cada zona, é associado um sinal de *chip select*, os quais são denominado por $\overline{XZCS0}$, $\overline{XZCS6}$ e $\overline{XZCS7}$, onde o símbolo de barrado significa que são sinais do tipo ativo baixo. Assim, se o processador escrever um dado na zona 0, o sinal correspondente assumirá o valor 0 pelo tempo em que os bits de dado e endereço estiverem disponíveis

nos pinos externos. Ou seja, eles têm como objetivo informar externamente qual zona de memória correspondente à XINTF se está acessando.

Ainda, não somente os anteriores mas também o sinal $\overline{XWE0}$ é também atualizado em operação de escrita, isto é, assume o valor 0 da mesma maneira como aqueles sinais de *chip select*. No entanto esse último tem a função de habilitação, comum a todas as zonas de modo que os dispositivos externos possam distinguir sinais de dados de sinais aleatórios provenientes do barramento de dados e endereço no momento da escrita. Já para a leitura, o sinal de *strobe* (sinal de validação), é o análogo \overline{XRD} . Porém, nesta descrição resumida do periférico XINTF será dada mais ênfase sobre o processo de escrita, uma vez que é a utilizada para o caso do modulador vetorial.

O XINTF fornece a possibilidade de configurar os tempos de escrita e de acesso à memória, de modo a proporcionar a possibilidade de uso de vários dispositivos de armazenamento. Assim, há a configuração dos *wait times*, tempos de espera, os quais são subdivididos em *Lead*, *Active* e *Trail*, podendo ser configurados de maneira independentes para operações de escrita ou de leitura.

O tempo *Lead* corresponde à porção de tempo que o barramento de endereço e o barramento de dados (quando em operação de escrita) permanece no estado desejado antes que o acesso seja efetivamente realizado. Nesse tempo, o sinal de *chip select* associado permanece no estado zero. Já o tempo *Active* se refere ao tempo em que o acesso à memória é de fato realizado. Para a operação de escrita, que é o caso em questão, $\overline{XWE0}$ assume o valor zero, enquanto que para a operação de leitura, é o sinal \overline{XRD} que permanece em zero. Após, durante o tempo *Trail* os referidos sinais de *strobe* voltam ao estado alto, porém mantendo os mesmos sinais de endereço e de dados (para escrita) nos barramentos, além do sinal de *chip select*. Há, ainda, a possibilidade de aumentar esses tempos através do uso do sinal de XREADY. Porém, nesse caso, não se faz necessário.

Os tempos mencionados anteriormente são necessários para respeitar a dinâmica de acesso a uma memória externa e são dados com base no sinal de *clock* XTIMCLK, o qual pode ser tido como igual ou metade de SYSCLKOUT, o *clock* geral do DSP, com frequência de 150 MHz, sendo configurado nos registradores XTIMINGx associados, onde o subíndice x representa a zona associada. Assim, os tempos de espera são dados em ciclos de relógio de XTIMCLK. Se faz importante mencionar, também, que o sinal XTIMCLK é tido como base para gerar o sinal XCLKOUT, que pode ser dado como uma fração do último, servindo como base para os eventos de XINTF.

Há ainda o que mencionar quanto à largura do barramento de dados. Em cada zona, esse pode ser configurado para 16 bits ou para 32 bits. Como para os conversores considerados não há a necessidade de mais que 16 bits e por questão de limitação de pinos disponíveis na FPGA, adotou-se a configuração de 16 bits. Entretanto, caso se deseje expandi-la para conversores com maior número de chaves, as configurações a serem ajustadas não são, em termos de programação, consideráveis.

Neste momento, cobriu-se as características principais do periférico XINTF do DSP TEXAS TMS320F28335 necessárias para o entendimento do que foi realizado quanto ao sistema de modulação vetorial. Em síntese, utiliza-se do XINTF para disponibilizar um barramento de endereço e outro de dados a fim de escrever um dado em um vetor da FPGA que se comporta como memória. Dessa forma, através da rotina de modulação executada no DSP, se consegue gravar dados na FPGA referentes à *Dwell Times* e vetores de comutação, além daquelas informações referentes aos registradores de controle. Para mais detalhes do periférico XINTF, consulte o manual de referência (Texas Instruments, 2010).

A seguir, é mostrado o código no DSP de configuração do XINTF Zone 7, a qual foi escolhida para ser utilizada para a finalidade proposta, bem como é realizado uma operação de escrita à FPGA.

```
1 void init_zone7(void)
2 {
3     // Assegura que o clock de XINTF está habilitado
4     EALLOW;
5     SysCtrlRegs.PCLKCR3.bit.XINTFENCLK = 1;
6     EDIS;
7
8     // Configura GPIO correspondentes à XINTF como um dado de
9     //16-bit (função nata do CodeComposerStudio)
10    InitXintf16Gpio();
11
12    EALLOW;
13    // Configurações aplicáveis a todas as zonas XTINF
14    // Configura XTIMCLK = SYSCLKOUT/2
15    XintfRegs.XINTCNF2.bit.XTIMCLK = 1;
16    //XCLKOUT = XTIMCLK/2 ---> XCLKOUT = SYSCLKOUT/4
17    XintfRegs.XINTCNF2.bit.CLKMODE = 1;
18    // Buffer de até 3 escritas
19    XintfRegs.XINTCNF2.bit.WRBUF = 3;
20    // XCLKOUT é habilitado
21    XintfRegs.XINTCNF2.bit.CLKOFF = 0;
22
23
24    //Configuração da ZONA 7 para os tempos de espera
25    //(escrita apenas)
26    XintfRegs.XTIMING7.bit.XWRLEAD = 3; //Tempo Lead
27    XintfRegs.XTIMING7.bit.XWRACTIVE = 4; //Tempo Active
```

```

28  XintfRegs.XTIMING7.bit.XWRTRAIL = 3;  //Tempo Trail
29
30  //A ZONA 7 não utilizará o sinal XREADY
31  XintfRegs.XTIMING7.bit.USEREADY = 0;
32  XintfRegs.XTIMING7.bit.READYMODE = 0;
33
34  //Configura ZONA 7 para um barramento de dados de 16 bits
35  XintfRegs.XTIMING7.bit.XSIZE = 3;
36
37  EDIS;
38
39  //Força 7 ciclos de espera para assegurar que a escrita ao
40  //último registrador ocorra antes da rotina ser executada,
41  //como recomenda o manual
42  asm(" RPT #7 || NOP");
43 }

```

Uma vez configurada a XINTF, necessita-se criar uma estrutura de dados de modo que, toda vez que se escreva em uma variável de certa classe, o processador registre na posição de memória referente à ZONA 7. Para tanto, é incluída, no cabeçalho de definições gerais do projeto, a seguinte diretiva C:

```

1  #pragma DATA_SECTION(EsvmRegs, "ZONE7DATA");

```

Essa diretiva, da forma com que foi exposta, permite alocar uma variável, vetor ou estrutura a partir de um endereço de memória. Neste caso, de está dizendo que as variáveis da estrutura EsvmRegs começaram a ocupar o primeiro endereço da seção de memória da ZONA 7. Se faz necessário mencionar que o barramento de endereço externo do XINTF se dá de forma relativa, isto é, a posição 0 de memória se refere ao início de cada zona. Dessa maneira, cada variável ocupará o endereço de memória na ordem com que aparecer na estrutura. Para ficar mais claro, considere o caso em que EsvmRegs fosse um vetor de 10 posições com palavras de 16 bits. A palavra do elemento 0 seria alocado na posição 0 de memória da ZONA 7; a do elemento 1, na posição 1 e assim por diante. Por se tratar de um estrutura, e desejando implementar o mapa de memória da Figura A.4, a estrutura em C, através do comando struct, deve ser criada da seguinte forma:

```

1  struct DATA_VEC{
2    Uint16 W0;
3    Uint16 W1;
4    Uint16 W2;
5    Uint16 W3;
6    Uint16 W4;

```

```
7   Uint16 W5;
8   Uint16 W6;
9   Uint16 W7;
10  Uint16 W8;
11  Uint16 W9;
12  Uint16 W10;
13  Uint16 W11;
14  Uint16 W12;
15  Uint16 W13;
16  Uint16 W14;
17  Uint16 W15;
18  Uint16 W16;
19  Uint16 W17;
20  Uint16 W18;
21  Uint16 W19;
22  Uint16 W20;
23  Uint16 W21;
24  Uint16 W22;
25  Uint16 W23;
26  Uint16 W24;
27  Uint16 W25;
28  Uint16 W26;
29  Uint16 W27;
30  Uint16 W28;
31  Uint16 W29;
32  Uint16 W30;
33 };
34
35 struct DATA_DWT{
36   Uint16 DW0;
37   Uint16 DW1;
38   Uint16 DW2;
39   Uint16 DW3;
40   Uint16 DW4;
41   Uint16 DW5;
42   Uint16 DW6;
43   Uint16 DW7;
44   Uint16 DW8;
45   Uint16 DW9;
```

```
46  Uint16 DW10;
47  Uint16 DW11;
48  Uint16 DW12;
49  Uint16 DW13;
50  Uint16 DW14;
51  Uint16 DW15;
52  Uint16 DW16;
53  Uint16 DW17;
54  Uint16 DW18;
55  Uint16 DW19;
56  Uint16 DW20;
57  Uint16 DW21;
58  Uint16 DW22;
59  Uint16 DW23;
60  Uint16 DW24;
61  Uint16 DW25;
62  Uint16 DW26;
63  Uint16 DW27;
64  Uint16 DW28;
65  Uint16 DW29;
66  };
67
68  struct DATA_REG{
69      struct DATA_DWT      DWT;
70      struct DATA_VEC      VEC;
71  };
72
73  struct CTL_REG_BITS{
74      Uint16 TBPRD;
75      Uint16 NW;
76  };
77
78  union CTL_REG{
79      Uint32      all;
80      struct CTL_REG_BITS      bit;
81  };
82
83  struct ESVM_REGS {
84      union CTL_REG      CTLREG;
```



```

85  struct DATA_REG DATA;
86  };

```

Dessa maneira, como a estrutura `EsvmRegs` já está indexada, por meio de cabeçalho de definições, à posição de memória desejada, basta o usuário fazer uso da estrutura definida. Por fim, observa-se do código exemplo mostrado no início deste apêndice, as funções `EsvmSampFreq`, `EsvmDwPerc`, `EsvmSequence`. A primeira tem a função de transformar uma frequência em unidades de Hertz em um valor de comparador. Como, na FPGA, o contador possui 16 bits, o seu valor máximo é de $2^{16} = 65536$. Como o *clock* da FPGA é de 50 MHz, têm-se que a frequência de amostragem *SampFreq* se relaciona com o registrador contador `TBPRD` por

$$SampFreq = \frac{50 \cdot 10^6}{TBPRD}$$

Dessa maneira, a pode-se definir a função `EsvmSampFreq` como segue:

```

1  float EsvmSampFreq(float SFreq) {
2      float result, TBPRD;
3      result = floor(50000000/SFreq);
4      TBPRD = result + floor(50000000/SFreq - result + 0.5);
5      return TBPRD;
6  }

```

Os arredondamentos foram introduzidos para o caso de o usuário inserir uma frequência de amostragem que não seja submúltipla da frequência de contagem de 50 MHz da FGPA.

Já a função a função `EsvmDwPerc` deve associar os *Dwell Times* com comparadores. No entanto, cabe uma questão a considerar. Esses tempos, dados em percentual do período de amostragem, são apenas a parcela de tempo com que certo vetor de comunicação deve permanecer no intervalo de amostragem. No entanto, considerando o diagrama de tempo do modulador vetorial da Figura A.3, fica claro que cada *Dwell Time* deve estar associado a um valor absoluto de comparação em relação a `TBPRD`. Portanto, tal função é implementada da seguinte maneira:

```

1  float EsvmDwPerc(float P, int DwIndex) {
2      float result;
3      volatile Uint16 *p;
4      if(DwIndex != 0) {
5          p = &EsvmRegs.DATA.DWT.DW0+DwIndex-1;
6          result = floor(EsvmRegs.CTLREG.bit.TBPRD*P/100) + *p;
7      }
8      else {
9          result = floor(EsvmRegs.CTLREG.bit.TBPRD*P/100);

```

```

10 }
11 return result;
12 }

```

Basicamente, deve-se informar na função, não somente a parcela de tempo que o vetor de comutação permanecerá, mas também o índice que do registrador que se está alterando, tal como exemplificado no código exemplo no início deste apêndice. Dessa maneira, é acessado, através do ponteiro p o valor do comparador em .DATA.DWT.DWx, onde x se refere ao registrador de *Dwell Time* imediatamente anterior, acrescentando-se a este à parcela de tempo (em termos de valor de comparação com o contador interno da FPGA) referente ao registrador que se está configurando.

Por fim, a função EsvmSequence converte um número binário em um decimal. Deve-se salientar que apesar de ser um número binário, o argumento da função é declarado como inteiro. Portanto, tal função pode ser definida como:

```

1 int EsvmSequence(int64 seq){
2     int64 i, j, DFpga, ResDiv, pot10, pot2;
3
4     ResDiv = seq;
5     DFpga = 0;
6     for (i=0; i<16; i++){
7         pot10=1;
8         for (j=0; j<(15-i); j++){
9             pot10=pot10*10;
10        }
11        pot2=1;
12        for (j=0; j<(15-i); j++){
13            pot2=pot2*2;
14        }
15
16        DFpga = DFpga + floor(ResDiv/pot10)*pot2;
17        ResDiv = ResDiv % pot10;
18    }
19    return (int) DFpga;
20 }

```

Assim, fica descrito o funcionamento do modulador vetorial, desde a sua estrutura conceitual até a sua implementação, tanto por parte da FPGA como do DSP.

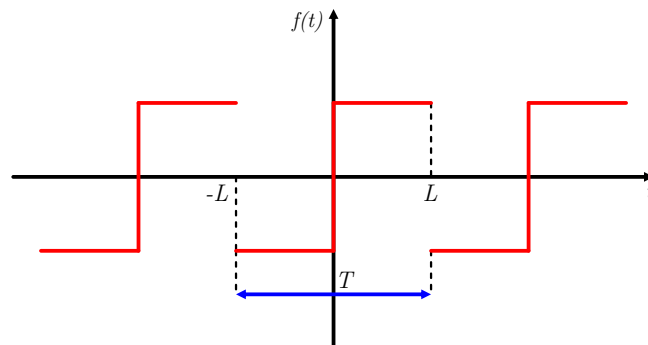
Apêndice B – SÉRIE DUPLA DE FOURIER PARA SINAL PWM

A fim de analisar o espectro harmônico de saída de um sinal PWM obtido a partir da comparação de uma modulante com uma portadora, visando a possibilidade de cancelamento de harmônicos, a expansão em Série de Fourier para uma função com duas variáveis será apresentada (BENNETT, 1933). Neste caso, a função de interesse é a tensão pulsada de saída de um braço de inversor, sendo que a primeira variável é a posição angular do sinal modulante e, a segunda, da portadora com respeito ao sinal modulante.

A Série de Fourier representa uma função periódica através da soma de senos e cossenos de diferentes amplitudes e frequências (GOMES, 2008). Dessa maneira, recorda-se, primeiramente, da expansão sob a forma trigonométrica para uma função de apenas uma variável $f : \mathbb{R} \rightarrow \mathbb{R}$. Sendo ela periódica, conforme mostra a Figura B.1, existe um valor mínimo T , chamado de período, tal que a seguinte condição seja satisfeita:

$$f(t + T) = f(t), \quad \forall t \quad (\text{B.1})$$

Figura B.1 – Função periódica com período T



Fonte: Autor.

A função $f(t)$ pode, então, ser reescrita da seguinte maneira (CHEN, 1995):

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{n\pi t}{L}\right) + b_n \sin\left(\frac{n\pi t}{L}\right) \right] \quad (\text{B.2})$$

Ou, alternativamente, observando que o período $T = 2L$ e que a frequência angular é dada por $\omega = \frac{2\pi}{T}$:

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} [a_n \cos(n\omega t) + b_n \sin(n\omega t)] \quad (\text{B.3})$$

Onde os coeficientes a_0 , a_n e b_n são dados pelas equações (B.7), (B.11) e (B.12), respectivamente, ou de forma alternativa, pelas equações (B.13), (B.14) e (B.15).

Observando a equação (B.3), verifica-se que há uma componente contínua e outras parcelas alternandas, onde a de menor frequência é chamada de fundamental e, as

demais, de harmônicas. Dessa maneira, ω é frequência angular da função trigonométrica fundamental e as constantes a_0 , a_n e b_n são parâmetros a determinar.

Para tanto, observa-se, primeiramente, que a função cosseno é par. Uma função $g : \mathbb{R} \rightarrow \mathbb{R}$ é dita par quando $g(t) = g(-t)$, $\forall t$, ou seja, quando há simetria em relação ao eixo y (eixo das ordenadas). Dessa forma, conclui-se que, para uma função par:

$$\int_{-L}^L g(t)dt = 2 \int_0^L g(t)dt \quad (\text{B.4})$$

Por outro lado, observa-se que a função seno é ímpar. Uma função é dita ímpar quando $g(t) = -g(-t)$, ou seja, quando há simetria em relação à origem. Dessa maneira, conclui-se que:

$$\int_{-L}^L g(t)dt = 0 \quad (\text{B.5})$$

Com as observações acima realizadas, pode-se encontrar os valores das constantes a_0 , a_n e b_n . Se integrarmos os dois lados da equação (B.2) em relação a t de $-L$ a L , observaremos que os termos referente ao seno se cancelarão, reduzindo-se à:

$$\int_{-L}^L f(t)dt = \frac{1}{2} \int_{-L}^L a_0 dt + 2a_n \sum_{n=1}^{\infty} \int_0^L \cos\left(\frac{n\pi t}{L}\right) dt \quad (\text{B.6})$$

No entanto, caso se desenvolva as integrais do somatório do lado direito da equação (B.6), se observará que elas resultarão em zero para qualquer n . Dessa maneira:

$$a_0 = \frac{1}{L} \int_{-L}^L f(t)dt \quad (\text{B.7})$$

Na sequência, deseja-se calcular, para cada n do somatório, o coeficiente a_n correspondente. Para tanto, multiplica-se a equação (B.2) por $\cos\left(\frac{n\pi t}{L}\right)$ e integra-se os dois lados da equação em relação a t de $-L$ a L :

$$\begin{aligned} \int_{-L}^L f(t)\cos\left(\frac{n\pi t}{L}\right) dt &= \int_{-L}^L \frac{a_0}{2}\cos\left(\frac{n\pi t}{L}\right) dt + \\ &\sum_{m=1}^{\infty} \int_{-L}^L \left[a_m \cos\left(\frac{m\pi t}{L}\right) \cos\left(\frac{n\pi t}{L}\right) + b_m \sin\left(\frac{n\pi t}{L}\right) \cos\left(\frac{n\pi t}{L}\right) \right] dt \end{aligned} \quad (\text{B.8})$$

Onde foi introduzida a variável m referente ao índice do somatório a fim de não confundi-lo com o termo de índice n que está sendo obtido. Pode-se observar de (B.8) que o primeiro termo do lado direito da equação se anulará. Já quanto a segunda parcela do somatório, observa-se que se tem uma função seno (ímpar) multiplicada por uma função cosseno (par). O resultado da multiplicação de uma função par por uma ímpar é uma consequente função ímpar. Assim, todos esses termos resultarão em zero. Além disso, é possível provar que:

$$\int_{-L}^L \cos\left(\frac{m\pi t}{L}\right) \cos\left(\frac{n\pi t}{L}\right) dt = 0 \quad , \quad m \neq n \quad (\text{B.9})$$

Dessa maneira, conclui-se que, do lado direito da equação (B.8), restará apenas o primeiro termo do somatório onde $m = n$. Dessa maneira:

$$\int_{-L}^L f(t) \cos\left(\frac{n\pi t}{L}\right) dt = a_n \int_{-L}^L \cos^2\left(\frac{n\pi t}{L}\right) dt \quad (\text{B.10})$$

Portanto:

$$a_n = \frac{1}{L} \int_{-L}^L f(t) \cos\left(\frac{n\pi t}{L}\right) dt \quad (\text{B.11})$$

Pode-se repetir o mesmo procedimento, agora multiplicando a equação (B.2) por $\text{sen}\left(\frac{n\pi t}{L}\right)$ antes de integrar em relação à t de $-L$ a L . De maneira análoga, se obterá:

$$b_n = \frac{1}{L} \int_{-L}^L f(t) \text{sen}\left(\frac{n\pi t}{L}\right) dt \quad (\text{B.12})$$

Equivalentemente, considerando a equação (B.3):

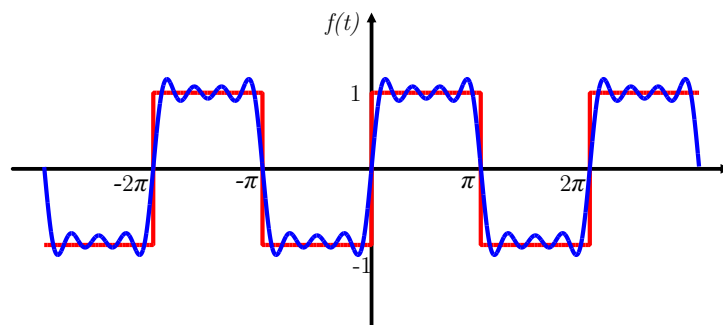
$$a_0 = \frac{\omega}{\pi} \int_{-\frac{\pi}{\omega}}^{\frac{\pi}{\omega}} f(t) dt \quad (\text{B.13})$$

$$a_n = \frac{1}{L} \int_{-\frac{\pi}{\omega}}^{\frac{\pi}{\omega}} f(t) \cos(n\omega t) dt \quad (\text{B.14})$$

$$b_n = \frac{1}{L} \int_{-\frac{\pi}{\omega}}^{\frac{\pi}{\omega}} f(t) \sin(n\omega t) dt \quad (\text{B.15})$$

Onde $\omega = 2\pi f$, sendo f a frequência fundamental do sinal. A Figura B.2 retrata a função quadrada original, juntamente com algumas componentes de frequência da Série de Fourier.

Figura B.2 – Função periódica quadrada e Série de Fourier sobreposta com harmônicas de até 7ª ordem



Fonte: Autor.

É possível verificar que, para uma onda quadrada com amplitude unitária, con-

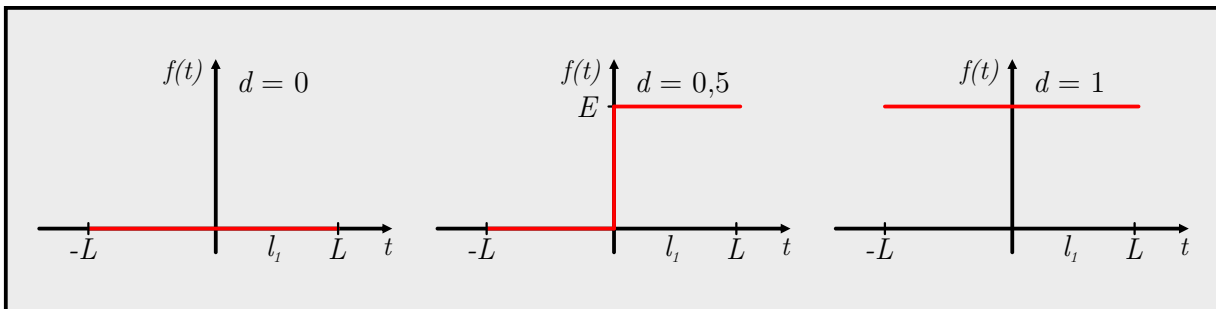
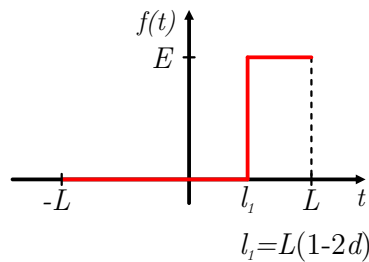
forme mostrada na Figura B.2, os coeficientes da Série de Fourier resultam em:

$$\begin{aligned}
 a_0 &= 0 \\
 a_n &= 0 \\
 b_n &= \frac{2}{n\pi} [1 - \cos(n\pi)]
 \end{aligned}
 \tag{B.16}$$

Desse modo, observa-se que não há harmônicas para índices n pares, uma vez que para esses $b_n = 0$.

Na sequência, deseja-se aplicar a Série de Fourier para um sinal PWM. Este tipo de sinal é muito semelhante a uma onda quadrada, exceto pelo fato de que as larguras de pulsos superiores e inferiores não são simétricas, estando essa diferença associada à razão cíclica d . Dessa maneira, quando $d = 0,5$, temos uma forma de onda quadrada convencional. Quando do contrário, temos uma forma de onda conforme indicada na Figura B.3.

Figura B.3 – Sinal PWM relacionado à razão cíclica d .



Fonte: Autor.

Neste momento, pode-se aplicar as equações (B.7), (B.11) e (B.12) para o sinal PWM representado na Figura B.3, a fim de escrevê-lo da forma como apresentada em (B.3). Dessa maneira, obtém-se:

$$a_0 = 2dE \tag{B.17}$$

$$a_n = \frac{E}{n\pi} [\text{sen}(n\pi(1 - 2d))] \tag{B.18}$$

$$b_n = \frac{E}{n\pi} [\cos(n\pi(1-2d)) - \cos(n\pi)] \quad (\text{B.19})$$

Como esperado, levando a_0 em (B.3), observará que a componente contínua do sinal PWM é justamente o seu valor médio, dado por dE : a razão cíclica d multiplicada pela amplitude E do sinal.

B.1 SÉRIE DUPLA DE FOURIER

Até o momento, a Série de Fourier foi desenvolvida para uma variável. Deseja-se, porém, expandi-la para funções de duas variáveis, obtido por comparação de uma modulante senoidal com uma portadora triangular, o que será válido para estudar o espectro harmônico para sinais PWM, relacionando-o tanto com o sinal modulante (senoidal) como com a portadora (triangular). Considere uma função $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, periódica, tal que:

$$f(x, y) = f(x + T_1, y + T_2) \quad (\text{B.20})$$

Pode-se, então, escrevê-la na forma da Série Dupla de Fourier conforme Equação (B.21)

$$\begin{aligned} f(x, y) = & \frac{A_{00}}{2} + \sum_{n=1}^{\infty} \left[A_{0n} \cos\left(\frac{n\pi y}{L_y}\right) + B_{0n} \cos\left(\frac{n\pi y}{L_y}\right) \right] + \\ & \sum_{m=1}^{\infty} \left[A_{m0} \cos\left(\frac{m\pi x}{L_x}\right) + B_{m0} \cos\left(\frac{m\pi x}{L_x}\right) \right] + \\ & \sum_{m=1}^{\infty} \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} A_{mn} \cos\left(\left(\frac{mx}{L_x} + \frac{ny}{L_y}\right)\pi\right) + B_{mn} \sin\left(\left(\frac{mx}{L_x} + \frac{ny}{L_y}\right)\pi\right) \end{aligned} \quad (\text{B.21})$$

Onde

$$A_{mn} = \frac{1}{2L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \cos\left(\left(\frac{mx}{L_x} + \frac{ny}{L_y}\right)\pi\right) dx dy \quad (\text{B.22})$$

$$B_{mn} = \frac{1}{2L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \sin\left(\left(\frac{mx}{L_x} + \frac{ny}{L_y}\right)\pi\right) dx dy \quad (\text{B.23})$$

As equações (B.21) à (B.23) podem ser obtidas naturalmente a partir da expansão da Série de Fourier de uma variável, como será agora mostrado. Se, na função $f(x, y)$, se considerar y constante, se obterá uma função apenas em x , reduzindo-a ao caso de apenas

uma variável:

$$f(x) = \left[\frac{a_0(y)}{2} + \sum_{m=1}^{\infty} a_m(y) \cos\left(\frac{m\pi x}{L_x}\right) + b_m \sin\left(\frac{m\pi x}{L_x}\right) \right]_{y=y_0} \quad (\text{B.24})$$

Nota-se que os coeficientes da série se tornam função de y , pois, nas expressões (B.7), (B.11) e (B.12), sendo $f(x, y)$ uma função de duas variáveis, a integração para obtenção dos termos a_0 , a_n e b_n resultará em uma expressão dependente de y :

$$a_0(y) = \frac{1}{L_x} \int_{-L_x}^{L_x} f(x, y) dx \quad (\text{B.25})$$

$$a_n(y) = \frac{1}{L_x} \int_{-L_x}^{L_x} f(x, y) \cos\left(\frac{m\pi x}{L_x}\right) dx \quad (\text{B.26})$$

$$b_n(y) = \frac{1}{L_x} \int_{-L_x}^{L_x} f(x, y) \sin\left(\frac{m\pi x}{L_x}\right) dx \quad (\text{B.27})$$

Uma vez observados os coeficientes das equações (B.25), (B.26) e (B.27), deseja-se agora expandi-los, cada um, também como uma Série de Fourier. Ou seja, cada coeficiente em função de y é tido como a função $f(y)$ a expandir. Dessa maneira, fazendo:

$$a_0(y) = \frac{a_0 a_0}{2} + \sum_{n=1}^{\infty} a_0 a_n \cos\left(\frac{n\pi y}{L_y}\right) + a_0 b_n \cos\left(\frac{n\pi y}{L_y}\right) \quad (\text{B.28})$$

$$a_m(y) = \frac{a_m a_0}{2} + \sum_{n=1}^{\infty} a_m a_n \cos\left(\frac{n\pi y}{L_y}\right) + a_m b_n \sin\left(\frac{n\pi y}{L_y}\right) \quad (\text{B.29})$$

$$b_m(y) = \frac{b_m a_0}{2} + \sum_{n=1}^{\infty} b_m a_n \cos\left(\frac{n\pi y}{L_y}\right) + b_m b_n \sin\left(\frac{n\pi y}{L_y}\right) \quad (\text{B.30})$$

Obterá-se, de acordo com as equações (B.7), (B.11) e (B.12):

$$a_0 a_0 = \frac{1}{L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) dx dy \quad (\text{B.31})$$

$$a_0 a_n = \frac{1}{L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \cos\left(\frac{n\pi y}{L_y}\right) dx dy \quad (\text{B.32})$$

$$a_0 b_n = \frac{1}{L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \sin\left(\frac{n\pi y}{L_y}\right) dx dy \quad (\text{B.33})$$

$$a_m a_0 = \frac{1}{L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \cos\left(\frac{m\pi x}{L_x}\right) dx dy \quad (\text{B.34})$$

$$a_m a_n = \frac{1}{L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \cos\left(\frac{m\pi x}{L_x}\right) \cos\left(\frac{n\pi y}{L_y}\right) dx dy \quad (\text{B.35})$$

$$a_m b_n = \frac{1}{L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \cos\left(\frac{m\pi x}{L_x}\right) \sin\left(\frac{n\pi y}{L_y}\right) dx dy \quad (\text{B.36})$$

$$b_m a_0 = \frac{1}{L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \sin\left(\frac{m\pi x}{L_x}\right) dx dy \quad (\text{B.37})$$

$$b_m b_n = \frac{1}{L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \sin\left(\frac{m\pi x}{L_x}\right) \cos\left(\frac{n\pi y}{L_y}\right) dx dy \quad (\text{B.38})$$

$$b_m b_n = \frac{1}{L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \sin\left(\frac{m\pi x}{L_x}\right) \sin\left(\frac{n\pi y}{L_y}\right) dx dy \quad (\text{B.39})$$

Levando as equações (B.31) a (B.39) em (B.24), fica fácil perceber que a última pode ser escrita conforme a equação (B.40):

$$\begin{aligned} f(x, y) = & \frac{A_{00}}{2} + \sum_{n=1}^{\infty} \left[A_{0n} \cos\left(\frac{n\pi y}{L_y}\right) + B_{0n} \sin\left(\frac{n\pi y}{L_y}\right) \right] + \\ & \sum_{m=1}^{\infty} \left[A_{m0} \cos\left(\frac{m\pi x}{L_x}\right) + B_{m0} \sin\left(\frac{m\pi x}{L_x}\right) \right] + \\ & \sum_{m=1}^{\infty} \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} \left[A_{mn} \cos\left(\left(\frac{mx}{L_x} + \frac{ny}{L_y}\right)\pi\right) + B_{mn} \sin\left(\left(\frac{mx}{L_x} + \frac{ny}{L_y}\right)\pi\right) \right] \end{aligned} \quad (\text{B.40})$$

Sendo

$$A_{mn} = \frac{1}{2L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \cos\left(\left(\frac{mx}{L_x} + \frac{ny}{L_y}\right)\pi\right) dx dy \quad (\text{B.41})$$

$$B_{mn} = \frac{1}{2L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) \sin\left(\left(\frac{mx}{L_x} + \frac{ny}{L_y}\right)\pi\right) dx dy \quad (\text{B.42})$$

Ou, na forma complexa:

$$A_{mn} + jB_{mn} = \frac{1}{2L_x L_y} \int_{-L_y}^{L_y} \int_{-L_x}^{L_x} f(x, y) e^{j\left(\left(\frac{mx}{L_x} + \frac{ny}{L_y}\right)\pi\right)} dx dy \quad (\text{B.43})$$

Para o caso particular onde $L_x = L_y = \pi$:

$$f(x, y) = f(x + 2\pi, y + 2\pi) \tag{B.44}$$

$$f(x, y) = \frac{A_{00}}{2} + \sum_{n=1}^{\infty} [A_{0n} \cos(ny) + B_{0n} \sin(ny)] + \sum_{m=1}^{\infty} [A_{m0} \cos(mx) + B_{m0} \sin(mx)] + \sum_{m=1}^{\infty} \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} [A_{mn} \cos(mx + ny) + B_{mn} \sin(mx + ny)] \tag{B.45}$$

$$A_{mn} + jB_{mn} = \frac{1}{2\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} f(x, y) e^{j(mx+ny)} dx dy \tag{B.46}$$

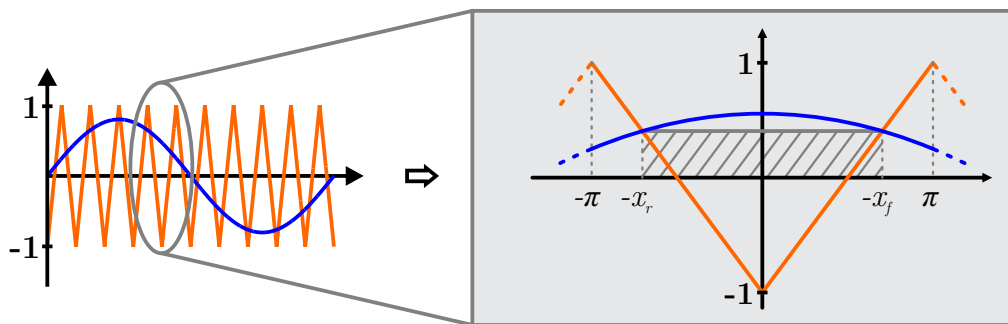
O sinal PWM é uma função do ângulo da portadora e do ângulo do sinal modulante, uma vez que o sinal é 1 quando o sinal modulante é maior que a portadora. Portanto, considera-se x como sendo a posição angular da portadora e y como a do sinal modulante:

$$x = \omega_c t + \theta_c \tag{B.47}$$

$$y = \omega_0 t + \theta_0 \tag{B.48}$$

Para obter os coeficientes da série de Fourier, é necessário determinar os instantes onde ocorre a transição do sinal PWM. Observe a Figura B.4.

Figura B.4 – Instante de comutação do sinal modulante em relação à portadora.



Fonte: Autor.

Definindo a função da portadora como:

$$f(x) = \begin{cases} \frac{-2}{\pi}x - 1 & -\pi \leq x \leq 0 \\ \frac{2}{\pi}x - 1 & 0 \leq x \leq \pi \end{cases} \tag{B.49}$$

E o sinal modulante como:

$$f(y) = m_a \cos(y) \quad (\text{B.50})$$

Os instantes x_r e x_f , são, respectivamente:

$$x_r = -\frac{\pi}{2} (1 + m_a \cos(y)) \quad (\text{B.51})$$

$$x_f = \frac{\pi}{2} (1 + m_a \cos(y)) \quad (\text{B.52})$$

Consequentemente:

$$A_{mn} + jB_{mn} = \frac{1}{2\pi^2} \int_{-\pi}^{\pi} \int_{x_r}^{x_f} E e^{j(mx+ny)} dx dy \quad (\text{B.53})$$

Aplicando B.51 e B.52 em B.53:

$$A_{mn} + jB_{mn} = \frac{E}{2\pi^2} \int_{-\pi}^{\pi} \int_{-\frac{\pi}{2}(1+m_a \cos(y))}^{\frac{\pi}{2}(1+m_a \cos(y))} e^{j(mx+ny)} dx dy \quad (\text{B.54})$$

Primeiramente, obtém-se os coeficientes A_{00} e B_{00} :

$$A_{00} + jB_{00} = \frac{E}{2\pi^2} \int_{-\pi}^{\pi} (1 + m_a \cos(y)) dy \quad (\text{B.55})$$

$$A_{00} + jB_{00} = E \quad (\text{B.56})$$

Dessa maneira:

$$\begin{aligned} A_{00} &= E \\ B_{00} &= 0 \end{aligned} \quad (\text{B.57})$$

Na sequência, pode-se fazer $m = 0$ e determinar a amplitude das componentes de frequência devido ao sinal modulante, incluindo a fundamental bem como as suas múltiplas:

$$A_{0n} + jB_{0n} = \frac{E}{2\pi^2} \int_{-\pi}^{\pi} \int_{-\frac{\pi}{2}(1+m_a \cos(y))}^{\frac{\pi}{2}(1+m_a \cos(y))} e^{jny} dx dy \quad (\text{B.58})$$

$$A_{0n} + jB_{0n} = \frac{E}{2\pi} \int_{-\pi}^{\pi} e^{jny} (1 + m_a \cos(y)) dy \quad (\text{B.59})$$

Expandindo o termo exponencial e realizando a multiplicação, obtém-se:

$$A_{0n} + jB_{0n} = \frac{E}{2\pi} \int_{-\pi}^{\pi} \{ \cos(ny) + m_a \cos(ny) \cos(y) + j [\sin(ny) + m_a \cos(y) \sin(ny)] \} dy \quad (\text{B.60})$$

Observando a equação B.60, e lembrando que a integral sobre o período do produto de duas funções ortogonais é nula, vê-se que o resultado da integral será nulo para todo n , exceto para $n = 1$, para o qual:

$$A_{01} + jB_{01} = \frac{E}{2\pi} \int_{-\pi}^{\pi} m_a \cos^2(y) dy \quad (\text{B.61})$$

Consequentemente:

$$\begin{aligned} A_{01} &= m_a \frac{E}{2} & n &= 1 \\ A_{0n} &= 0 & n &> 1 \\ B_{0n} &= 0 & n &\geq 1 \end{aligned} \quad (\text{B.62})$$

Ou seja, o sinal PWM apresenta uma componente na frequência do sinal modulante, o que já era esperado analisando a forma com que foi gerado. No entanto, nota-se também que o sinal não apresenta componentes nas frequências múltiplas do sinal modulante.

Uma vez feito isso, vamos fazer $n = 0$ e calcular os coeficientes das componentes de frequência da portadora, incluindo sua fundamental e suas múltiplas. Da equação (B.43):

$$A_{m0} + jB_{m0} = \frac{E}{2\pi^2} \int_{-\pi}^{\pi} \int_{-\frac{\pi}{2}(1+m_a \cos(y))}^{\frac{\pi}{2}(1+m_a \cos(y))} e^{jmx} dx dy \quad (\text{B.63})$$

Fornecendo

$$A_{m0} + jB_{m0} = \frac{E}{2\pi^2} \int_{-\pi}^{\pi} \frac{e^{jm[\frac{\pi}{2}(1+m_a \cos(y))]} - e^{-jm[\frac{\pi}{2}(1+m_a \cos(y))]}}{jm} dy \quad (\text{B.64})$$

Pode ser provado que a resolução para a equação (B.64) pode ser dada por (Humberto, p. 23):

$$\begin{aligned} A_{m0} &= \frac{2EJ_0 \left(m \frac{\pi}{2} m_a \right) \sin \left(m \frac{\pi}{2} \right)}{m\pi} \\ B_{m0} &= 0 \end{aligned} \quad (\text{B.65})$$

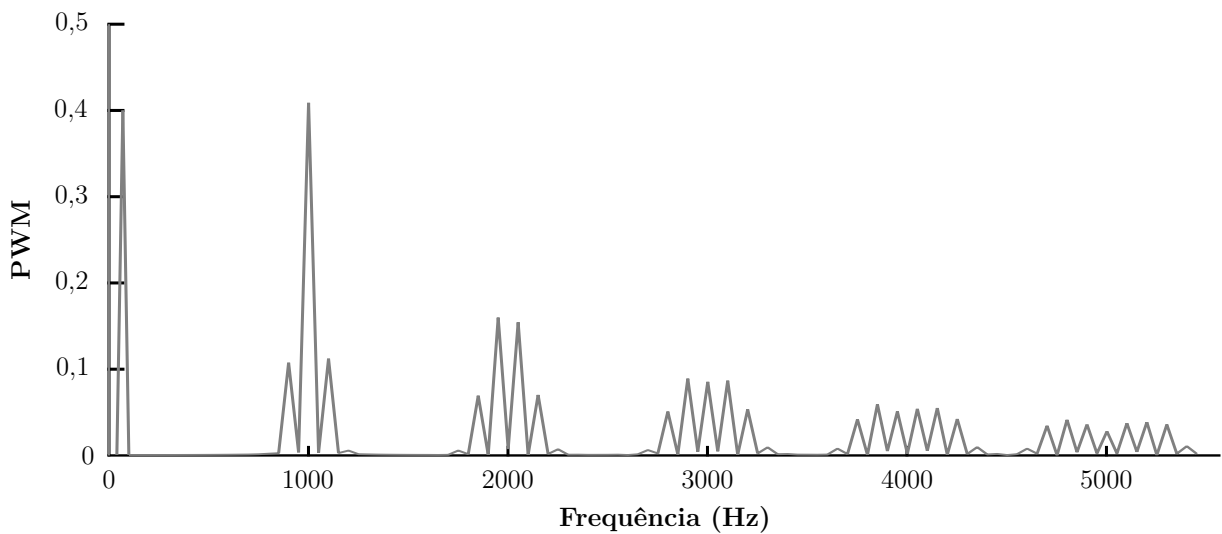
Onde J_0 é a função de Bessel do tipo um e ordem zero. Analisando o termo em que aparece o seno na equação (B.65), fica evidente que não há harmônicas nas frequências múltiplas pares da portadora. Agora resta apenas os termos para m e n diferentes de

zero. Pode ser provado que:

$$\begin{aligned} A_{m0} &= \frac{2EJ_n\left(m\frac{\pi}{2}m_a\right)\sin\left((m+n)\frac{\pi}{2}\right)}{m\pi} \\ B_{m0} &= 0 \end{aligned} \quad (\text{B.66})$$

Onde J_0 é a função de Bessel do tipo um e ordem n . A figura B.5 apresenta um espectro harmônico para um sinal modulante $v_{ag} = 0.8\cos(2\pi 50t)$ e $\omega_c = 2\pi 1000\text{rad/s}$ e $\theta_c = 0$.

Figura B.5 – Espectro harmônico para um sinal PWM obtido através da comparação de uma modulante senoidal com uma portadora triangular.



Fonte: Autor.

Como se pode observar, há uma componente espectral em 50Hz referente ao sinal modulante, sendo a componente de interesse a sintetizar na saída do conversor. Observa-se, também, componentes na frequência da portadora e suas múltiplas ímpares. Em torno das frequências múltiplas da portadora, nota-se componentes nas bandas laterais distanciadas entre si pela frequência do sinal modulante. É possível concluir, ainda, que as componentes das bandas laterais se tornam desprezíveis a partir da quarta ordem.

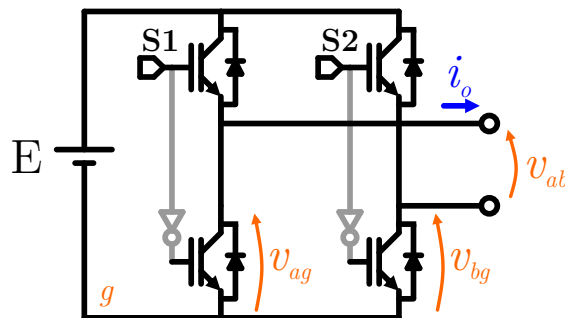
A partir da investigação do conteúdo harmônico para um sinal PWM gerado por comparação de um sinal modulante com uma portadora triangular, com a obtenção de uma forma matemática fechada para as componentes espectrais, torna-se viável a investigação de configurações que visem o cancelamento de harmônicos. É importante mencionar que filtros, na maioria das vezes passivos passa-baixas, devem ser empregados para atenuar as harmônicas indesejáveis, concentradas em torno da frequência da portadora e suas múltiplas. Dessa maneira, evidencia-se que quanto maior for a frequência das harmônicas melhor é a atuação do filtro, exigindo uma frequência de corte não tão reduzida. Assim sendo, pode ser mais benéficas harmônicas com amplitudes maiores, porém localizadas na

faixa de maior frequência do espectro em comparação com harmônicas de menor amplitude que se encontram mais próximas da componente de interesse (de baixa frequência - 50 Hz).

Com vistas ao cancelamento de harmônicos, se considerar um inversor ponte completa representado na Figura B.6, como $v_{ab} = v_{ag} - v_{bg}$, observa-se que todos os conteúdos harmônicos na saída dos braços que estiverem em fase se cancelarão e, os que estiverem em contrafase se duplicarão. Assim, desejando-se maximizar a tensão fundamental de saída e propiciar o cancelamento de harmônicos.

É possível mostrar, matematicamente, que se, para uma mesma portadora triangular, usando duas modulantes em contra-fase, se duplicará a componente de baixa frequência e se cancelará o conjunto de harmônicas presentes na frequência da portadora e seus múltiplos ímpares, porém, duplicando as componentes de frequência par. Tal resultado é mostrado no espectro harmônica da Figura B.7 Com efeito, a frequência de comutação efetiva na saída do inversor é duplicada. Essa configuração de modulação para o inversor de ponte completa é chamada de unipolar (NAMBOODIRI; WANI, 2014).

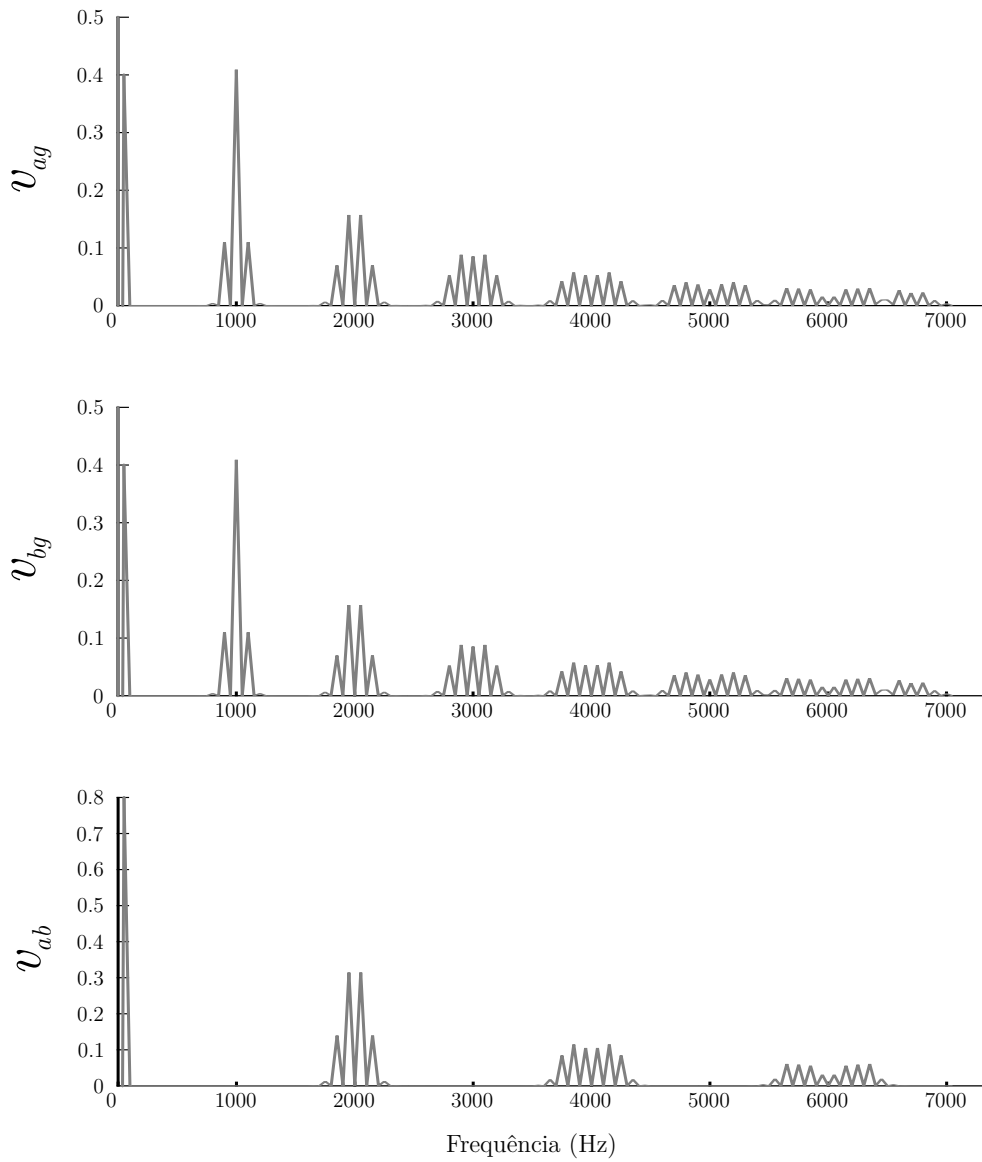
Figura B.6 – Inversor monofásico Ponte Completa.



Fonte: Autor.

O mesmo raciocínio desenvolvido para o inversor ponte completa pode ser expandido para módulos de conversores em série e em conversores multiníveis, culminando nas estratégias PD, POD e APOD (MCGRATH; HOLMES, 2002b; BABKRANI et al., 2017). No entanto, essa análise foge do escopo desse apêndice.

Figura B.7 – Cancelamento de harmônicos em um inversor Ponte Completa: Tensões de fase e de linha correspondente.



Fonte: Autor.