



SIMULADOR EM TEMPO-REAL BASEADO NA INTEGRAÇÃO DE MÓDULO
FPGA E CPU_s PARA AVALIAÇÃO DE CONTROLADORES EMBARCADOS DE
CONVERSORES ELETRÔNICOS DE POTÊNCIA

Oscar Antonio Solano Rueda

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Elétrica.

Orientador: Luís Guilherme Barbosa Rolim

Rio de Janeiro
Outubro de 2019

SIMULADOR EM TEMPO-REAL BASEADO NA INTEGRAÇÃO DE MÓDULO
FPGA E CPU_s PARA AVALIAÇÃO DE CONTROLADORES EMBARCADOS DE
CONVERSORES ELETRÔNICOS DE POTÊNCIA

Oscar Antonio Solano Rueda

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ
COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA
UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM
CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. Luís Guilherme Barbosa Rolim, Dr. –Ing.

Prof. Edson Hirakazu Watanabe, D. Eng.

Prof. Robson Francisco da Silva Dias, D.Sc.

Prof. Pedro Gomes Barbosa, D.Sc.

Prof. Luís Fernando Corrêa Monteiro, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

OUTUBRO DE 2019

Rueda, Oscar Antonio Solano

Simulador em tempo real baseado na integração de módulo FPGA e CPUs para avaliação de controladores embarcados de conversores eletrônicos de potência/ Oscar Antonio Solano Rueda. – Rio de Janeiro: UFRJ/COPPE, 2019.

XVII, 115 p.: il.; 29,7 cm.

Orientador: Luís Guilherme Barbosa Rolim

Tese (doutorado) – UFRJ/ COPPE/ Programa de Engenharia Elétrica, 2019.

Referências Bibliográficas: p. 88- 99.

1. Simulação em tempo real. 2. Modelos chaveados de conversores controláveis. 3. FPGA. 4. Co-simulação multitaxa. 5. Bancada HIL. I. Rolim, Luís Guilherme Barbosa. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

“El buen jugador siempre tiene suerte”
José Raúl Capablanca

Agradecimentos

Agradeço a Deus e à vida.

Agradeço aos meus pais, Miguel e Betty, pelo seu esforço em manter uma estrutura familiar sólida, que impulsionasse o crescimento profissional meu e de meus irmãos.

Agradeço a minha esposa Cindy, por ser essa pessoa incondicional que acredita como ninguém nas minhas capacidades, pela motivação, pelo aconselhamento e pelo amor.

Agradeço ao Cepel (Centro de Pesquisas de Energia Elétrica), por disponibilizar os recursos para a realização deste trabalho.

Agradeço ao prof. Luís Guilherme Barbosa Rolim por sua orientação, caracterizada por confiança no meu desempenho, ótimas apreciações e objetivo direcionamento.

Agradeço aos colegas e amigos do Cepel que me acompanharam nesta longa jornada: ao Cesar Bandim, companheiro de sala, pelo apoio, pelas discussões técnicas e pela ajuda nas dúvidas gramaticais de português; a Ricardo Ross e José Eduardo Alves pelas discussões e sugestões; a Marta Olivieri, Ary Pinto Jr. e Wagner Duboc pelo apoio motivacional; ao Tiago Moraes, pela ajuda na bancada; aos funcionários da biblioteca do Cepel por cuidar desse valioso espaço; e a todos os demais colegas que, de uma maneira ou de outra, ajudaram nesta tese.

Agradeço aos professores membros da banca pelas objetivas sugestões, focadas em melhorar a qualidade técnica desta tese.

Agradeço aos passeios com o Pepper, que muitas vezes me ajudaram a dispersar minha mente, de forma a enxergar as barreiras encontradas durante a tese desde outra perspectiva.

Agradeço aos meus amigos colombianos residentes no Brasil, por ajudar a diminuir a distância com minha cultura e costumes.

Agradeço a todos os amigos e familiares que estiveram na torcida e vibraram com esta conquista.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

SIMULADOR EM TEMPO-REAL BASEADO NA INTEGRAÇÃO DE MÓDULO
FPGA E CPU_s PARA AVALIAÇÃO DE CONTROLADORES EMBARCADOS DE
CONVERSORES ELETRÔNICOS DE POTÊNCIA

Oscar Antonio Solano Rueda

Outubro/2019

Orientador: Luís Guilherme Barbosa Rolim

Programa: Engenharia Elétrica

Esta tese propõe uma estratégia de aprimoramento do método ADC (*Associated Discrete Circuit*) para modelagem de chaves para simulação em tempo real. O método ADC se caracteriza por originar uma matriz de admitâncias do sistema constante, independentemente do estado das chaves existentes. Essa característica faz dele um método de rápida execução e, portanto, apropriado para aplicações nas que são requeridos passos de simulação da ordem de um microssegundo. Entretanto, esse método introduz oscilações numéricas à simulação. A estratégia proposta reduz significativamente essas oscilações com um incremento de complexidade menor do que o ocasionado por abordagens similares previamente apontadas para este fim. Com base na estratégia de modelagem proposta e no uso de uma FPGA genérica como núcleo computacional, um SDTR (Simulador digital em tempo real) foi desenvolvido e validado. Esta validação foi realizada através de uma bancada HIL (*Hardware-in-the-loop*) experimental onde um modelo de VSC trifásico, simulado em tempo real com passo de solução de 1 μ s, é comandado por um dispositivo DSP em malha fechada. Além disso, um sistema de co-simulação multitaxa, no qual o SDTR baseado em FPGA desenvolvido é acoplado a um SDTR comercial baseado em processadores CPU, é apresentado. A aplicabilidade desse sistema integrado de co-simulação na avaliação de controladores embarcados de conversores de potência conectados à rede é exemplificada com alguns casos de estudo.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

REAL-TIME SIMULATOR BASED ON THE INTEGRATION OF FPGA MODULE
AND CPU_s FOR TESTING OF POWER ELECTRONICS CONVERTERS
EMBEDDED CONTROLLERS

Oscar Antonio Solano Rueda

October/2019

Advisor: Luís Guilherme Barbosa Rolim

Department: Electrical Engineering

This thesis proposes an improvement strategy for the ADC (Associated Discrete Circuit) switch method for real-time simulation. The ADC method is characterized by producing a constant system admittance matrix, regardless of the state of the existing switches. This feature makes it a fast-execution method and, therefore, suitable for applications where simulation time steps in the range of a microsecond are required. However, this method introduces numerical oscillations to the simulation. The proposed strategy significantly reduces these oscillations with less complexity increment than that caused by similar approaches previously appointed for this purpose. Based on the proposed modeling strategy and the use of a generic FPGA as computational core, a RTS (Real Time Digital Simulator) is developed and validated. This validation is performed through an experimental HIL (Hardware-in-the-loop) test-bench in which a three-phase VSC model, simulated in real-time with a time step of 1 μ s, is controlled in closed loop by a DSP device. In addition, a multirate co-simulation system, in which the developed FPGA-based RTS is coupled with a commercial CPU-based RTS, is presented. The applicability of this integrated co-simulation system for testing of embedded controllers for grid connected power converters is exemplified by some case studies.

Sumário

Lista de Figuras	xi
Lista de Tabelas	xvi
Lista de Abreviaturas.....	xvii
1. Introdução.....	1
1.1. Simulação digital de sistemas elétricos em tempo real.....	1
1.2. Ensaio <i>Hardware-in-the-loop</i> (HIL)	3
1.3. Eventos de chaveamento entre passos de amostragem	5
1.4. Simulação em tempo real de modelos chaveados de conversores com FPGAs	7
1.5. Motivação e foco	9
1.5.1. Objetivos.....	11
1.5.2. Contribuições da tese.....	13
1.5.3. Produção acadêmica	13
1.6. Estrutura do Texto	14
2. Modelagem de chaves controláveis para simulação em tempo real.....	16
2.1. Algoritmos de solução para simulação EMT.....	16
2.1.1. Espaço de estados	17
2.1.2. Algoritmo de Dommel.....	17
2.1.2.1. Métodos de integração numérica Trapezoidal e Euler regressivo para simulação em tempo real.....	20
2.2. Revisão das técnicas de modelagem de chaves controláveis.....	22
2.2.1. Técnicas com foco na chave.....	22
2.2.1.1. Métodos analíticos	22
2.2.1.2. Métodos de modelagem comportamental	23
2.2.2. Técnicas com foco no sistema elétrico	24
2.2.2.1. Método de valor médio	24
2.2.2.2. Método de chave ideal	26
2.2.2.3. Método de resistência variável	26
2.2.2.4. Método de circuito discreto associado (ADC).....	28
2.3. Técnicas complementares propostas nesta tese	33

3.3.1.	Validação mediante simulação <i>off-line</i>	34
3.	Desenvolvimento de SDTR baseado em módulo FPGA genérico.....	40
3.1.	Descrição do <i>hardware</i> utilizado.....	40
3.2.	Algoritmo de solução.....	42
3.3.	Abordagem para minimizar o fenômeno de <i>Aliasing</i> nas ondas analógicas de saída	43
3.3.1.	Sincronismo entre DSP e SDTR para adequação dos sinais.....	44
3.4.	Metodologia de programação do SDTR proposto	46
3.4.1.	Solução do sistema	46
3.4.2.	Conversão D/A	47
3.4.2.1.	Programação no módulo FPGA	47
3.4.2.2.	Programação no controlador do sistema PXI.....	48
3.4.3.	HMI	49
3.4.4.	Armazenamento de formas de onda	49
3.5.	Validação do SDTR proposto	49
3.5.1.	Descrição da bancada HIL.....	49
3.5.2.	Simulação de conversor VSC operando em malha aberta.....	50
3.5.2.1.	Avaliação da precisão da bancada HIL.....	51
3.5.2.2.	Comparação do uso de recursos da FPGA para cada método de modelagem	52
3.5.2.3.	Desempenho da abordagem para adequação de sinais analógicos	53
3.5.3.	Simulação de conversor VSC operando em malha fechada.....	55
3.5.3.1.	Sistema elétrico.....	55
3.5.3.2.	Estrutura de controle	56
3.5.3.3.	Resultados obtidos	58
3.5.4.	Caracterização dos limites de desempenho do SDTR/FPGA.....	63
4.	Desenvolvimento do SDTR integrado	67
4.1.	Fundamentação teórica	67
4.1.1.	Simulação em tempo real de sistemas elétricos com processadores tipo CPUs	67
4.1.2.	Co-simulação multitaxa.....	69
4.2.	Descrição do SDTR integrado	71
4.2.1.	Plataformas de <i>hardware</i> utilizadas	71
4.2.2.	Interface de acoplamento.....	72

4.3.	Simulações e discussões	73
4.3.1.	Caso I: conexão de VSC com equivalente de rede	73
4.3.2.	Caso II: conexão com rede <i>benchmark</i> de distribuição	78
4.3.3.	Discussão	83
5.	Considerações finais	85
5.1.	Conclusões	85
5.2.	Principais atividades futuras	86
	Referências Bibliográficas	88
A.	Apêndices	100
A.	Análise numérica das perdas de potência virtuais no circuito discreto associado de um indutor	100
B.	Determinação das multiplicações matriciais para solução do sistema	104
C.	Descrição dos blocos de controle em malha fechada	107
D.	Separação do sistema elétrico através do modelo Bergeron de linha	114

Lista de Figuras

Figura 1.1 – Representação do tempo de execução da simulação em tempo real.	2
Figura 1.2 – Esquema básico dos sistemas HIL.	3
Figura 1.3 – Ilustração de uma bancada HIL aplicada à avaliação de controladores embarcados de conversores. HUT: Equipamento sobre ensaio.	5
Figura 1.4 – Representação de Eventos de chaveamento entre passo de amostragem de simulação (ITS)	6
Figura 1.5 – Representação do método ADC para modelagem de chaves.	8
Figura 2.1 – Circuitos discretos associados utilizados para representar elementos passivos armazenadores de energia nos algoritmos de solução baseados no algoritmo de Dommel: (a) Indutor; (b) Capacitor.	18
Figura 2.2 – Tensão em um indutor após interrupção de corrente, método Trapezoidal de integração numérica. Adaptada de [75].	21
Figura 2.3 – Modelo comportamental de uma chave, adaptado de [27]: (a) circuito elétrico; (b) formas de onda descrevendo o comportamento linearizado da chave. $t_{d(on)}$: tempo de atraso para fechar chave, t_{ri} : tempo de subida corrente, t_{fv} : tempo descida tensão, V_{on} : tensão de circuito fechado, $t_{d(off)}$: tempo de atraso para abrir chave, t_{rv} : tempo de subida tensão e t_{fi} : tempo descida.	23
Figura 2.4 – Modelo de chave de valor médio: (a) circuito exemplo; (b) valor médio da função de chaveamento. Adaptada de [83].	25
Figura 2.5 – Modelagem de um ramo de um conversor VSC pelo método de resistência variável.	27
Figura 2.6 – Modelagem da chave controlável pelo método ADC.	28
Figura 2.7 – Circuito simplificado para ilustrar o problema de perda de potência virtual no método ADC.	30
Figura 2.8 – Representação dos estados da chave na representação L/RC do método ADC.	32
Figura 2.9 – Lógica para determinação das correntes históricas no método G-ADC. ...	33
Figura 2.10 – Sistema elétrico utilizado na validação <i>off-line</i> de exatidão.	35
Figura 2.11 – Correntes no lado CA do inversor: (a) formas de onda calculadas pelo método de chave ideal (conforme definição da Seção 2.2.2.2), (b) formas de onda dos erros de corrente para cada um dos outros métodos empregados.	36

Figura 2.12 – Visão ampliada das comparações entre a corrente i_a calculada pelo método de chave ideal e a mesma corrente no caso dos métodos de representação ADC, G-ADC, ADC-I e G-ADC-SI.	38
Figura 2.13 – Comparação entre a forma de onda da tensão v_{ia} obtida com o método ideal e as resultantes com os métodos: (a) ADC, (b) G-ADC, (c) ADC-I, (d) G-ADC-SI.	39
Figura 2.14 – <i>Zoom</i> vertical na comparação entre a forma de onda da tensão v_{ia} obtida com o método ideal e as resultantes com os métodos G-ADC e ADC-I.	39
Figura 3.1 – Plataforma PXI utilizada como <i>hardware</i> do SDTR/FPGA.	41
Figura 3.2 – Diagrama de fluxo do algoritmo utilizado para simulação EMT.	43
Figura 3.3 – Representação da transferência de sinais para uma bancada HIL na qual as chaves modeladas no SDTR proposto sejam controladas por um HUT. Nomenclatura: seta cinza: sinais digitais internos ao SDTR; seta verde: sinais externos ao SDTR; retângulo em linha tracejada: indica em que componente(s) do SDTR é realizada cada operação; <i>Tex</i> : passo de exportação de sinais; ZOH: <i>Zero-order hold</i>	44
Figura 3.4 – Representação da segunda abordagem, que usa sincronismo entre o HUT e o SDTR. Considerando como exemplo o caso de um conversor de meia ponte.	45
Figura 3.5 – Programação do cálculo da corrente histórica $j_s[k]$ em LabVIEW FPGA considerando o método G-ADC: (a) $S_k = 0$ e $S_{k-1} = 1$; (b) $S_k = 1$ e $S_{k-1} = 0$; (c) $S_k = 0$ e $S_{k-1} = 0$; (d) $S_k = 1$ e $S_{k-1} = 1$	47
Figura 3.6 – Programação da lógica de sincronismo DSP-SDTR no módulo FPGA: (a) operação quando é identificada uma borda de subida no sinal de sincronismo; (b) operação nos outros instantes.	48
Figura 3.7 – Programação da lógica de sincronismo DSP-SDTR e exportação de sinais analógicos em LabVIEW.	48
Figura 3.8 – Fotografia da bancada HIL. SDTR: Simulador Digital em Tempo Real desenvolvido; DSP: processador digital de sinais.	50
Figura 3.9 – Medição experimental de uma onda analógica de saída do SDTR/FPGA (i_{aD}/A), sem o uso de adequação de sinais: (a) i_{aD}/A ; (b) <i>zoom</i> horizontal na onda i_{aD}/A ; (c) harmônicos próximos da frequência fundamental.	53
Figura 3.10 – Medição experimental de uma onda analógica de saída do SDTR/FPGA (i_{aD}/A), com o uso da técnica de adequação por sincronismo: (a) i_{aD}/A ; (b) <i>zoom</i> horizontal na onda i_{aD}/A ; (c) harmônicos próximos da frequência fundamental.	54

Figura 3.11 – Sistema elétrico utilizado na validação com ensaio HIL.	55
Figura 3.12 – Diagrama de blocos da estrutura de controle em malha fechada.	57
Figura 3.13 – Comportamento do VSC diante de uma mudança de referência de potência ativa: (a) resultados experimentais – chaves modeladas com a técnica ADC-I; (b) resultados de simulação <i>off-line</i>	59
Figura 3.14 – Correntes do inversor armazenadas no registro FIFO DMA – ensaio de mudança de referência ativa. Chaves modeladas com a estratégia ADC-I.	60
Figura 3.15 – Desempenho do controle interno de corrente - ensaio de mudança de referência ativa: (a) correntes i_d e i_q ; (b) erro de corrente; (c) saída do controlador de corrente.	61
Figura 3.16 – Medição com osciloscópio dos sub-harmonicos de corrente próximos da frequência fundamental.	61
Figura 3.17 – Tensão <i>via</i> durante o ensaio de mudança de referência ativa. Visão detalhada dos valores armazenados no registro FIFO DMA. Chaves modeladas pela estratégia ADC-I.	62
Figura 3.18 – Formas de onda de saída no SDTR. $P_{ref}= 0,5$ pu e $Q_{ref}= 0,5$ pu. Chaves modeladas com a estratégia ADC-I.	62
Figura 3.19 –Diagrama temporal da operação de um controle em malha fechada sincronizado com os pulsos PWM: (a) condição normal; (b) condição obtida com o SDTR/FPGA.	64
Figura 3.20 – Comportamento do VSC quando as fontes equivalentes têm distorção harmônica de 7% na frequência de 300 Hz, com defasagem de π rad: (a) resultados experimentais – chaves modeladas com a técnica ADC-I; (b) resultados de simulação <i>off-line</i>	66
Figura 4.1 –Ilustração de requisitos de computação e de passo de simulação para diferentes tipos de aplicações. Adaptada de [7]. EMT: Transitório Eletromagnético....	68
Figura 4.2 – Representação dos passos de simulação e de intercâmbio de informações numa co-simulação multitaxa.	69
Figura 4.3 – Laço algébrico entre dois subsistemas: (a) caso de acoplamento ideal; (b) acoplamento mediante atraso.	70
Figura 4.4 – Representação da bancada HIL para o SDTR integrado. SDTR: Simulador Digital em Tempo-Real; D/A: Conversor digital/analógico; ED: Entrada Digital; A/D: Conversor analógico/digital; HUT: Equipamento sobre ensaio; HMI: Interface homem-máquina.	71

Figura 4.5 – Representação do método de interface de transformador ideal: (a) sistema elétrico original; (b) subsistemas acoplados. K_{ti} : Constante de transformação. PDS: Ponto de separação.	73
Figura 4.6 – Diagrama do sistema de co-simulação do caso I.	73
Figura 4.7 – Correntes injetadas pelo modelo do VSC durante o evento de afundamento: (a) resultados experimentais da bancada HIL; (b) resultados de simulação <i>off-line</i> com representação dos atrasos da bancada HIL; (c) resultados de simulação <i>off-line</i> sem atrasos.	75
Figura 4.8 – Correntes injetadas pelo modelo do VSC e tensão da fase a no PCC durante o evento de degrau de frequência: (a) resultados da bancada HIL, medidos com osciloscópio; (b) resultados de simulação <i>off-line</i> sem atrasos. Potência de referência de 0,85 pu.	77
Figura 4.9 – <i>Zoom</i> no eixo vertical nas ondas de corrente da Figura 4.8.	78
Figura 4.10 – Diagrama unifilar do sistema de co-simulação do caso de estudo II, baseado no <i>benchmark</i> IEEE 34 barras.	79
Figura 4.11 – Resultado da simulação de aumento de geração pelo VSC: (a) ondas representativas medidas pelo osciloscópio; (b) valor eficaz das correntes injetadas; (c) valor eficaz de algumas tensões na rede.	80
Figura 4.12 – Resultados da bancada de co-simulação multitaxa durante o evento de curto-circuito na barra 814: (a) tensões no barramento 830; (b) correntes injetadas pelo VSC.	81
Figura 4.13 – Resultados de simulação <i>off-line</i> durante o evento de curto-circuito na barra 814: (a) tensões no barramento 830; (b) correntes injetadas pelo VSC.	82
Figura 4.14 – Ondas exportadas pelo SDTR/FPGA durante o evento.	83
Figura A.1 – Circuito simplificado considerado na análise de perdas virtuais: (a) circuito contínuo, (b) circuito discreto associado.	100
Figura A.2 – Fluxos de potência ativa em função do passo de solução para o método de integração Euler regressivo.	101
Figura A.3 – Fluxos de potência ativa em função da reatância indutiva para o método de integração Euler regressivo.	102
Figura A.4 – Fluxos de potência ativa em função do passo de solução para o método de integração Trapezoidal.	102
Figura A.5 – Fluxos de potência ativa em função da reatância indutiva para o método de integração Trapezoidal.	103

Figura A.6 – Circuito discreto associado considerado nas simulações efetuadas.	104
Figura A.7 – Diagrama de blocos do PLL utilizado neste trabalho.	107
Figura A.8 – Diagrama do comportamento linearizado para pequenos sinais do PLL utilizado.	108
Figura A.9 – Comportamento do PLL projetado diante de um deslocamento de fase de 180°, resultados de simulação <i>off-line</i> : (a) frequência rastrejada; (b) ângulo θ_{PLL} e tensão na fase a.	109
Figura A.10 – Diagrama esquemático do controlador de corrente.	110
Figura A.11 – Diagrama esquemático do controlador de corrente.	112
Figura A.12 – Representação do modelo de linha Bergeron.	114

Lista de Tabelas

Tabela 2.1– Valores de condutância e fonte de corrente em função do método de integração numérica.....	19
Tabela 2.2– Parâmetros do sistema elétrico da Figura 2.10	35
Tabela 2.3– Avaliação de exatidão das abordagens de representação de chaves.....	37
Tabela 3.1– Principais características do módulo PXIe-7965R.	41
Tabela 3.2– Configuração da representação numérica em ponto fixo	51
Tabela 3.3– Avaliação de exatidão das abordagens de representação de chaves.....	51
Tabela 3.4– Características de compilação do módulo FPGA PXIe-7965R.	52
Tabela 3.5– Parâmetros do sistema elétrico da Figura 3.11.	55
Tabela 3.6– Configuração dos ganhos dos controladores PI.....	57
Tabela 3.7– Resultados de avaliação do indicador ε_{exp} para o SDTR/FPGA diante distorções harmônicas na rede.	65

Lista de Abreviaturas

- ADC – *Associated Discrete Circuit*
- CHIL – *Control Hardware-in-the-loop*
- CPU – *Central Processing Unit*
- D/A – *Digital-to-Analog Converter*
- DSP – *Digital Signal Processor*
- EMT – *Eletromagnetic Transient*
- EMTP – *ElectroMagnetic Transients Program*
- FACTS – *Flexible Alternating Current Transmission System*
- FAMNM – *Fixed Admittance Matrix Nodal Method*
- FPGA – *Field Programmable Gate Array*
- FXP – *Fixed-Point*
- HIL – *Hardware-in-the-loop*
- HMI – *Human-machine Interface*
- HUT – *Hardware Under Test*
- HVDC – *High Voltage Direct Current*
- IGBT – *Insulated Gate Bipolar Transistor*
- ITS – *Inter-simulation Time-step Switching*
- MMC – *Modular Multilevel Converter*
- PCC – *Point of Common Coupling*
- PHIL – *Power Hardware-in-the-loop*
- PI – *Proportional-Integral Controller*
- PLL – *Phase-Locked Loop*
- PMU – *Phasor Measurement Unit*
- PWM – *Pulse-width Modulation*
- DER – *Distributed Energy Resources*
- SDTR – *Simulador digital em tempo real*
- TNA – *Transient Network Analyzer*
- TS- *Transient Simulation*
- VSC – *Voltage Source Converter*

Capítulo 1

Introdução

1.1. Simulação digital de sistemas elétricos em tempo real

A simulação de sistemas elétricos de potência desempenha um papel estratégico no planejamento e operação destes sistemas, bem como no projeto dos equipamentos que os constituem. Operadores da rede, fabricantes de equipamentos, pesquisadores, entre outros, utilizam um amplo portfólio de ferramentas de simulação para realizar estudos que permitam investigar os impactos provocados pela conexão de um novo subsistema na rede ou pelo ajuste nos parâmetros dos equipamentos existentes, de forma a garantir que a confiabilidade e eficiência do sistema elétrico não seja deteriorada [1], [2].

O uso de modelos de simulação para análise do comportamento dinâmico dos sistemas elétricos é anterior à consolidação da computação digital. O *Transient Network Analyzer* (TNA) é um exemplo de um computador eletrônico analógico, criado na década de 1950, que foi amplamente utilizado na análise de estabilidade de sistemas de potência [3]. Neste tipo de dispositivos, o sistema de potência objeto de estudo era representado em escala reduzida por modelos analógicos de bancada compostos por seções de linha (modelo π -equivalente), amplificadores operacionais, transformadores, capacitores, cargas, entre outros [4].

O desenvolvimento dos computadores digitais, e de programas para análise de transitórios eletromagnéticos (EMT – *Electromagnetic Transient*) e eletromecânicos (TS - *Transient Simulation*) ocasionou que os TNAs fossem gradualmente substituídos por ferramentas digitais de simulação [3]. Para isto acontecer, foram realizados numerosos estudos de comparação de desempenho, que evidenciaram circunstâncias nas quais as perdas e o amortecimento dos circuitos do TNA impediam a representação de problemas existentes no sistema real sob estudo. Para estes casos, os simuladores digitais, por outro lado, conseguiam obter resultados mais próximos dos medidos em campo [1]. A portabilidade, escalabilidade e fácil desenvolvimento e manutenção do

código desenvolvido para simulação¹ foram outros fatores importantes nesta mudança de paradigma [5].

Em contraste com a simulação analógica, que é executada naturalmente em tempo real, as simulações digitais podem ser de dois tipos: *off-line* ou em tempo real, em função do tempo de execução (t_e) de um passo de simulação do modelo (Δt). O tempo de execução, que consiste no tempo (medido por um relógio terrestre) requerido pelo simulador para medir as entradas, solucionar as equações do sistema e estabelecer as variáveis de saída, é consequência da complexidade do sistema elétrico modelado e dos recursos computacionais disponíveis na ferramenta de cálculo utilizada [6]. Nas simulações *off-line*, o tempo de execução é independente do passo de simulação². Nas simulações em tempo real, por outro lado, o tempo de execução deve ser igual ou menor que o passo de simulação do modelo, que é fixo e definido em função das características dinâmicas do sistema e dos fenômenos objeto de interesse [4], [7], [8]. Nos casos em que t_e é menor do que Δt , um tempo ocioso deve ser adicionado no algoritmo de solução, de forma a manter sincronismo entre Δt e o início do próximo passo de solução, como descreve a Figura 1.1.

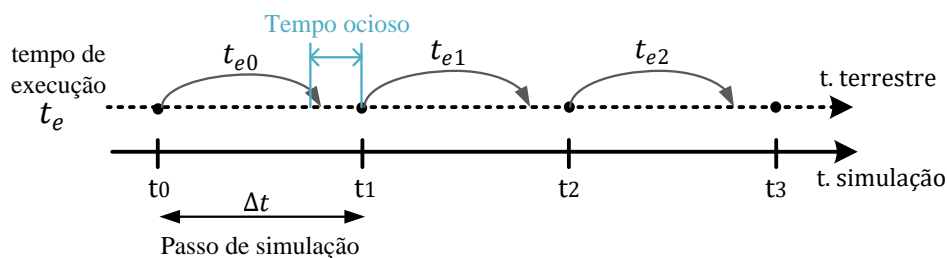


Figura 1.1 – Representação do tempo de execução da simulação em tempo real.

Inicialmente, os simuladores digitais estavam limitados a aplicações de simulação *off-line*, uma vez que o processador utilizado para resolver numericamente as equações do sistema não conseguia atingir os tempos de execução requeridos para manter o sincronismo exigido pela simulação em tempo real. Neste sentido, trabalhos como o realizado por Roitman et al. [9] propuseram abordagens de simulação híbrida (analógica/ digital) em tempo real, nas quais modelos digitais programáveis eram acoplados a simuladores analógicos. O desenvolvimento contínuo das tecnologias de

¹ Em contraste os TNAs, onde os circuitos analógicos deviam ser desmontados cada vez que um novo estudo fosse iniciado e, ainda, exigiam manutenção caso permanecessem montados por longos períodos de tempo.

² Simulações *off-line* podem fazer uso de passos variáveis de simulação, neste caso aumenta ainda a variabilidade da relação entre o tempo de execução e o passo de simulação.

computação, entretanto, permitiu, na década de 1990, o surgimento de Simuladores Digitais em Tempo-Real (SDTR) comerciais [10].

A principal motivação para o advento dos SDTR foi o de testar *hardware* de baixa potência (por exemplo, relés ou controladores de sistemas HVDC) interagindo com um sistema elétrico digital, para verificar o funcionamento do referido *hardware* num ambiente controlado, porém representativo do sistema ao que será conectado no mundo real [1]. A referida configuração é conhecida como *Hardware-in-the-loop* (HIL), descrita na próxima seção.

1.2. Ensaio *Hardware-in-the-loop* (HIL)

No contexto dos sistemas elétricos, um ensaio HIL, representado pelo esquema básico da Figura 1.2, pode ser descrito como uma bancada híbrida real/digital no qual uma parte do sistema elétrico é modelada e simulada em tempo real, enquanto que a parte restante consiste em um equipamento real de *hardware* (denominado HUT- *hardware under test*) que interage com a simulação em tempo real através de interfaces de conexão tais como canais analógicos, canais digitais, equipamentos de condicionamento de sinais ou protocolos de comunicação [11], [12]. Desta forma, um requisito do SDTR utilizado neste tipo de ensaios é o de dispor de canais de entrada e saída que permitam tal troca de informações com componentes externos.

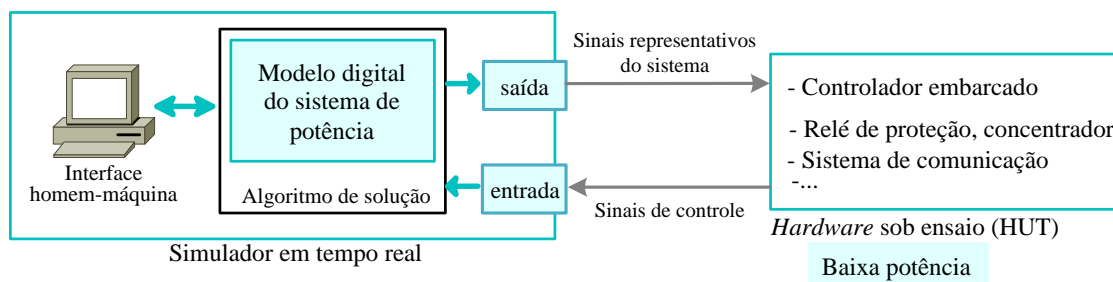


Figura 1.2 – Esquema básico dos sistemas HIL.

Os ensaios HIL podem ser classificados em dois tipos: i) HIL de controle (também denominado CHIL), no qual todo o sistema de potência é representado pelo modelo em tempo real e o HUT constitui um equipamento de controle, através do qual não existe fluxo de potência e ii) *Power* HIL (PHIL) no qual o HUT constitui uma parte do sistema de potência, que intercambia potência com o sistema elétrico digital modelado. Uma vez que o SDTR é um dispositivo eletrônico, incapaz de absorver ou injetar potência, este tipo de bancada utiliza amplificadores de potência como interface entre o sistema digital

e o equipamento externo [8], [11], [13]. A presente tese se foca unicamente nos ensaios tipo HIL de controle.

Os ensaios HIL têm ganhado notoriedade como uma poderosa ferramenta para verificar a operação de equipamentos com controle embarcado, uma vez que tem sido mostrado que podem apresentar vantagens técnicas e/ou econômicas sobre as simulações totalmente computacionais, bem como sobre os ensaios experimentais em laboratório [13], [14]. Apesar das possíveis inexatidões às quais os ensaios HIL estão sujeitos, em relação às condições reais de campo (por exemplo, erros na modelagem da rede elétrica, latência do laço de realimentação ou quantização nos sinais de interface), existe um consenso no entendimento de que estes ensaios podem permitir a criação de condições de teste suficientemente próximas da realidade para avaliar de forma acurada a resposta do HUT diante de variações nas condições operacionais do sistema elétrico ao qual esse equipamento será conectado [11], [14]–[16].

Ensaio HIL têm sido utilizados na validação e certificação de sistemas eletrônicos de controle nas indústrias automotiva e aeroespacial por mais de vinte anos [17]. No setor elétrico, a primeira aplicação de simulação em tempo real e ensaios HIL tratou da validação de sistemas especiais de proteção [1]. Com o aprimoramento da técnica, sua aplicação tem se expandido a outras áreas, incluindo controladores de sistemas HVDC [18], dispositivos FACTS [19], esquemas de proteção de grandes áreas baseados em dispositivos PMUs [20], sistemas de acionamento de motores [21], sistemas eletrônicos de conversão de energia [16], [22], controladores de microrredes [23], [24] e novas topologias de conversores [25], entre outros. No contexto brasileiro, merece destaque o edital de licitação do sistema de transmissão HVDC do Rio Madeira, que exigiu como requisito técnico o fornecimento de uma bancada HIL, onde o desempenho de réplicas das unidades controladoras das estações de conversão seria avaliado pelo ONS antes de sua aprovação para implantação em campo [26].

A abordagem mais comumente utilizada para avaliação de controladores embarcados de conversores eletrônicos mediante ensaios HIL é representada de modo esquemático na Figura 1.3. Nesta abordagem, utilizada no desenvolvimento experimental desta tese, o modelo elétrico digital representado no SDTR utiliza modelos chaveados de conversores eletrônicos, que respondem aos pulsos de chaveamento de entrada originados pelo HUT. Assim, o SDTR realiza, para cada passo de simulação, um algoritmo como o descrito de forma simplificada a seguir: i) medir os pulsos de chaveamento, ii) solucionar as equações dinâmicas do sistema e iii) estabelecer nos seus

terminais de saída os sinais analógicos representativos das grandezas do sistema no ponto de conexão do HUT. O HUT, por sua parte, utiliza tais sinais analógicos como entrada, executa sua função específica de controle e retorna com os pulsos de chaveamento a serem considerados no próximo passo de simulação.

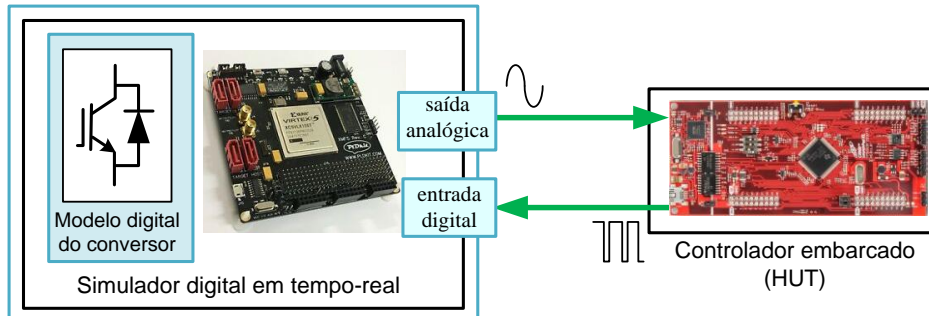


Figura 1.3 – Ilustração de uma bancada HIL aplicada à avaliação de controladores embarcados de conversores. HUT: Equipamento sobre ensaio.

1.3. Eventos de chaveamento entre passos de amostragem

Simulações digitais de modelos chaveados de conversores eletrônicos com passo fixo de solução (sejam estas em tempo real ou *off-line*) estão sujeitas a erros de precisão causados pelos eventos de chaveamento entre passos de simulação (*Inter-simulation Time-step Switching* (ITS), em inglês) [27]. Este fenômeno, representado graficamente pela Figura 1.4, tem origem quando um comando de chaveamento ocorre entre dois instantes de simulação (por exemplo, t_{ch1} na referida figura). Nesta circunstância, uma chave ideal atenderia de forma instantânea o comando enviado pelo controlador³. O modelo digital da chave, no entanto, unicamente mudará de estado no próximo instante de simulação (t_2 na figura). Portanto, o resultado calculado pelo simulador durante o intervalo entre t_{ch1} e t_2 , destacado na Figura 1.4, é incorreto. Os ITS ocasionam erros na simulação que originam harmônicos não característicos e podem chegar a comprometer o funcionamento do modelo do conversor [28].

Uma forma de lidar com os ITS é mediante o uso de algoritmos de sincronização de chaveamento. O PSCAD, por exemplo, é um *software off-line* que utiliza um algoritmo deste tipo, com o seguinte funcionamento: uma vez identificado um evento de ITS entre os instantes $t[k - 1]$ e $t[k]$, é executado um algoritmo de interpolação numérica para

³ Uma chave real apresenta como características físicas tempos de subida e de descida diferentes de zero, dependentes do valor da corrente circulante. No entanto, para estudos com foco no sistema, essas dinâmicas costumam ser desconsideradas, adotando-se um modelo chaveado simplificado.

determinar as tensões e correntes no instante exato de chaveamento (t_{chav} , que constitui um instante de simulação anterior a $t[k]$). Em seguida, o *software* soluciona as equações do sistema para o instante equivalente a um passo de simulação após o instante de chaveamento ($t_{chav} + \Delta t$). Finalmente, utiliza-se uma segunda interpolação para voltar à estampa de tempo original da simulação [29].

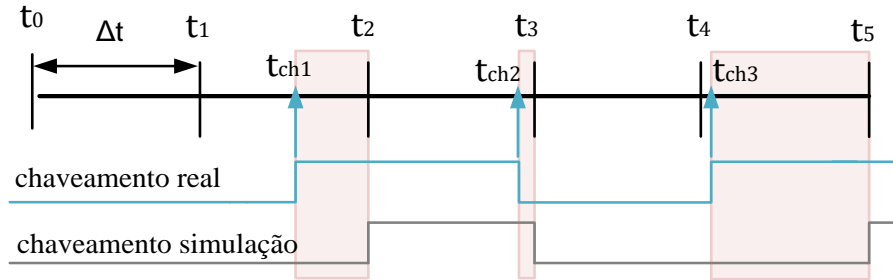


Figura 1.4 – Representação de Eventos de chaveamento entre passo de amostragem de simulação (ITS).

Dado que a execução de um algoritmo de sincronização de chaveamento demanda maior número de operações matemáticas (e conseqüentemente, maior tempo de execução para o mesmo *hardware* de cálculo) do que a solução normal do sistema para um passo de simulação, a aplicação destes algoritmos é desafiadora em SDTR. No entanto, Dinavahi et al. propuseram em [30] uma técnica de correção de ITS baseada em uma única interpolação, utilizada em outros trabalhos da área, por exemplo [11], [31]. Contudo, nesta técnica, a correção das variáveis de estado é realizada um passo de simulação após o evento de chaveamento. Em [16], Li et al. utilizam interpolação linear entre a portadora triangular PWM e o sinal de modulação para estimar o instante de ocorrência do pulso de chaveamento. Esta abordagem, contudo, é dependente da disponibilidade desses sinais, externos ao SDTR em um ensaio HIL como o representado pela Figura 1.3.

A forma mais comumente empregada para lidar com os ITS, por outro lado, é mediante o uso de um passo de solução suficientemente pequeno para representar com um número satisfatório de pontos temporais cada período de chaveamento (T_{ch}) [16], [32]. Neste sentido, a relação ($T_{ch}/\Delta t$) é um indicador importante do efeito dos erros originados pelos ITS na exatidão global da simulação. Em [28], Matar et al. realizaram uma análise de sensibilidade na qual a exatidão da simulação de um conversor fonte de tensão (VSC – *Voltage Source Converter*) trifásico de dois níveis foi avaliada à medida que a relação $T_{ch}/\Delta t$ era modificada. Este trabalho concluiu que uma relação superior a

100 (que permite uma resolução de 1% no ciclo de trabalho do pulso PWM) garante que os erros relacionados a ITS sejam desprezíveis para esta topologia.

Na prática, o anterior implica que a simulação do conversor VSC trifásico de dois níveis utilizando, por exemplo, um chaveamento PWM com frequência de 10 kHz, requer um passo de solução em torno de 1 μ s. Atingir tal tempo de execução tem se mostrado desafiador para dispositivos de processamento baseados em operações sequenciais, como CPUs. Assim, o paralelismo e determinismo das FPGAs (*Field Programmable Gate Array*) têm sido explorados, tanto por equipamentos comerciais quanto por trabalhos acadêmicos, para realizar a simulação em tempo real de conversores eletrônicos chaveados, conforme discussão da próxima seção.

1.4. Simulação em tempo real de modelos chaveados de conversores com FPGAs

Atualmente dispositivos FPGA são considerados o *hardware* de propósito geral mais adequado para ser utilizado como núcleo computacional para simulações em tempo real de modelos chaveados de conversores nas quais seja empregado um passo de solução igual ou inferior a 1 μ s [33]–[35]. A principal vantagem das FPGA é seu processamento paralelo em *hardware*, que permite reduzir de forma significativa o tempo de execução, em comparação com CPUs ou DSPs, por exemplo. Adicionalmente, a latência associada à leitura/exportação dos canais de entrada e saída é geralmente menor do que nesses outros dispositivos, permitindo a criação de ensaios HIL como o ilustrado na Figura 1.3. Por último, por serem dispositivos plenamente configuráveis podem ser adaptados a qualquer caso de aplicação [36], [37].

Conforme mostra a revisão da Seção 2.2 desta tese, dispositivos FPGAs têm sido utilizados como núcleos computacionais de SDTR em diversos trabalhos acadêmicos, com a representação das chaves feita tanto por modelos detalhados [38], [39], quanto por modelos chaveados simplificados (modelos nos quais não são considerados os transitórios das comutações de estado) [28], [34], [36], [40]–[42].

Adicionalmente, os SDTR comerciais que permitem o uso de representação chaveada de conversores para simulações EMT também utilizam FPGAs como núcleo de cálculo. Exemplos são as ferramentas eHS da empresa OPAL-RT [32], GTFPGA Unit da empresa RTDS [43], HIL 602 da empresa Typhoon HIL [35], [44] e IO334 da SpeedGoat [45]. A empresa RTDS é a única que atualmente oferece, além da solução

baseada em FPGA, a opção de simular modelos chaveados de conversores utilizando *hardware* tipo CPU como núcleo de cálculo [46], [47]. No entanto, isto é unicamente possível utilizando seu processador proprietário, projetado especificamente para a aplicação de simulação em tempo real, e para topologias convencionais de conversores, com passo mínimo de simulação da ordem de 1,5 μs .

Apesar das vantagens das FPGAs como ferramenta para simulação em tempo real, seu uso está atrelado a certas limitações. Por um lado, nelas é difícil programar um método eficaz de inversão ou fatorização LU (*Lower-Upper*) das matrizes resultantes do sistema [33], [34], [48], o que é um obstáculo para o uso de técnicas de representação de chaves que requeiram estas operações, como é o caso do método de resistência variável, ver Seção 2.2.2.3. Por outro lado, a representação de modelos elétricos complexos nestes dispositivos costuma ser mais limitada do que em CPUs [25], [49].

A primeira limitação tem sido contornada mediante o uso do método ADC (também referenciado como método Pejovic ou FAMNM, do inglês *Fixed Admittance Matrix Nodal Method*) [50] para modelagem das chaves. A descrição detalhada do método ADC é feita na Seção 2.2.2.4 desta tese. Neste ponto considera-se conveniente apenas ilustrar a representação de chaves utilizada por este método, o que é feito mediante a Figura 1.5. Conforme apresentado, cada chave é modelada como uma condutância constante (G_s) em paralelo com uma fonte de corrente ($j_s[k]$), cujo valor numérico depende da tensão e/ou corrente da chave em instantes de simulação anteriores, de seu estado de comutação e do algoritmo de integração numérica empregado. Uma vez que a condutância G_s é independente do estado da chave, a matriz nodal do sistema obtida a partir deste método é fixa e não é necessária a sua inversão ou decomposição LU durante a simulação, o que permite tempos de execução muito mais rápidos do que no caso do método de resistência variável [4].

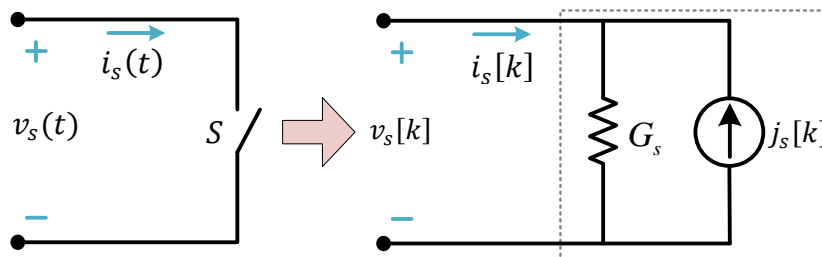


Figura 1.5 – Representação do método ADC para modelagem de chaves.

O método ADC, no entanto, introduz oscilações numéricas e perdas de potência virtuais, erros que podem ser considerados permissíveis para algumas simulações,

porém inaceitáveis em outras, principalmente para altas frequências de chaveamento PWM ou longos períodos de análise [51], [52]. Diferentes trabalhos exploram as limitações do método ADC, propondo estratégias para reduzir os erros numéricos observados. Alguns deles estão focados na definição do valor ótimo para a condutância G_s , principal parâmetro de ajuste deste método [47], [53], [54]; outros na introdução de fontes de compensação no circuito equivalente desta modelagem [55]; e, finalmente, alguns estão baseados em modificar a metodologia com a qual se determina o valor da fonte de corrente $j_s[k]$ a cada instante de simulação [51], [52]. Esta última abordagem é mais contemporânea e apresenta lacunas a serem preenchidas, por exemplo, uma avaliação da relação entre exatidão da representação e o uso de recursos de cálculo nas diferentes estratégias sob esta abordagem.

No que se refere à limitação das FPGAs para representar sistemas elétricos complexos, SDTRs formados por múltiplas FPGA operando em paralelo podem ser empregados, como mostra o trabalho de Li et al. [56], onde um mecanismo de interface sincronizada é utilizado para simular, utilizando um conjunto de quatro FPGAs Altera Stratix®V e passo de $3 \mu s$, uma microrrede composta por múltiplos sistemas fotovoltaicos e uma unidade de armazenamento. Outra abordagem que tem sido empregada para ter maior representabilidade de sistemas elétricos é o uso de co-simulação multitaxa em tempo real baseada em dispositivos FPGA e CPU, na qual os conversores eletrônicos são simulados pela FPGA enquanto que os componentes do sistema com dinâmica mais lenta são simulados pela CPU, com um passo de simulação maior [57], [58].

1.5. Motivação e foco

Entende-se que os sistemas de distribuição brasileiros devem experimentar níveis de penetração de recursos energéticos distribuídos (DER – *Distributed Energy Resources*)⁴ cada vez maiores. Conforme mostra a experiência de outros países, os DER têm o potencial de mudar significativamente as características dos sistemas de distribuição, podendo afetar sua confiabilidade e qualidade de energia [59]. Contraditoriamente, os DER baseados em conversores eletrônicos constituem equipamentos muito mais

⁴ Recursos Energéticos Distribuídos são definidos pela EPE (Empresa de Pesquisa Energética) como tecnologias de geração e/ou armazenamento de energia elétrica, localizados dentro dos limites da área de uma determinada concessionária de distribuição, normalmente junto a unidades consumidoras, atrás do medidor.

flexíveis e controláveis do que as cargas elétricas tradicionais. Assim, nos locais adequados e sob o comando de apropriados laços superiores de controle, podem prover flexibilidade ao sistema, contribuir para a redução de perdas e melhorar o uso da capacidade de potência disponível na rede [60]. Para materializar tais benefícios, no entanto, são necessárias as implantações de novas metodologias de controle bem como de requisitos técnicos de conexão mais exigentes, dentro do contexto das chamadas redes elétricas inteligentes (*Smart grids*).

Os ensaios HIL, viáveis mediante o uso de um SDTR, vêm se consolidando como uma ferramenta de ponta para testar o desempenho de equipamentos com sistemas de controle avançados, como os mencionados anteriormente. Os ensaios HIL se destacam por permitir representar de forma realística as particularidades da rede de distribuição onde os DERs serão conectados [14], [61]. Laboratórios de renome mundial dedicados ao ensaio e certificação de conversores eletrônicos com funcionalidades avançadas fazem uso desta tecnologia [62]. No Brasil, o Cepel – Centro de Pesquisas de Energia Elétrica – está desenvolvendo uma infraestrutura, focada na verificação de funcionamento de DERs com funções avançadas de controle, que contará com uma bancada Power HIL com capacidade de potência de 45 kVA em baixa tensão [63].

A simulação em tempo real de modelos chaveados de conversores eletrônicos, necessária nos ensaios HIL em discussão, tem como principal desafio o requisito de um passo de simulação muito reduzido, da ordem de 1 μ s para conversores VSCs de dois níveis. O método ADC é identificado como a principal abordagem utilizada para modelagem de chaves eletrônicas, dada sua rapidez de execução e sua adequação para implementação em dispositivos FPGA. Contudo este método introduz erros numéricos que podem comprometer a exatidão do ensaio. Neste contexto, o estudo sobre estratégias de aprimoramento do método ADC revela-se importante, representando um tópico de pesquisa atual no contexto mundial.

O uso da simulação em tempo real como ferramenta de pesquisa, ensino, treinamento e/ou validação de protótipos tem como barreira o alto custo dos SDTR comerciais. Esta dificuldade tem motivado vários trabalhos focados em desenvolver SDTR baseados em plataformas genéricas de *hardware*, de menor custo, por exemplo, [40], [41], [51], [56], [64]. No cenário brasileiro podem ser identificadas iniciativas neste contexto, entretanto ainda em estágio inicial [65]–[67]. Um dos aspectos que precisa ser entendido de forma mais clara, neste contexto, é a identificação dos requisitos mínimos do *hardware*

utilizado, no que se refere à capacidade de cálculo e às especificações dos terminais de entrada/saída, para representar determinadas topologias de conversores ou aplicações.

Quanto à limitação das FPGA para representar sistemas elétricos complexos, o uso de plataformas integrando FPGA e CPU na mesma simulação, comumente referenciadas como co-simulação multitaxa em tempo real, tem sido reportado na literatura [25], [57], [58]. O emprego de plataformas de múltiplos fabricantes para desenvolver essas co-simulações tem sido pouco explorado, principalmente devido a sua pouca necessidade prática quando um SDTR comercial é utilizado, uma vez que esses equipamentos geralmente integram dispositivos FPGA-CPU internamente. Entretanto, na tendência de desenvolvimento de SDTR a partir de plataformas genéricas, identificar as capacidades e restrições deste tipo de arranjos se considera importante.

1.5.1. Objetivos

Assim sendo, pelo exposto, o principal objetivo desta tese é realizar uma análise experimental das características de desempenho e de uso de recursos computacionais de um conjunto de estratégias focadas em aprimorar o método ADC para representação chaveada de conversores eletrônicos para simulação em tempo real. As estratégias consideradas nesta análise são: o método ADC Generalizado (G-ADC), proposto por Wang et al [51], e duas técnicas concebidas nesta tese (ADC-I e G-ADC-SI, ver Seção 2.3) que utilizam como base o equacionamento matemático do método G-ADC.

Para realizar a referida análise experimental, um SDTR baseado em um sistema genérico de instrumentação e medição foi desenvolvido. Este SDTR, que realiza a solução das equações do sistema elétrico modelado (que contém chaves eletrônicas) utilizando uma FPGA como núcleo computacional (pelo qual é denominado neste trabalho como SDTR/FPGA), é acoplado em laço fechado a um microprocessador digital de sinais (DSP – *Digital Signal Processor*), de forma a criar uma bancada HIL.

Em uma segunda etapa, o SDTR/FPGA é acoplado a um SDTR comercial baseado em CPU, constituindo um sistema integrado de co-simulação multitaxa em tempo real (denominado neste trabalho como SDTR integrado). O objetivo desta combinação de plataformas é o de ter maior capacidade de representação de sistemas elétricos, em comparação à que seria possível utilizando unicamente o SDTR/FPGA. Com o intuito de demonstrar as capacidades do SDTR integrado para avaliar o desempenho de controladores embarcados de conversores eletrônicos, os resultados experimentais

obtidos na simulação de um sistema elétrico que utiliza como base a rede *benchmark* IEEE 34 barras são comparados com os resultantes na simulação desta rede no *software off-line* Simulink.

A seguir é apresentada uma lista das principais atividades requeridas para atingir os objetivos mencionados:

- (1) Apresentar os elementos que motivam o uso de dispositivos FPGA e do método de representação ADC nas simulações em tempo real de modelos chaveados de conversores eletrônicos;
- (2) Analisar os fundamentos do método G-ADC, proposto por Wang et al [51] para aprimoramento do método ADC tradicional. Propor modificações no método G-ADC, de forma a criar técnicas de modelagem de chaves focadas em melhorar a relação entre exatidão e uso de recursos de computação (técnicas ADC-I e G-ADC-SI);
- (3) Comparar a exatidão das técnicas de representação de chaves objeto de estudo desta tese (ADC, G-ADC, ADC-I e G-ADC-SI) na representação de um conversor VSC trifásico de dois níveis, mediante simulações *off-line* e em tempo real;
- (4) Comparar as características de compilação e de execução em tempo real das técnicas de representação de chaves objeto de estudo desta tese. Para isto, uma plataforma FPGA é programada para operar como SDTR com passo de execução de 1 μ s;
- (5) Propor uma abordagem para lidar com as limitações da plataforma FPGA utilizada no que se refere a suas interfaces de entrada/saída;
- (6) Desenvolver uma bancada HIL em laço fechado, como a apresentada esquematicamente na Figura 1.3, integrando o SDTR/FPGA e um microprocessador DSP. Onde no primeiro é simulado um sistema elétrico que contém um conversor VSC trifásico de dois níveis e no segundo é executado o controle de chaveamento do modelo de VSC;
- (7) Propor e desenvolver o SDTR integrado mediante o acoplamento, em *hardware* e em *software*, das plataformas disponíveis baseadas em FPGA e em CPU. Efetuar, na plataforma baseada em CPU, a simulação em tempo real de um sistema de distribuição de *benchmark*, com passo de 50 μ s;

- (8) Apresentar os resultados que validem as propostas e desenvolvimentos propostos e adotados.

1.5.2. Contribuições da tese

Apesar de utilizar como base o método G-ADC, a técnica ADC-I (ADC com inicialização) constitui uma proposição inédita na área de estratégias para representação chaveada de conversores eletrônicos para simulação em tempo real baseada em FPGA. Neste contexto, a análise de comparação experimental efetuada no Capítulo 3 mostra que a referida técnica melhora de forma significativa a exatidão do método ADC tradicional, com um custo computacional menor do que o requerido pelo método G-ADC. Esta comprovação faz da técnica ADC-I uma alternativa interessante em situações nas que a capacidade do *hardware* empregado como simulador seja limitante.

A segunda contribuição desta tese é o desenvolvimento experimental de um SDTR baseado em um módulo FPGA genérico no qual a criação dos sinais analógicos de saída é realizada com uma frequência menor do que a frequência de simulação. SDTRs anteriormente reportados para aplicações como a descrita esquematicamente pela Figura 1.3 utilizam módulos FPGA com conversor D/A de alta velocidade acoplado, de forma a permitir a exportação das variáveis de saída em cada passo de solução. Nesta tese, por outro lado, é proposta uma técnica de adequação de sinais que permite o uso de uma FPGA sem conversão D/A sem incorrer em significativos erros de exatidão pelo fenômeno de *Aliasing*. Esta iniciativa pode motivar ainda mais o desenvolvimento de SDTR de baixo custo.

Finalmente, a terceira contribuição é a descrição metodológica do desenvolvimento de um sistema de co-simulação multitaxa integrado utilizando plataformas de dois fabricantes diferentes. Este assunto, pouco explorado até agora, pode ganhar maior relevância com o incremento da penetração de DER. Neste cenário, ferramentas que permitam analisar a interação entre modelos de conversores eletrônicos e redes elétricas de distribuição detalhadas em tempo real podem ganhar maior destaque.

1.5.3. Produção acadêmica

Como resultado desta tese o artigo “uma técnica baseada no método de circuito discreto associado para simulação em tempo-real de conversores de potência” foi submetido à Revista Eletrônica de Potência da Sobraep. Em setembro de 2019 este artigo foi recomendado para publicação pelos revisores.

1.6. Estrutura do Texto

No Capítulo 2 são discutidas as principais técnicas de modelagem de chaves controláveis, destacando sua aplicabilidade para diferentes tipos de simulações em tempo real. Tais técnicas são categorizadas naquelas com foco no dispositivo e aquelas com foco no sistema elétrico, sendo esta última classificação a de interesse para este trabalho. As vantagens e dificuldades do método de modelagem de circuito discreto associado (ADC) são expostas em detalhe, bem como uma revisão do estado da arte no que se refere a estratégias que têm sido utilizadas para aprimorar o desempenho desse método. Essa discussão finaliza com a criação e validação *off-line* da técnica de modelagem ADC-I, contribuição desta tese.

O Capítulo 3 trata da descrição metodológica e validação do SDTR baseado em FPGA genérica proposto. São apresentados resultados de simulação em tempo real, considerando operação em malha aberta e em malha fechada de um sistema HIL, onde um micro-controlador DSP estabelece os pulsos de chaveamento para o VSC modelado no SDTR/FPGA. As características de desempenho e de compilação de quatro técnicas de modelagem chaveada baseadas no método ADC, entre elas a estratégia ADC-I, são mostradas. Adicionalmente, a abordagem proposta para adequar os sinais exportados pelo simulador, a fim de evitar interferências relevantes pelo fenômeno de *Aliasing* e é discutida neste capítulo. Por último, a caracterização dos limites de desempenho do SDTR/FPGA é apresentada.

No Capítulo 4 é apresentado o desenvolvimento do SDTR integrado, plataforma que utiliza o método de transformador ideal como interface de acoplamento entre os dois simuladores digitais que o compõem. A capacidade de simulação e o desempenho dessa plataforma de co-simulação multitaxa são avaliados com dois casos de estudo, um deles considerando a conexão de um modelo chaveado de VSC no sistema de distribuição *benchmark* IEEE 34 barras. As limitações dessa plataforma são também discutidas.

O Capítulo 5 apresenta as conclusões do trabalho, as implicações esperadas e os possíveis trabalhos futuros.

Na parte final do documento são apresentados os apêndices. No Apêndice A é realizada uma análise numérica das perdas de potência virtuais no circuito discreto associado de um indutor ideal, para o qual os métodos de integração Euler regressivo e Trapezoidal são considerados. No Apêndice B são detalhadas as matrizes e vetores empregados na multiplicação matricial para solução do sistema através do Algoritmo de

Dommel, técnica programada no SDTR/FPGA. O Apêndice C descreve os componentes da rotina de controle em malha fechada programada no microcontrolador DSP. Por fim, o Apêndice D apresenta uma sucinta revisão teórica do modelo Bergeron de linha para separação do sistema elétrico digital.

Capítulo 2

Modelagem de chaves controláveis para simulação em tempo real

A modelagem de chaves para simulações de transitórios eletromagnéticos (EMT) é uma tarefa desafiadora, em especial quando se trata de simulações em tempo real, que busca equilíbrio entre dois requisitos que se opõem: precisão e baixo esforço computacional. Modelos detalhados que representam as características físicas dos dispositivos semicondutores, por exemplo, são considerados os mais acurados, porém são aqueles que requerem maior esforço computacional para serem executados. Assim, a escolha de um método de modelagem de chaves para simulação em tempo real deve estar baseada no grau de detalhamento desejado, em função das características dinâmicas dos fenômenos de interesse do estudo, e na disponibilidade de recursos computacionais.

Neste capítulo é apresentada uma revisão dos principais métodos para modelagem de chaves controláveis para aplicações em ensaios HIL, bem como uma breve descrição dos principais algoritmos de solução utilizados para resolver numericamente as equações que definem o comportamento dinâmico de um sistema elétrico. A partir da análise das características do método ADC Generalizado (G-ADC), duas técnicas de modelagem de chaves foram propostas e validadas mediante simulação off-line.

2.1. Algoritmos de solução para simulação EMT

Considerando a natureza discreta da computação digital, a principal tarefa de um simulador de transitórios eletromagnéticos é a de solucionar, para cada período Δt , as equações, diferenciais discretizadas e algébricas, que definem o comportamento do sistema. De modo geral, os algoritmos de solução utilizados para simulação de sistemas elétricos estão baseados em duas principais abordagens [5], [8]: i) Modelagem por Espaço de Estados e ii) Modelagem pelo Algoritmo de Dommel, descritas a seguir.

2.1.1. Espaço de estados

A representação matricial das equações de espaço de estado para o caso de sistemas lineares invariantes no tempo vem dada por (2.1). Conforme mostrado, este método representa dinamicamente o sistema objeto de estudo como um conjunto de equações diferenciais de primeira ordem acopladas às variáveis internas (conhecidas como variáveis de estado $\mathbf{x}(t)$), e um conjunto de equações algébricas que combinam as variáveis de estado e as entradas ($\mathbf{u}(t)$) para obter as saídas ($\mathbf{y}(t)$) [68]. Onde as matrizes \mathbf{A} , \mathbf{B} , \mathbf{C} e \mathbf{D} descrevem o sistema modelado.

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A} \mathbf{x}(t) + \mathbf{B} \mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C} \mathbf{x}(t) + \mathbf{D} \mathbf{u}(t)\end{aligned}\tag{2.1}$$

No caso de sistemas não lineares, as equações de espaço adquirem o formato generalizado de:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}(t, \mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) &= \mathbf{h}(t, \mathbf{x}(t), \mathbf{u}(t)),\end{aligned}\tag{2.2}$$

cuja formulação numérica costuma apresentar maior dificuldade do que para o caso de sistemas lineares [5].

Alternativamente, sistemas não lineares podem ser solucionados separando o sistema elétrico em dois subsistemas (parte linear e parte não linear) e aplicando o método de espaço de estados unicamente ao subsistema linear (utilizando (2.1)), conforme realizado, por exemplo, pelo *software* comercial Simulink [69].

2.1.2. Algoritmo de Dommel

Muitas das principais ferramentas computacionais de simulação EMT estão baseadas no algoritmo de Dommel, também conhecido como EMTP (*ElectroMagnetic Transients Program*) [5]. No algoritmo de Dommel todos os elementos da rede são representados por circuitos discretos equivalentes, a partir dos quais é possível estabelecer, mediante a aplicação das leis de Kirchhoff e de Ohm, um conjunto de equações nodais algébricas que descrevem o comportamento do sistema para qualquer instante discreto de tempo k [70]. Matricialmente, o sistema de equações resultante neste algoritmo pode ser escrito como segue:

$$\mathbf{Y} \mathbf{e}[k] = \mathbf{i}[k] - \mathbf{I},\tag{2.3}$$

onde \mathbf{Y} é a matriz de condutâncias do sistema, $\mathbf{e}[k]$ é o vetor de tensões nodais no instante k , $\mathbf{i}[k]$ é o vetor de correntes nodais injetadas no instante k e \mathbf{I} é o vetor de fontes de corrente equivalentes de valor conhecido.

Uma abordagem para o algoritmo de Dommel é a de Análise Nodal Modificada, proposta por Ho et al. [71], na qual a solução do sistema é dada a partir da seguinte expressão:

$$\mathbf{H} \mathbf{x}[k] = \mathbf{b}[k], \quad (2.4)$$

onde $\mathbf{x}[k]$ é o vetor de variáveis desconhecidas (neste caso tensões e correntes) no instante k , \mathbf{H} é a matriz representativa de admitâncias do sistema e $\mathbf{b}[k]$ é o vetor de variáveis conhecidas (fontes de corrente). Em (2.4) fica evidente que a determinação das variáveis desconhecidas vem dada pela multiplicação da matriz inversa de \mathbf{H} pelo vetor de variáveis conhecidas $\mathbf{b}[k]$, ou por uma técnica algébrica equivalente.

Os circuitos discretos equivalentes utilizados para representar os elementos passivos armazenadores de energia (indutores/capacitores) nos algoritmos de solução baseados no algoritmo de Dommel são formados por uma condutância em paralelo com uma fonte de corrente, conforme apresentado na parte direita da Figura 2.1. Nesta representação, os valores numéricos dos componentes desses circuitos discretos estão “associados” ao método de integração numérica utilizado para discretizar as equações diferenciais resultantes, conforme mostra a Tabela 2.1 para os casos dos métodos Euler regressivo e Trapezoidal; razão pela qual tais circuitos costumam ser denominados como circuitos discretos associados (ADC, *Associated Discrete Circuit*) [72].

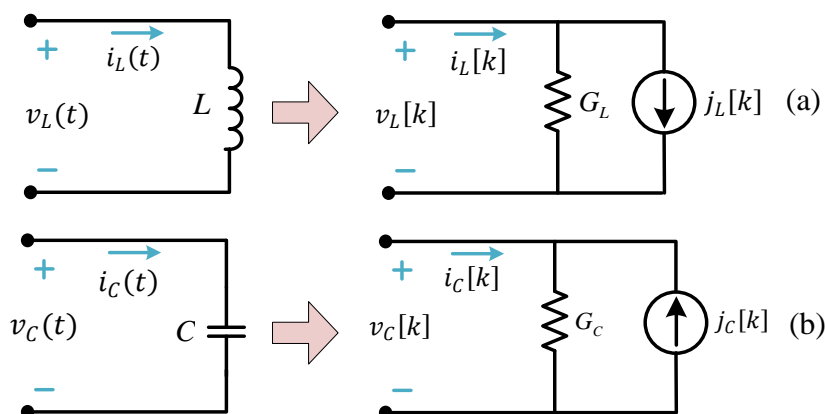


Figura 2.1 – Circuitos discretos associados utilizados para representar elementos passivos armazenadores de energia nos algoritmos de solução baseados no algoritmo de Dommel: (a) Indutor; (b) Capacitor.

Tabela 2.1– Valores de condutância e fonte de corrente em função do método de integração numérica.

Método de Integração	Indutor		Capacitor	
	G_L	$j_L[k]$	G_C	$j_C[k]$
Euler regressivo	$\frac{\Delta t}{L}$	$i_L[k - 1]$	$\frac{C}{\Delta t}$	$G_C \cdot v_C[k - 1]$
Trapezoidal	$\frac{\Delta t}{2L}$	$\frac{\Delta t}{2L} \cdot v_L[k - 1] + i_L[k - 1]$	$\frac{2C}{\Delta t}$	$-\left(\frac{2C}{\Delta t} \cdot v_C[k - 1] + i_C[k - 1]\right)$

No caso do indutor e método de integração numérica Euler regressivo, por exemplo, observa-se que as expressões da Tabela 2.1 são obtidas a partir da equação em tempo contínuo para a queda de tensão $v_L(t)$:

$$v_L(t) = L \cdot \frac{di_L(t)}{dt}, \quad (2.5)$$

que discretizada, utilizando o referido método, resulta em:

$$i_L[k] = \frac{\Delta t}{L} \cdot v_L[k] + i_L[k - 1]. \quad (2.6)$$

Ao comparar (2.6) com a Figura 2.1(a) observa-se que para existir equivalência entre as duas representações, a condutância e a fonte de corrente do circuito discreto associado devem ter os valores:

$$G_L = \frac{\Delta t}{L} \quad \text{e} \quad j_L[k] = i_L[k - 1], \quad (2.7)$$

conforme mostrado na Tabela 2.1.

Outra característica dos circuitos discretos da Figura 2.1 é que neles existe um componente que consome potência ativa (a condutância em paralelo), o que não é próprio dos elementos indutivos ou capacitivos originais. Destaca-se, entretanto, que a fonte de corrente em paralelo destes circuitos discretos constitui um componente ativo, capaz de injetar potência. Assim, como mostra a análise efetuada no Apêndice A, pode acontecer que, em função do método de integração numérica escolhido, a potência requerida pela condutância seja fornecida totalmente pela fonte de corrente em paralelo, caso no qual nenhuma potência ativa seria requerida da rede. O Apêndice A mostra também que os erros nos fluxos de potência da solução discreta, em contraste com os fluxos da solução em tempo contínuo, podem ser associados à inexatidão da técnica de integração numérica.

2.1.2.1. Métodos de integração numérica Trapezoidal e Euler regressivo para simulação em tempo real

Além da modelagem matemática, a simulação discreta de um sistema elétrico está condicionada ao método de integração numérica escolhido. Características como complexidade, exatidão (em contraste com o modelo em tempo contínuo) e estabilidade devem ser levadas em consideração na hora de realizar esta escolha.

Os métodos de integração numérica podem ser descritos como algoritmos que aproximam as equações diferenciais do modelo com base nas séries de Taylor [73], onde a ordem da aproximação tem repercute relevantemente nos erros de truncamento originados. Dentro dos diversos métodos existentes, a discussão apresentada nesta seção se foca nos dois métodos implícitos mais comumente utilizados: Trapezoidal (segunda ordem) e Euler regressivo (primeira ordem).

Método Trapezoidal

No método Trapezoidal, a aproximação do operador derivativo contínuo é dada por:

$$s \rightarrow \frac{2}{\Delta t} \cdot \frac{z - 1}{z + 1}. \quad (2.8)$$

Onde z é a frequência complexa em tempo discreto (Transformada Z).

Este método é classificado como estável absoluto, uma vez que sistemas contínuos estáveis são transformados em sistemas discretos estáveis, ao tempo que sistemas contínuos instáveis são transformados em sistemas discretos instáveis [74]. Adicionalmente, por ser um método de aproximação de segunda ordem o erro de truncamento local é da ordem de Δt^3 , o que é considerado suficientemente exato para aplicações práticas [70]. Assim, constitui uma técnica comumente utilizada para simulação de sistemas elétricos.

Entretanto, o método Trapezoidal pode apresentar “trepidação numérica” (*chatter*) nas tensões através de indutâncias em situações nas quais a derivada da corrente mude abruptamente [75]. Este fenômeno é ilustrado com ajuda da Figura 2.3, onde no instante $t = 1$ a corrente é interrompida pelo disjuntor S_1 . Nessa figura, a solução exata para v_L é mostrada pela linha sólida, que apresenta um salto a zero no instante da interrupção. A linha tracejada, por outro lado, representa a solução pelo método trapezoidal, caracterizada por uma trepidação de alta frequência. Isto porque a tensão no indutor é dada por:

$$v_L[k] = \frac{2L}{\Delta t} \cdot (i_L[k] - i_L[k - 1]) - v_L[k - 1]. \quad (2.9)$$

De forma que a partir do instante de tempo $t = 2$ (instante no qual $i_L[k]$ e $i_L[k - 1]$ são ambas zero), $v_L[k]$ passa a ser igual a $-v_L[k - 1]$. Por outras palavras $v_L[k]$ começa a oscilar ao redor do zero com amplitude igual ao valor prévio à interrupção.

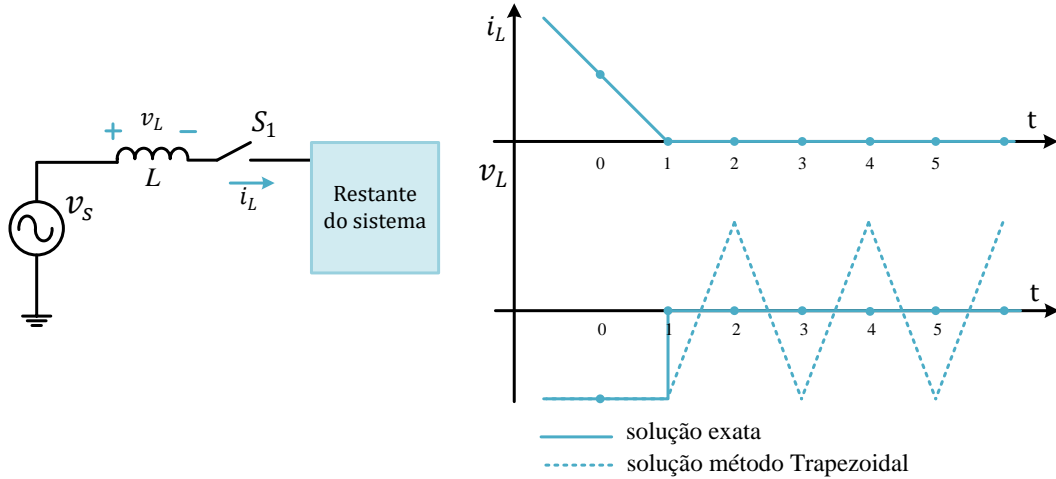


Figura 2.2 – Tensão em um indutor após interrupção de corrente, método Trapezoidal de integração numérica. Adaptada de [75].

Existem técnicas que buscam amortecer estas oscilações, por exemplo através de resistências de amortecimento [76] ou através de algoritmos de detecção e supressão, como realizado pelo software PSCAD [29]. Entretanto, estas técnicas aumentam a complexidade do método de integração e fogem do foco desta tese.

Método Euler regressivo

Neste método o operador derivativo contínuo é aproximado por:

$$S \rightarrow \frac{z - 1}{\Delta t \cdot z}. \quad (2.10)$$

Assim como o método Trapezoidal, o método Euler regressivo tem estabilidade absoluta. Entretanto, uma vez que se trata de um método de primeira ordem, tem um erro de truncamento local, maior, da ordem de Δt^2 . Por outras palavras, para atingir o mesmo nível de exatidão do que o equivalente no caso Trapezoidal, o modelo precisa ser solucionado com um passo menor [34]. Por outro lado, uma vantagem do método Euler regressivo é que ele tem uma característica de amortecimento [50], o que evita o problema de “trepidação numérica” discutido anteriormente. Adicionalmente, se trata de uma abordagem mais eficiente desde o ponto de vista computacional do que o método

Trapezoidal, o que é favorável para a simulação em tempo real de modelos chaveados de conversores. De fato, o método Euler regressivo tem sido aplicado no desenvolvimento de SDTR comerciais e acadêmicos [32], [40], [41], [55].

Pelo exposto, neste trabalho foi adotado o Algoritmo de Dommel para solução do sistema, mais especificamente uma técnica de Análise Nodal Modificada. Na qual a discretização do sistema elétrico utiliza o método de integração numérica de Euler regressivo. Assim, as discussões das próximas seções consideram as referidas escolhas.

2.2. Revisão das técnicas de modelagem de chaves controláveis

Os métodos de modelagem utilizados para representar as chaves nas simulações EMT podem ser divididos em dois grupos, em função do foco do estudo em que são empregados: i) foco na chave e ii) foco no sistema elétrico. A seguir é apresentada uma revisão das principais técnicas dentro dessas classificações.

2.2.1. Técnicas com foco na chave

Conforme mencionado na introdução deste capítulo, a modelagem detalhada das chaves, considerando suas propriedades físicas, é a abordagem mais acurada para representar seu comportamento. Este tipo de técnicas é de interesse em estudos com foco nos fenômenos dinâmicos nas transições de estado, por exemplo, análises precisas das perdas por chaveamento ou das características térmicas do dispositivo [39], [50], estudos caracterizados por passos de simulação relativamente curtos, na ordem de nanosegundos, e duração da simulação considerando apenas alguns períodos de chaveamento. Dado que o foco de análise neste caso está no comportamento interno da chave, nestes estudos não é necessária uma representação detalhada do sistema elétrico ao qual o conversor está conectado.

2.2.1.1. Métodos analíticos

São modelos que se baseiam na descrição do comportamento físico dos semicondutores que compõem o dispositivo [77]. Segundo Myaing e Dinavahi [39], os métodos mais comumente utilizados dentro desta categoria para representação de chaves IGBT são os modelos de Hefner [78] e de Kraus [79] que têm sido implementados em *software off-line* como SABER e SPICE. No entanto, dada sua

complexidade computacional, nenhum destes métodos tem sido utilizado para simulações em tempo real.

2.2.1.2. Métodos de modelagem comportamental

Os métodos dentro desta categoria descrevem a dinâmica de mudança de estado da chave sem considerar os fenômenos físicos dos semicondutores. Para isto, estes métodos utilizam técnicas de ajuste de curvas que determinam, a partir de medições experimentais de tensão e corrente, expressões simplificadas para representar as características dinâmicas da chave a ser modelada [77]. A título de exemplo, a Figura 2.3, adaptada do livro clássico de Mohan [27], mostra a tensão e corrente num modelo linearizado de uma chave durante um período de chaveamento.

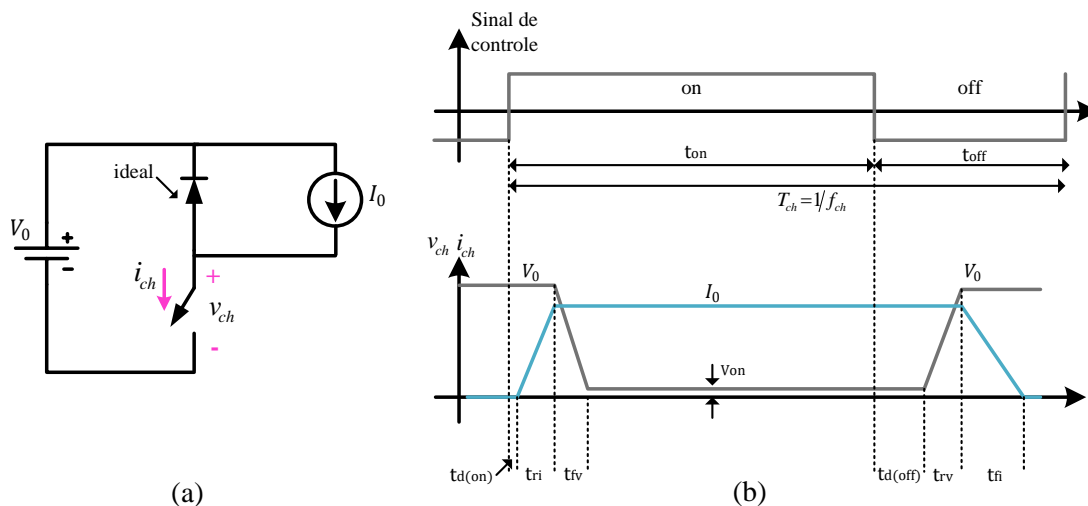


Figura 2.3 – Modelo comportamental de uma chave, adaptado de [27]: (a) circuito elétrico; (b) formas de onda descrevendo o comportamento linearizado da chave. $t_{d(on)}$: tempo de atraso para fechar chave, t_r : tempo de subida corrente, t_f : tempo descida tensão, V_{on} : tensão de circuito fechado, $t_{d(off)}$: tempo de atraso para abrir chave, t_{rv} : tempo de subida tensão e t_{fi} : tempo descida.

Parma e Dinavahi apresentaram em [38] uma das primeiras aplicações conhecidas sobre o uso de FPGAs para simulação de VSCs em tempo real. Nesse trabalho foi utilizada linguagem VHDL, um modelo comportamental linearizado de um IGBT e uma abordagem baseada no uso de uma máquina de estados para simular um conversor VSC trifásico de dois níveis com passo de simulação de 12,5 ns. Myaing e Dinavahi aprimoraram esta representação em [39], modelando de forma não linear os transitórios de comutação (para o qual foram empregadas tabelas *lookup*). Nos dois trabalhos, os VSCs servem como dispositivo de acionamento para modelos digitais de motores de indução que são executados com um passo de simulação maior, de 10 μ s.

2.2.2. Técnicas com foco no sistema elétrico

Este tipo de técnicas é de interesse em estudos focados em avaliar o comportamento do conversor eletrônico como um componente de um sistema elétrico, analisando seu impacto no sistema de potência bem como o desempenho dos laços de controle, diante de variadas condições de operação [39], [80]. As técnicas incluídas dentro desta categoria idealizam os transitórios de transição de estado, o que as tornam abordagens mais simples do que os métodos focados no dispositivo, porém, ao mesmo tempo, menos exatas. Entretanto, geralmente é considerado que os erros introduzidos por esta idealização são pouco relevantes no contexto de análises com foco no sistema [81], o que torna estas técnicas a principal opção nos *software* de simulação de sistemas elétricos [29], [82]. Adicionalmente, o contínuo aprimoramento das tecnologias de dispositivos semicondutores faz com que cada vez sejam menores as diferenças entre uma chave idealizada e um dispositivo real existente no mercado [27].

No contexto dos ensaios HIL, foco deste trabalho, as técnicas de modelagem de chave com foco no sistema elétrico se tornam mais adequadas, uma vez que facilitam análises da iteração entre o HUT e o restante do sistema elétrico. A seguir é mostrada uma descrição dos principais métodos dentro desta categoria.

2.2.2.1. Método de valor médio

Este método está focado em representar apenas o comportamento de baixa frequência do conversor, ignorando as oscilações originadas pela comutação das chaves [83]. Assim, nesta abordagem as grandezas elétricas são representadas pelos seus valores médios, para o qual se costuma selecionar um período de chaveamento PWM como passo de simulação⁵. Neste sentido, as grandezas nas chaves controláveis são modeladas a partir do valor médio da sua função de chaveamento idealizada [81], [84]. Isto pode ser mostrado a partir do circuito simplificado da Figura 2.4(a) onde a tensão $v_o(t)$ pode ser expressa em termos da função de chaveamento $S(t)$ e da tensão de entrada E (considerada constante por simplicidade) por:

$$v_o(t) = E \cdot S(t). \quad (2.11)$$

O valor médio de $v_o(t)$, durante um período de chaveamento, é:

⁵ Em casos nos quais o período de amostragem do controlador seja diferente do período de chaveamento PWM, o primeiro pode ser selecionado como passo de simulação.

$$\langle v_o(t) \rangle = E \cdot \langle S(t) \rangle, \quad (2.12)$$

onde $\langle S(t) \rangle$ é o valor médio da função de chaveamento, que, considerando o caso no qual a unidade de tempo é um período PWM, corresponde ao ciclo de trabalho de $S(t)$.

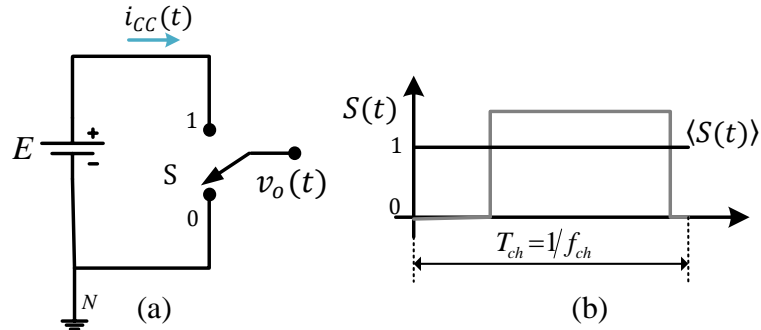


Figura 2.4 – Modelo de chave de valor médio: (a) circuito exemplo; (b) valor médio da função de chaveamento. Adaptada de [83].

Como vantagens deste método podem ser destacadas sua baixa complexidade e o uso de um passo de simulação maior do que nos modelos chaveados, o que facilita sua aplicação em simulações em tempo real. Para Yazdani [81] a modelagem de valor médio fornece uma relação mais direta entre os sinais de referência do controlador e as saídas do conversor, pelo qual esse livro o pondera como o método mais indicado para avaliação de laços de controle. No entanto, deve ser destacado que a desconsideração das comutações limita a largura de banda e a exatidão desta representação [83].

Este tipo de modelagem tem sido empregado por SDTRs. Em [85], por exemplo, Jeon et al. utilizam uma modelagem média de uma microrrede para avaliar, com uma configuração HIL em tempo real, o desempenho dos laços secundários do controlador projetado. Em [15] é proposto um SDTR de baixo custo, para teste de controladores de conversores eletrônicos, baseado em modelagem de valor médio. Em [86] é proposta uma metodologia para determinar o modelo de valor médio de dispositivos FACTS para simulação em tempo real. Em todos estes casos, são utilizados CPUs como núcleos computacionais, uma vez que o passo de simulação não impõe um requisito tão exigente como no caso da modelagem chaveada.

Modelos de valor médio, contudo, não são úteis em estudos nos quais seja requerida uma análise detalhada dos transientes originados pelos eventos de chaveamento [48], [81]. Nestes casos se faz necessário o uso de métodos chaveados, como os apresentados a seguir.

2.2.2.2. Método de chave ideal

Na modelagem ideal, a chave é representada por um circuito aberto no estado de alta impedância (*Off*), bloqueando tensões diretas e reversas com zero corrente de condução, e por um curto-circuito no estado de baixa impedância (*On*), conduzindo correntes arbitrárias sem originar queda de tensão nos seus terminais [27]. Apesar de ser um método cuja fundamentação é simples, apresenta a desvantagem de modificar a configuração do sistema para cada possível estado. Isto porque uma chave caracteriza dois pontos elétricos no estado *Off* (um em cada terminal), porém este mesmo modelo de chave junta os dois pontos elétricos no estado *On*. Assim, um sistema que contenha um conversor eletrônico com m possíveis estados é representado por m matrizes do sistema (\mathbf{H}_m no caso da Análise Nodal Modificada, ver (2.4)), cujas dimensões podem ser variáveis [40]. Esta característica e a necessidade de conhecer as topologias resultantes do conversor para formular as funções de chaveamento idealizadas equivalentes restringe o uso deste método [50], [80].

Os SDTR baseados em FPGA propostos por Vekic et al. [35], Chen e Dinavahi [36] e por Lucía et al. [42], utilizam modelos ideais de chaves. Nestes casos, todas as m matrizes nodais do sistema originadas pelos m possíveis estados de comutação das chaves são calculadas antes do início da simulação e armazenadas como constantes.

2.2.2.3. Método de resistência variável

Este método (e suas adaptações) constitui a principal técnica utilizada por simuladores *off-line* para modelagem de chaves [52]. Nele cada chave é representada por um par de valores de resistências: um valor “pequeno” no estado *On* (R_{on}) e um valor “grande” no estado *Off* (R_{off})⁶ [27], ver Figura 2.5. Isto ocasiona que a matriz representativa do sistema seja modificada cada vez que alguma mudança de estado em qualquer chave acontecer, existindo, da mesma forma que para o caso do método ideal, m diferentes matrizes. Neste caso, porém, a dimensão das matrizes \mathbf{H}_m permanece constante para todos os estados, o que facilita a solução de (2.4). Adicionalmente, nesta técnica, não é necessário conhecimento prévio das topologias resultantes, o que torna de mais fácil automação o procedimento de solução numérica.

⁶ No *software* PSCAD, por exemplo, os valores padrões para estas resistências são: $R_{on}=1\text{ m}\Omega$ e $R_{off}=1\text{ M}\Omega$ [29].

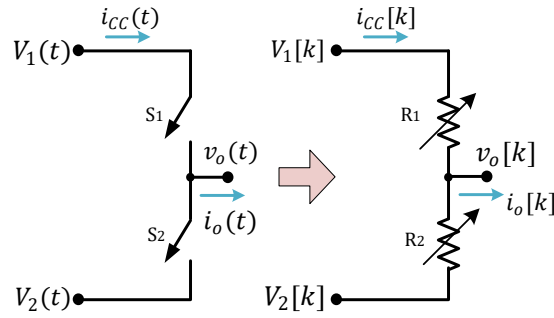


Figura 2.5 – Modelagem de um ramo de um conversor VSC pelo método de resistência variável.

Nos *software* comerciais *off-line* que utilizam esta técnica, a matriz representativa do sistema (\mathbf{H}) é atualizada e invertida (ou descomposta pela fatoração LU) durante a simulação, no instante em que algum chaveamento acontecer [29], [82]. Esta abordagem é restritiva para SDTR, pois o número de operações matemáticas requeridas para redefinir e obter a solução de (2.4) através da fatoração LU⁷, considerando a dimensão de \mathbf{H} como $n \times n$, é [87]:

$$(2/3) \cdot n^3 + 2 \cdot n^2. \quad (2.13)$$

Esforço computacional que contrasta com o número de operações matemáticas nos instantes em que a matriz do sistema se mantém constante ($2 \cdot n^2$).

Assim, abordagens focadas em calcular de modo *off-line* todas as matrizes inversas de \mathbf{H}_m e armazená-las para seu uso durante a simulação têm sido reportadas, por exemplo, em [88]. Entretanto, essas abordagens estão limitadas a aplicações nas quais o número de chaves seja reduzido [32], [80]. Por outro lado, alguns trabalhos prévios têm proposto técnicas para adequar o método de resistência variável (com redefinição on-line da matriz do sistema) para simulação em tempo real baseada em FPGA. Exemplos são [48], onde é proposta uma técnica de separação matemática do sistema elétrico, orientada a criar sub-redes com e sem chaves, de forma a aliviar a carga computacional da reformulação de \mathbf{H} ; e [37] onde é proposta uma técnica de inversão de matrizes otimizada para uso em FPGAs.

Atualmente, o único SDTR comercial que permite o uso do método de resistência variável com inversão on-line da matriz do sistema é o RTDS [46]. No entanto, isto é unicamente possível para topologias tradicionais de conversores e utilizando o *hardware* proprietário (tipo CPU) desenvolvido especificamente para a aplicação de simulação em tempo real.

⁷ O número de operações requeridas para efetuar $\mathbf{x}[k] = \mathbf{H}^{-1}\mathbf{b}[k]$, ou seja, computar a inversa de \mathbf{H} e realizar a multiplicação matricial, é de $(8/3) \cdot n^3$.

2.2.2.4. Método de circuito discreto associado (ADC)

Pejovic et al. propuseram em [50] um método de representação chaveada de conversores focado em manter a matriz representativa do sistema \mathbf{H} constante independentemente do estado das chaves existentes, eliminando assim a necessidade de atualizar e calcular a sua inversa em cada instante de comutação. Neste método, cada chave é modelada como uma condutância em paralelo com uma fonte de corrente, conforme mostrado pela Figura 2.6.

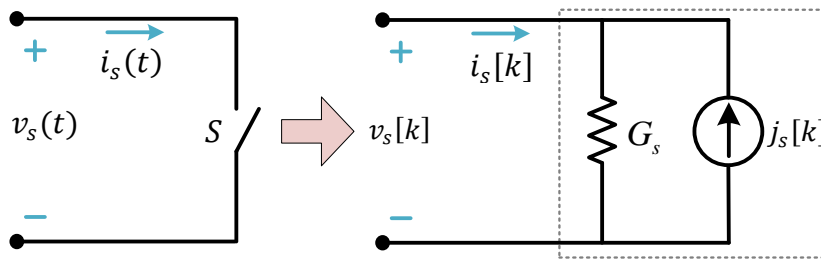


Figura 2.6 – Modelagem da chave controlável pelo método ADC.

Nesta representação o requisito de manter a matriz \mathbf{H} constante é atingido mediante o uso de um valor fixo de G_s . Assim, o estado de comutação é representado pelo valor numérico da fonte de corrente $j_s[k]$, que depende também dos valores históricos de tensão e/ou corrente na chave, bem como do método de integração numérica utilizado⁸. A definição de (2.14), por exemplo, aplica para o caso do método de integração numérica de Euler regressivo [73]. Destaca-se que, considerando a solução do sistema pela abordagem de Análise Nodal Modificada – ver (2.4), $j_s[k]$ constitui um elemento do vetor de variáveis conhecidas ($\mathbf{b}[k]$) o qual deve ser atualizado para cada passo de simulação, inclusive em sistemas elétricos sem chaves [71].

$$j_s[k] = \begin{cases} -i_s[k-1] & \text{se chave fechada} \\ G_s \cdot v_s[k-1] & \text{se chave aberta} \end{cases} \quad (2.14)$$

Este método recebe diferentes nomes na literatura técnica. Sendo que em algumas referências ele é denominado como método Pejovic, fazendo referência ao propositor; em outras como método ADC (circuito discreto associado) porque a representação da chave é feita por um circuito equivalente, tal como realizado pelos algoritmos de solução baseados no algoritmo de Dommel para representar elementos armazenadores de energia (ver Seção 2.1.2). Neste sentido, a partir da comparação de (2.14) com a

⁸ A corrente $j_s[k]$ é denominada como “corrente histórica” do modelo ADC de chave no decorrer desta tese.

Tabela 2.1, interpreta-se que neste método a chave é modelada como um indutor no estado fechado e como um capacitor no estado aberto, onde os valores destes elementos virtuais são definidos pela relação:

$$G_s = \frac{\Delta t}{L} = \frac{C}{\Delta t}, \quad (2.15)$$

o que faz com que o método também seja denominado como *L/C switching*; finalmente, quando este método é utilizado em conjunto com um algoritmo de solução de Análise Nodal, o conjunto recebe o nome de FAMNM (*Fixed Admittance Matrix Nodal Method*).

Experimentalmente, tem sido comprovado que a simulação em tempo real de sistemas elétricos utilizando o método ADC é executada de forma mais rápida do que no caso do método de resistência variável. Esta característica e sua facilidade de automação fazem dele a principal escolha dos SDTR baseados em FPGA, com modelagem chaveada e sem uso de algoritmos de sincronização de chaveamento. Conforme mencionado na Seção 1.4, os principais SDTR comerciais utilizam esta abordagem [32], [43]–[45]. No que se refere a trabalhos acadêmicos, por outro lado, Matar e Iravani apresentaram em [40] e [28] um SDTR para aplicações de eletrônica de potência baseado em FPGA utilizando este método, no qual todo o sistema elétrico modelado é solucionado com passo de 1 μ s. Propostas similares são apresentadas nos SDTR desenvolvidos nas teses de doutorado de Razzaghi [34] e Dagbagi [80].

Este método, contudo, apresenta dois conhecidos problemas de desempenho:

- (1) Sobretensões e sobrecorrentes virtuais nas formas de onda resultantes.
- (2) Perdas de potência virtuais.

O problema número (1) se origina como consequência da interação da indutância ou capacitância virtual, que representam respectivamente os estados *On* e *Off* da chave, com o restante do circuito, o que origina frequências naturais de oscilação que não são próprias do circuito original.

O problema número (2) pode ser ilustrado a partir do circuito simplificado da Figura 2.7. Considerando que este circuito está operando em regime permanente com a chave *S*, modelada pelo método ADC, em estado *On*, as grandezas do circuito equivalente do modelo ADC da chave no instante de tempo *k* são:

$$j_s[k] = -i_s[k - 1] = -i_{load} \quad \text{com} \quad v_s[k] = 0; i_s[k] = i_{load}. \quad (2.16)$$

Nesta condição, a energia armazenada no indutor equivalente é:

$$E_{L-s} = 0,5 \cdot L \cdot i_s[k] = 0,5 \cdot L \cdot i_{load}^2. \quad (2.17)$$

Se no instante de tempo $k + 1$ a chave muda seu estado de baixa para alta impedância e, por simplicidade para esta análise, a fonte de tensão de entrada (E) muda seu valor instantaneamente para zero, as condições resultantes no circuito equivalente do modelo ADC da chave são:

$$j_s[k + 1] = -G_s \cdot v_s[k] = 0 \quad \text{com} \quad v_s[k] = 0; i_s[k] = 0. \quad (2.18)$$

O que resulta em:

$$E_{C-s} = 0,5 \cdot C \cdot v_s[k]^2 = 0. \quad (2.19)$$

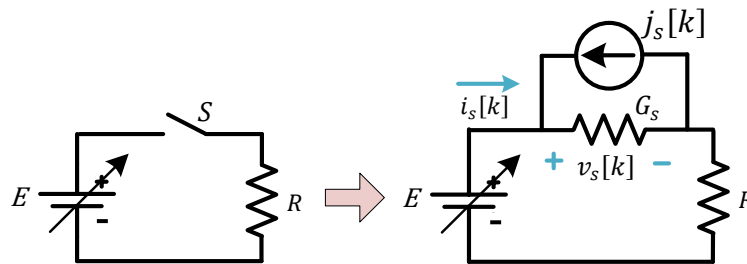


Figura 2.7 –Circuito simplificado para ilustrar o problema de perda de potência virtual no método ADC.

Ao comparar (2.17) com (2.19) se observa que a energia entregue pela fonte para o modelo da chave durante o estado *On* (que um modelo ideal de chave não demandaria) desapareceu instantaneamente durante a comutação *On/Off*. De fato, cada vez que acontece uma mudança de estado no modelo ADC, um capacitor ou indutor desaparece da simulação e, junto com ele, sua energia armazenada. Uma vez que estas perdas de potência virtuais estão associadas à frequência de chaveamento, as perdas de energia equivalentes podem ser muito maiores do que as perdas de um conversor real para altas frequências de chaveamento, o que se traduz em uma representação não realística do modelo [52].

Estratégias para aprimorar o desempenho do método ADC

A relevância dos mencionados problemas do método ADC depende, em parte, dos valores numéricos do indutor e capacitor equivalentes dos estados *On/Off*. Em termos gerais, quanto menores sejam as impedâncias fictícias (L e C) menores serão também as amplitudes dos transitórios numéricos e das perdas virtuais de potência originados por esta representação [50]. De (2.15) tem-se que o produto de L e C é igual ao quadrado do passo de simulação (Δt^2). Portanto, a abordagem mais direta para obter uma melhor

representação com o método ADC é a escolha de um passo de simulação tão pequeno quanto possível. Este parâmetro, contudo, está limitado nas simulações em tempo real pela capacidade da FPGA utilizada, bem como pela complexidade do sistema modelado. Neste contexto, vários trabalhos têm se focado em propor estratégias para determinar o valor de G_s que permita uma relação ótima entre L e C para o circuito modelado, a partir de um passo de simulação previamente definido.

Em [54] Gou et al. propõem um método para escolher G_s baseado em uma análise que busca diminuir as perdas de potência virtuais totais do conversor. Nesse trabalho conclui-se que, para a topologia de conversor VSC trifásico de dois níveis, G_s deve ser igual à relação entre o valor eficaz da corrente de carga e a tensão no elo CC. Razzaghi et al. propõem em [53] uma estratégia baseada na minimização numérica do erro entre os autovalores da matriz nodal que se origina com o método ADC e os autovalores das m matrizes dos m estados originadas pelo método de resistência variável. Esta estratégia, porém, é de mais difícil implementação, em especial à medida que o número de estados do sistema aumenta.

Maguire propôs em [47], utilizar um equivalente RC série em lugar de C para o estado *Off* da chave, de forma a atenuar os transitórios de tensão/corrente. Esta estratégia modifica a definição da fonte de corrente $j_s[k]$ para o estado *Off*, como mostrado a seguir:

A partir da Figura 2.8, a corrente no ramo série RC da representação da chave no estado *Off* é dada por:

$$i_{RC}(t) = C \cdot \frac{d(v_{RC}(t) - R \cdot i_{RC}(t))}{dt} \quad (2.20)$$

A discretização de (2.20) utilizando o método de Euler regressivo⁹ (método utilizado neste trabalho, conforme discussão da seção 2.1.2.1) dá como resultado:

$$i_{RC}[k] = \frac{1}{\Delta t/C + R} \cdot v_{RC}[k] + \frac{1}{\Delta t/C + R} \cdot (R \cdot i_{RC}[k] - v_{RC}[k - 1]). \quad (2.21)$$

Assim, considerando a definição do modelo ADC da Figura 2.6, os parâmetros desta representação para o estado *Off*, na abordagem de Maguire, são: $G_s = 1/(\Delta t/C + R)$ e $j_s[k] = G_s \cdot (v_s[k - 1] - R \cdot i_s[k - 1])$.

⁹ A proposta original de Maguire [47] utiliza o método Trapezoidal de integração numérica.

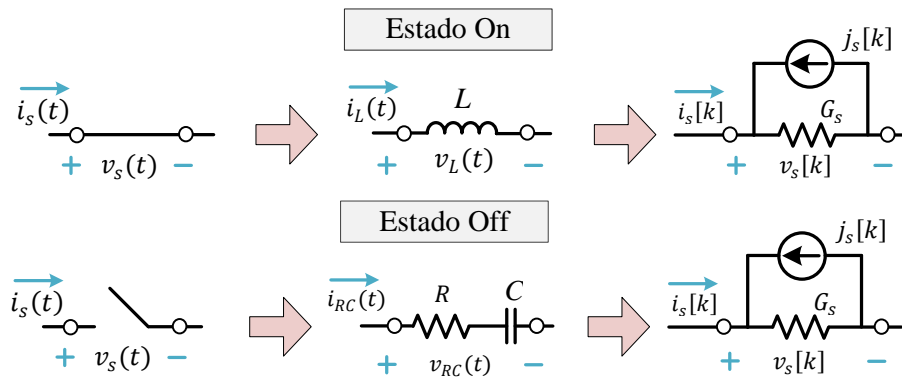


Figura 2.8 – Representação dos estados da chave na representação L/RC do método ADC.

De forma semelhante ao realizado por Gou et al. no caso da representação ADC tradicional (L/C) [54], Maguire propõe ajustar os parâmetros da representação L/RC com foco na redução das perdas de potência virtuais [47]. Em contrapartida, Dufour destaca em [52] que a resistência incluída nesta estratégia pode, em determinados casos, aumentar essas perdas.

Um problema que as estratégias focadas em determinar o valor ótimo de G_s têm em comum é o comprometimento da sua eficácia frente às variações das condições operacionais do sistema elétrico modelado [51]. Em outras palavras, a mera mudança da amplitude da corrente de carga do inversor pode ocasionar o desajuste de G_s . Este é o mesmo motivo pelo qual a compensação das perdas de potência virtuais do modelo do inversor, de forma a aproximar estas perdas àsquelas do inversor real também é pouco praticada.

Em vez de concentrar-se em determinar um valor ótimo para G_s , estratégias com diferentes enfoques têm sido propostas nos últimos anos. No trabalho de Mu et al. (2014) [55] foi proposta a introdução de uma fonte de compensação na representação equivalente de cada estado de chaveamento (uma fonte de corrente em paralelo para o estado *On* e uma fonte de tensão em série para o estado *Off*). O valor numérico dessas fontes de compensação, calculado nos instantes de chaveamento, procura que em estado estável a energia armazenada nos elementos L e C da representação da chave seja nula. Na estratégia proposta na patente de Dufour (2016) [52], aplicável a topologias de conversores com chaves operando em forma complementar (chave superior com lógica de chaveamento inversa à da chave inferior), o valor da fonte $j_s[k]$ no primeiro instante de simulação após uma mudança de estado é calculado a partir de valores históricos de tensão ou corrente da chave complementar e não da própria chave, técnica que é denominada como “inicialização cruzada”. Uma das contribuições mais recentes na

área, finalmente, é o trabalho de Wang et al. (2019) [51], que consistiu no método “ADC Generalizado” (G-ADC), no qual o valor da fonte de corrente histórica em paralelo da representação da chave da Figura 2.6 é definido como:

$$j_s[k] = \begin{cases} \alpha_{on} \cdot v_s[k-1] + \beta_{on} \cdot i_s[k-1] & \text{para chave fechada} \\ \alpha_{off} \cdot v_s[k-1] + \beta_{off} \cdot i_s[k-1] & \text{para chave aberta} \end{cases} \quad (2.22)$$

A comparação de (2.14) com (2.22) permite considerar o método ADC tradicional como um caso específico do método G-ADC, no qual $\alpha_{on} = 0$, $\beta_{on} = -1$, $\beta_{off} = 0$, e $\alpha_{off} = G_s$. No trabalho de Wang et al. a parametrização destes coeficientes é feita utilizando a abordagem de “melhor amortecimento” que dá como resultado, para o caso do inversor trifásico VSC de dois níveis, as seguintes definições:

$$\begin{aligned} \alpha_{on} &= (-1 - \sqrt{2}) \cdot G_s; & \beta_{on} &= -1; \\ \alpha_{off} &= G_s; & \beta_{off} &= (1 - \sqrt{2}). \end{aligned} \quad (2.23)$$

Adicionalmente, no algoritmo de simulação do trabalho de Wang et al. [51] é proposta a lógica da Figura 2.9 para cálculo da corrente $j_s[k]$ que, de forma semelhante à técnica de “inicialização cruzada” do trabalho de Dufour [52], busca reduzir o erro inicial após uma comutação de estado. Nesta lógica, o valor da corrente histórica no primeiro passo de simulação após um chaveamento para o estado *On* toma o valor da fonte histórica do último instante do anterior estado *On*. Uma alocação correspondente é feita no caso do estado

```

if (S[k] = S[k - 1])
    j_s[k] = resultado de (2.22)
    if (S[k] = 1)    j_s-ONmemoria = j_s[k];
    else            j_s-OFFmemoria = j_s[k];
    end
else
    if (S[k] = 1)   j_s[k] = j_s-ONmemoria;
    else           j_s[k] = j_s-OFFmemoria;
    end
end

```

Figura 2.9 – Lógica para determinação das correntes históricas no método G-ADC.

2.3. Técnicas complementares propostas nesta tese

Os métodos para modelagem chaveada de chave ideal, resistência variável e ADC foram objeto das análises preliminares desta tese. Estes três métodos foram

implementados no desenvolvimento experimental de um SDTR utilizando um módulo FPGA genérico e um algoritmo de solução baseado na Análise Nodal Modificada¹⁰ (a descrição do SDTR desenvolvido é apresentada no Capítulo 3), a partir do qual foi possível verificar a melhor adequação do método ADC para a aplicação de simulação em tempo real, bem como seus problemas de exatidão anteriormente mencionados. Foi corroborado, adicionalmente, o bom desempenho do método G-ADC proposto por Wang et al. [51] para atenuar os erros do método ADC tradicional, aprimorando seu desempenho de forma significativa. No entanto, percebeu-se que o método G-ADC faz uso de maiores recursos computacionais da FPGA para o mesmo sistema digital, uma vez que o cálculo da corrente histórica $j_s[k]$ requer mais operações matemáticas do que no caso ADC tradicional (ver (2.14) e Figura 2.9).

Como descrito na Seção 2.2.2.4, o método G-ADC [51] se caracteriza por duas inovações (em relação ao método ADC tradicional [50]): i) generalização do cálculo da corrente $j_s[k]$ para os estados *On* e *Off* e o ajuste de parâmetros para obter a resposta com melhor amortecimento, e ii) proposição de uma nova metodologia para inicialização das correntes históricas no instante de chaveamento. Neste sentido, uma análise que não foi apresentada no trabalho de Wang et al. [51] nem em outra referência técnica disponível na literatura é a comparação das inovações i) e ii) de forma individual, no que se refere ao aprimoramento que cada uma delas traz ao método ADC, bem como ao incremento de recursos computacionais demandado por cada uma delas. Essa lacuna é preenchida neste trabalho.

Para realizar as comparações acima mencionadas, duas técnicas complementares, baseadas no método G-ADC, são propostas: a) ADC com inicialização (ADC-I) no qual a corrente $j_s[k]$ é calculada por um algoritmo de inicialização similar ao apresentado na Figura 2.9, porém usando (2.14) em vez da (2.22); e b) G-ADC sem inicialização (G-ADC-SI) no qual o cálculo da corrente $j_s[k]$ utiliza a definição generalizada de (2.22), porém nenhuma lógica de inicialização das fontes de corrente é empregada.

2.3.1. Validação mediante simulação *off-line*

Para validar o desempenho das técnicas propostas, no que se refere a suas exatidões, é utilizado o sistema elétrico da Figura 2.10. Neste sistema um conversor trifásico VSC

¹⁰ A implementação de outros algoritmos de solução, por exemplo o método de espaço de estados, está fora do foco deste trabalho.

de dois níveis é conectado a uma fonte CC no seu barramento de corrente contínua e a uma carga resistiva no lado CA. Este conversor utiliza um indutor (L_f), cuja resistência dos enrolamentos é representada por R_f , para atenuar as componentes de alta frequência (devido ao PWM) nas ondas de corrente (i_a , i_b e i_c).

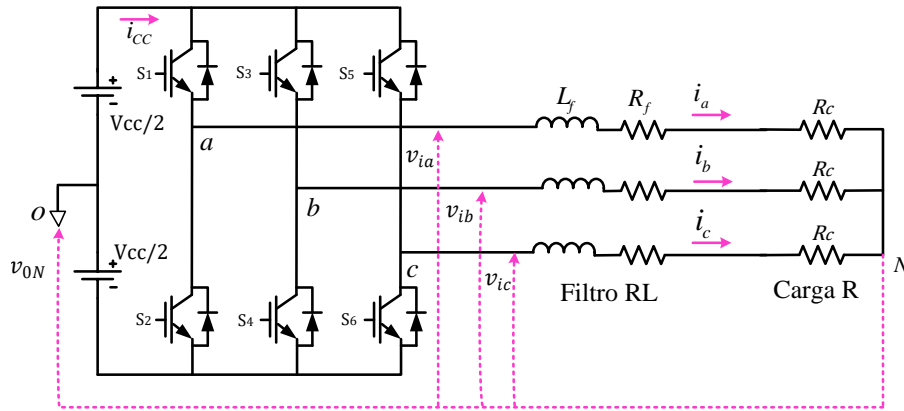


Figura 2.10 – Sistema elétrico utilizado na validação *off-line* de exatidão.

Os parâmetros deste sistema são dados na Tabela 2.2. A indutância do filtro foi especificada de tal forma que a reatância equivalente na frequência de 60 Hz seja aproximadamente 5% da impedância base (com $P_{base} = 250$ kVA e $V_{base} = 440$ V), cujo valor real é $0,77 \Omega$. O fator de qualidade do indutor, X_L/R_f , foi definido em 25.

Tabela 2.2– Parâmetros do sistema elétrico da Figura 2.10.

Grandeza	Símbolo	Valor	Grandeza	Símbolo	Valor
Tensão nominal elo CC	V_{cc}	800 V	Potência base	P_{base}	250 kVA
Tensão base CA	V_{base}	440 V			
Resistência filtro	R_f	1,55 m Ω	Indutância filtro	L_f	102,7 μ H
Frequência de chaveamento	f_{ch}	10 kHz	Resistência de carga	R_c	0,77 Ω

Um algoritmo de solução baseado na Análise Nodal Modificada (ver Seção 2.1.2) com passo de solução de 1μ s foi desenvolvido no ambiente Simulink[®] para simular o sistema da Figura 2.10 utilizando (de forma independente) cinco técnicas de representação das chaves: 1) método de chave ideal (definido na Seção 2.2.2.2), 2) ADC, 3) G-ADC, 4) ADC-I e 5) G-ADC-SI. A equação matricial de solução do sistema para as técnicas baseadas no método ADC (técnicas 2) a 5)) é mostrada no Apêndice B. Para tais técnicas, o valor de G_s foi definido como $G_s = I_o/V_{cc}$, conforme indicado pelo trabalho de Gou et al. [54], onde I_o é o valor eficaz nominal da corrente de carga.

Um controlador simplificado em laço aberto que cria ondas de referência senoidais balanceadas, com amplitude de 0,85 pu (na base da amplitude da onda portadora

triangular do modulador PWM) e frequência de 60 Hz é utilizado como entrada para um modulador PWM, com frequência de 10 kHz, que estabelece pulsos de chaveamento idênticos para todos os métodos de modelagem.

A Figura 2.11(a) mostra as correntes trifásicas do lado CA do inversor para o caso da modelagem por chave ideal, que neste exercício é considerado o método *benchmark* para fins de comparação. Assim, a Figura 2.11(b) mostra o erro instantâneo de corrente por fase para cada uma das técnicas baseadas no método ADC utilizadas. Observa-se que a técnica com maior erro instantâneo é ADC tradicional, no qual o valor pico das ondas de erro é próximo de 30 A, enquanto que o método G-ADC é aquele que determina resultados mais próximos aos do caso ideal.

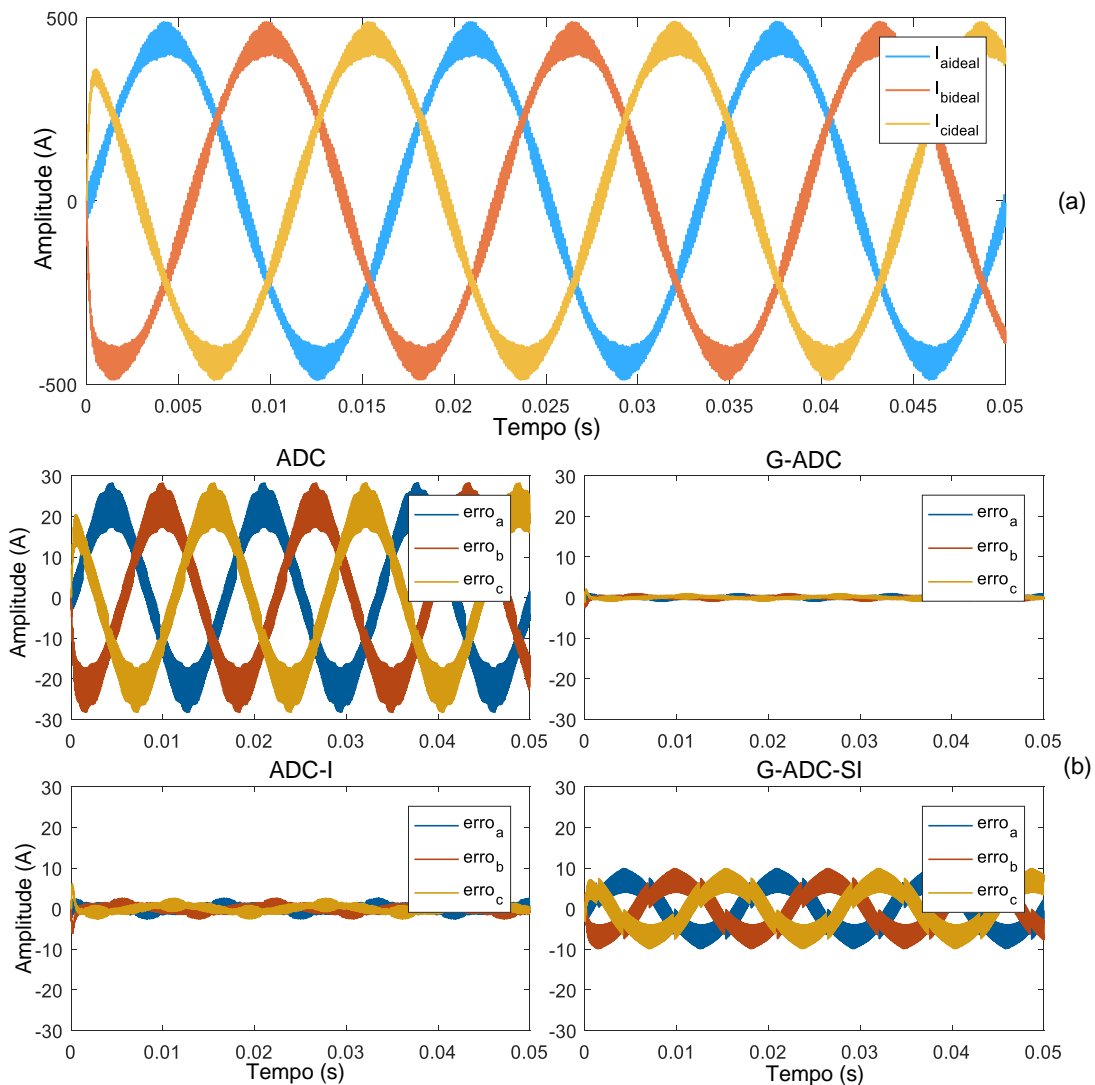


Figura 2.11 –Correntes no lado CA do inversor: (a) formas de onda calculadas pelo método de chave ideal (conforme definição da Seção 2.2.2.2), (b) formas de onda dos erros de corrente para cada um dos outros métodos empregados.

Nas figuras acima os erros instantâneos são calculados como:

$$erro_{a,b,c}^X = i_{a,b,c\ ideal} - i_{a,b,c}^X \quad (2.24)$$

Onde X é o método de representação de chaves considerado no cálculo (ADC, G-ADC, ADC-I ou G-ADC-SI).

De forma a obter um indicador da exatidão de cada técnica de representação, foi definido o erro (ε_x) que, conforme mostra (2.25), consiste na média da normatização dos valores eficazes dos erros instantâneos em função do valor eficaz da corrente ideal.

$$\varepsilon_x = \left(\frac{rms(erro_a^X)}{rms(i_{a\ ideal})} + \frac{rms(erro_b^X)}{rms(i_{b\ ideal})} + \frac{rms(erro_c^X)}{rms(i_{c\ ideal})} \right) / 3 \quad (2.25)$$

O resultado da avaliação de (2.25) é apresentado na Tabela 2.3.

Tabela 2.3– Avaliação de exatidão das abordagens de representação de chaves.

Método	Erro corrente CA (ε_x)
ADC tradicional	4,81%
G-ADC	0,07%
ADC-I (Proposta)	0,24%
G-ADC-SI (Proposta)	1,54%

Os resultados anteriores evidenciam o aprimoramento que o método G-ADC proporciona em relação ao método ADC tradicional. As estratégias propostas nesta tese têm também um desempenho melhor do que o método ADC tradicional. Entretanto, observa-se que a estratégia ADC-I, que utiliza um algoritmo de inicialização das fontes de corrente é mais exata do que a estratégia G-ADC-SI. Ainda, a performance da estratégia ADC-I é próxima à obtida pelo método G-ADC, mesmo com uma formulação mais simples para determinar os valores das fontes históricas $j_s[k]$. No Capítulo 3 é realizada uma comparação do esforço computacional destas técnicas explorando esta característica.

Com intuito ilustrativo, a Figura 2.12 mostra visões ampliadas da superposição das ondas de corrente da fase a calculadas pelos métodos ADC, G-ADC, ADC-I e G-ADC-SI na correspondente onda calculada pelo método de chave ideal ($i_{a\ ideal}$). Observa-se que em termos práticos os resultados para as ondas de corrente com os métodos G-ADC e ADC-I podem considerar-se iguais aos do método de chave ideal para o sistema elétrico em estudo.

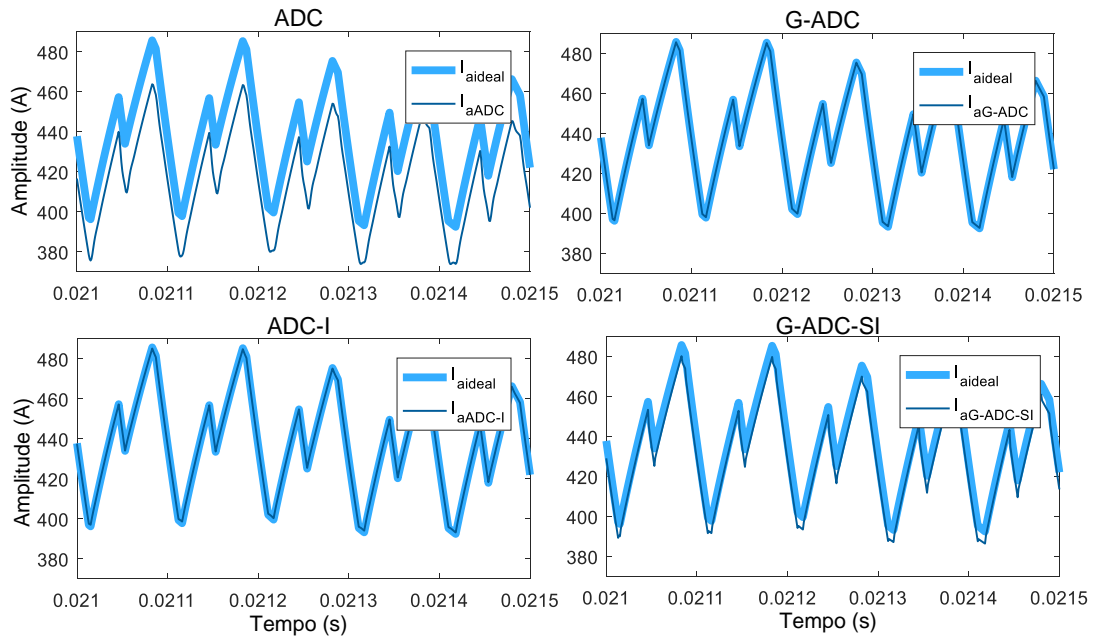


Figura 2.12 – Visão ampliada das comparações entre a corrente i_a calculada pelo método de chave ideal e a mesma corrente no caso dos métodos de representação ADC, G-ADC, ADC-I e G-ADC-SI.

A Figura 2.13 ilustra a comparação entre as formas de onda da tensão v_{ia} para o caso ideal e para cada uma das outras técnicas consideradas nesta análise, por um período equivalente a quatro ciclos PWM. Na Figura 2.13(a) (caso ADC) observam-se transitórios numéricos amortecidos após cada evento de chaveamento, comportamento discutido teoricamente na Seção 2.2.2.4 e que repercute negativamente na exatidão desta técnica, como mostra a Tabela 2.3. Oscilações com maiores valores de pico¹¹, porém atenuação mais rápida são mostradas na Figura 2.13(d) (caso G-ADC-SI). Os métodos com menor indicador de erro na Tabela 2.3 (G-ADC e ADC-I), por outro lado, apresentam oscilações numéricas de amplitude reduzida, ou seja, consistem em resultados mais próximos dos ideais também no que se refere às tensões na saída do conversor.

A visão detalhada da Figura 2.14 permite comparar os resultados para os casos G-ADC e ADC-I. Conforme mostrado, o método G-ADC tem um transitório numérico com melhor amortecimento, sendo necessários apenas dois passos de simulação após a mudança de estado para que a onda de tensão chegue ao seu valor final. O método ADC-I apresenta uma atenuação mais lenta, contudo, os valores de pico dos transitórios numéricos são inferiores a 4% da tensão ideal (assim como no caso G-ADC), o que

¹¹ Na Figura 2.13 são utilizadas diferentes escalas verticais para cada diagrama.

torna estas oscilações suficientemente pequenas para não comprometer a exatidão da estratégia, como mostrou a Tabela 2.3

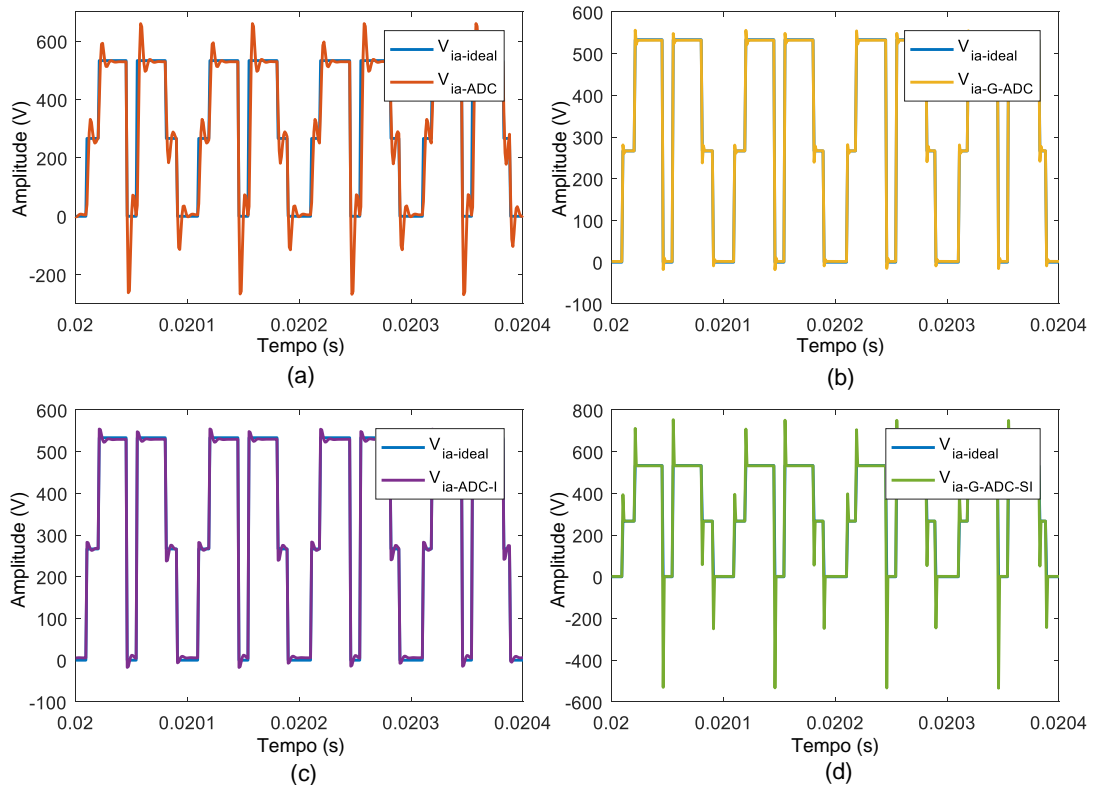


Figura 2.13 – Comparação entre a forma de onda da tensão v_{ia} obtida com o método ideal e as resultantes com os métodos: (a) ADC, (b) G-ADC, (c) ADC-I, (d) G-ADC-SI.

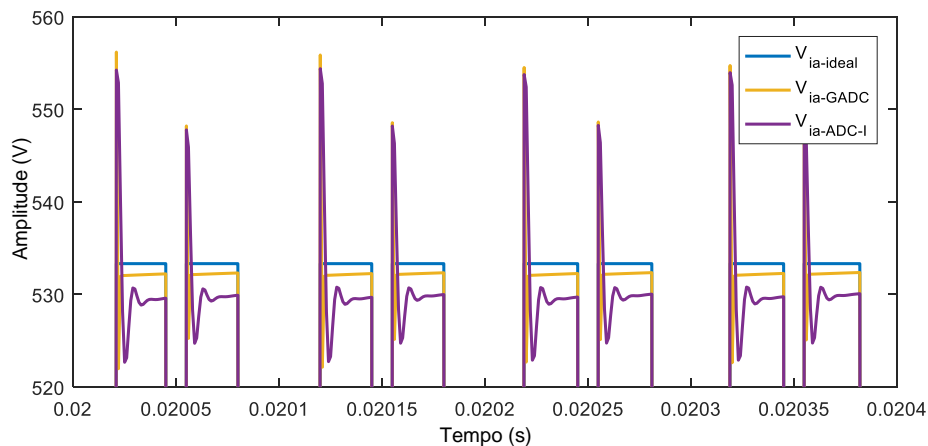


Figura 2.14 – Zoom vertical na comparação entre a forma de onda da tensão v_{ia} obtida com o método ideal e as resultantes com os métodos G-ADC e ADC-I.

A análise anterior indica que o algoritmo de inicialização da Figura 2.9 para determinar $j_s[k]$ é mais eficiente do que o uso da expressão generalizada de (2.22), com ajuste por (2.23), na redução da amplitude dos transitórios numéricos de comutação, uma vez que a estratégia ADC-I é mais exata do que a estratégia G-ADC-SI. Para complementar esta análise, no Capítulo 3 desta tese é realizada uma comparação dos recursos de FPGA requeridos por todas as técnicas discutidas.

Capítulo 3

Desenvolvimento de SDTR baseado em módulo FPGA genérico

Este capítulo apresenta a metodologia utilizada para desenvolver um SDTR no qual um módulo FPGA genérico, contido numa plataforma de medição e instrumentação, é empregado como núcleo computacional. Esse desenvolvimento é acompanhado de análises comparativas das características de compilação, execução e desempenho das técnicas de modelagem chaveada ADC, G-ADC, ADC-I e G-ADC-SI, descritas nas Seções 2.2.2.4 e 2.3. O SDTR desenvolvido foi acoplado em laço fechado a um microcontrolador DSP, de forma a criar uma bancada HIL em tempo-real focada em avaliar o desempenho de controladores de conversores eletrônicos. Resultados experimentais de simulação em tempo real de um sistema elétrico que contém um conversor VSC trifásico de dois níveis conectado à rede, emulado no SDTR com passo de solução de 1 μ s, são apresentados.

Considerando as especificações do *hardware* utilizado para desenvolver o SDTR proposto (ver Seção 3.1), no qual a exportação de sinais analógicos é realizada com uma taxa inferior à frequência de cálculo, foi necessário incluir na programação do SDTR uma técnica focada em minimizar os erros originados pelo fenômeno de *Aliasing* nos sinais analógicos de saída. Para este fim, uma abordagem de sincronização entre o DSP e o SDTR é proposta e validada neste capítulo. A proposição e aplicação desta abordagem constitui a segunda contribuição desta tese em relação aos trabalhos similares anteriormente desenvolvidos, nos quais são utilizados SDTR com capacidade de exportar sinais analógicos, ou de transferir os correspondentes dados através de protocolos digitais, na frequência de cálculo, com atualização a cada passo de simulação.

3.1. Descrição do *hardware* utilizado

O SDTR desenvolvido neste trabalho utiliza como núcleo computacional o módulo FPGA referência PXIe-7965R da *National Instruments*, cujos principais recursos são

apresentados na Tabela 3.1, na qual *DSP slices* se refere ao número de blocos multiplicadores e acumuladores predefinidos e LUTs significa tabelas *LookUp*.

Tabela 3.1– Principais características do módulo PXIe-7965R.

FPGA: Modelo Virtex-5 SX95T	LUTs/Flip-Flops: 58880
FPGA DSP slices: 640	FPGA Memory: 8784 kbit
Onboard Memory : 512 MB(DRAM)	Relógio padrão: 40 MHz
Periféricos no terminal frontal	
Entrada analógica: 16 canais (2 V _{pp} , 14-bits)	Entrada digital: 8 canais (0 a 3.5 V)
Saída analógica: Não tem	Saída digital: 8 canais (0 a 3.5 V)
Periféricos no terminal traseiro: Protocolo PXIe	
Processamento de dados utilizando matemática de ponto fixo	

Apesar de se tratar de uma unidade de processamento digital com operação independente, este módulo FPGA está inserido em uma plataforma PXI, equipamento modular de instrumentação composto por três componentes básicos: chassi, controlador e módulos periféricos [89]. Assim, o *hardware* utilizado nesta tese como SDTR/FPGA é a plataforma PXI apresentada na Figura 3.1, formada pelos seguintes componentes:

- a) Chassi: Ref. PXIe-1082 de 8-slots;
- b) Controlador: Ref. PXIe-8133, com processador Intel Core i7 820 QM;
- c) Módulo periférico 1: FPGA, Ref. PXIe-7965R, com FPGA Virtex-5;
- d) Módulo periférico 2: Conversor D/A, Ref. NI 6733, 16 bits, 8 canais, ± 10 V_{pico}.

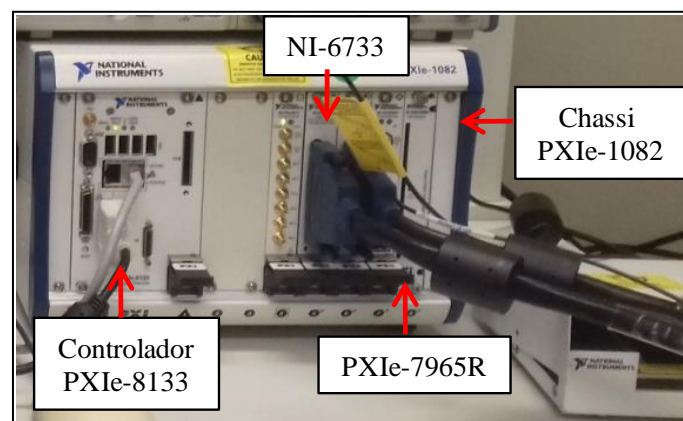


Figura 3.1 – Plataforma PXI utilizada como *hardware* do SDTR/FPGA.

O segundo periférico (componente d), conversor D/A) foi requerido como consequência da falta de canais de saída analógica no módulo FPGA (ver Tabela 3.1). Isto porque para desenvolver uma bancada HIL como a apresentada conceitualmente na Figura 1.3, objetivo parcial deste trabalho, se torna necessário dispor de sinais analógicos na saída do SDTR. Outras abordagens, utilizando terminais digitais para

exportação das grandezas calculadas pelo SDTR (por exemplo, através de protocolos de comunicação ou sinais digitais modulados) estão fora do foco desta tese.

O uso de plataformas PXI para o desenvolvimento de SDTR de baixo custo tem sido reportado na literatura. Nesses SDTR, tanto métodos de representação de conversores de valor médio [90], [91] quanto chaveados [51] têm sido aplicados. Existem, ainda, empresas que oferecem *software* de simulação em tempo real, baseados no método ADC para representação de chaves, que podem ser configurados para rodar em plataformas PXI [92], [93]. Em todos estes casos, entretanto, o SDTR dispõe da capacidade de exportar os sinais calculados na frequência de operação, o que não é possível no SDTR proposto neste trabalho, conforme discussão da Seção 3.1. Esta característica torna o simulador proposto um trabalho original na área.

3.2. Algoritmo de solução

Conforme discussão da Seção 2.1.2.1, nesta tese utiliza-se um algoritmo de solução (simulação EMT) que emprega a técnica Análise Nodal Modificada, onde os circuitos discretos associados dos elementos modelados são determinados utilizando o método de integração numérica Euler regressivo.

Quatro métodos de representação de chaves, todos baseados no método ADC, são comparados: ADC, G-ADC, ADC-I e G-ADC-SI. Uma vez que para todos estes métodos é selecionado o mesmo valor de condutância em paralelo (G_s), a matriz representativa do sistema (\mathbf{H} , que é constante independentemente dos estados das chaves modeladas) é a mesma para todos os métodos. Assim, a determinação da inversa de \mathbf{H} (\mathbf{H}^{-1}) é realizada de forma *off-line*, e a mesma é programada no módulo FPGA como uma constante matricial.

Um fluxograma do algoritmo executado no módulo FPGA para obter a solução do sistema, para cada passo de simulação, é apresentado na Figura 3.2. Conforme mostrado, a primeira etapa consiste em adquirir os pulsos de chaveamento $S[k]$ através dos canais de entrada digital disponíveis. A seguir, as correntes históricas dos circuitos discretos associados ($j_x[k]$) são calculadas. Para isto são necessárias informações de tensões e correntes no instante anterior da simulação bem como do tipo de elemento modelado. No destaque da parte direita da figura é mostrado que para os elementos passivos armazenadores (capacitor/indutor) o valor da fonte de corrente histórica em paralelo é determinado pelas expressões da Tabela 2.1, enquanto que para as chaves, o

cálculo de $j_s[k]$ depende do método de representação utilizado em cada caso da análise comparativa:

- (1) ADC: $j_s[k] =$ resultado de (2.14);
- (2) G-ADC: $j_s[k] =$ lógica da Figura 2.9 usando (2.22);
- (3) ADC-I: $j_s[k] =$ lógica da Figura 2.9 usando (2.14);
- (4) G-ADC-SI: $j_s[k] =$ resultado de (2.22).

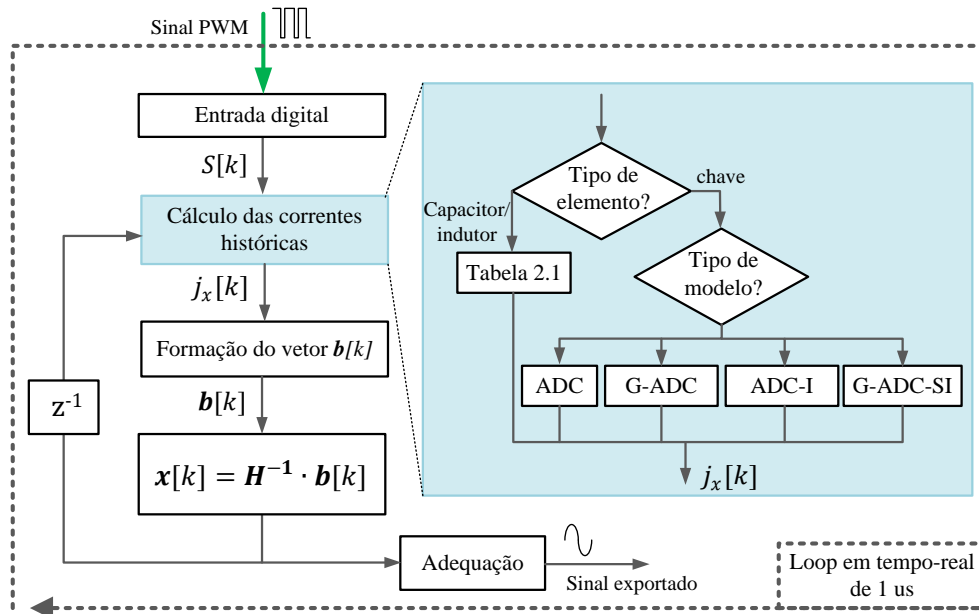


Figura 3.2 – Diagrama de fluxo do algoritmo utilizado para simulação EMT.

O seguinte passo do algoritmo é o de formar o vetor de fontes de correntes conhecidas $\mathbf{b}[k]$, requerido para executar a multiplicação matricial $\mathbf{H}^{-1} \cdot \mathbf{b}[k]$ cujo resultado é o valor atual do vetor $\mathbf{x}[k]$, finalizando assim um ciclo do laço de simulação. O bloco denominado “adequação” na figura é assunto de discussão da próxima seção. O algoritmo descrito é executado com passo de execução de $1 \mu\text{s}$, portanto a discretização dos elementos do modelo também é feita utilizando este Δt .

3.3. Abordagem para minimizar o fenômeno de *Aliasing* nas ondas analógicas de saída

Conforme mencionado na Seção 3.1, a exportação de sinais analógicos no SDTR proposto não é realizada diretamente pelo módulo FPGA (componente do simulador onde é efetuada a solução do sistema), mas por um conversor D/A acoplado à plataforma PXI que serve como *hardware* do SDTR. Neste sentido, uma representação da transferência de dados que se dá em uma bancada HIL na qual um controlador digital

seja interconectado ao SDTR proposto é apresentada na Figura 3.3. Nesta figura é mostrado que a interface de entrada do SDTR, na qual são amostrados os pulsos de chaveamento, e o algoritmo da Seção 3.2 são ambos implementados no módulo FPGA, com passo de $1 \mu\text{s}$ (Δt). Por outro lado, a exportação dos sinais analógicos calculados pelo algoritmo EMT utiliza a combinação de dois recursos: i) o controlador do sistema PXI (que lê digitalmente o vetor $x[k]$ e efetua ajustes nestes sinais de acordo com os limites de tensão da bancada HIL) e ii) o módulo D/A (que realiza a conversão). Conforme mostra a figura, o período de execução equivalente desta interface de saída é definido como T_{ex} .

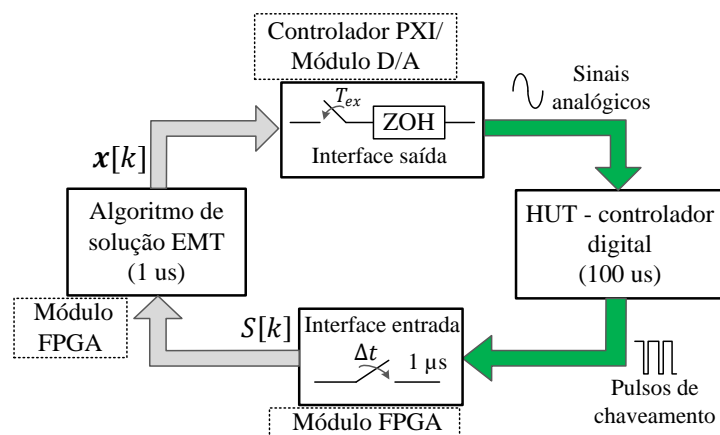


Figura 3.3 – Representação da transferência de sinais para uma bancada HIL na qual as chaves modeladas no SDTR proposto sejam controladas por um HUT. Nomenclatura: seta cinza: sinais digitais internos ao SDTR; seta verde: sinais externos ao SDTR; retângulo em linha tracejada: indica em que componente(s) do SDTR é realizada cada operação; T_{ex} : passo de exportação de sinais; ZOH: *Zero-order hold*.

A medição experimental de T_{ex} deu como resultado um período com média $29,5 \mu\text{s}$ e desvio padrão de $6,8 \mu\text{s}$. Considerando o teorema de Nyquist [94], este tipo de interface não poderia ser utilizado, uma vez que originaria, se nenhuma providência for tomada, a aparição do fenômeno de *Aliasing* nas ondas chaveadas exportadas. Entretanto, nesta tese é proposta uma abordagem de adequação de sinais que objetiva permitir a aplicação do SDTR/FPGA em estudos de avaliação de controladores embarcados de conversores eletrônicos de potência. Essa abordagem é descrita a seguir.

3.3.1. Sincronismo entre DSP e SDTR para adequação dos sinais

Neste trabalho se propõe utilizar uma técnica de amostragem sincronizada entre o DSP e o SDTR para atenuar significativamente o fenômeno de *Aliasing* nos sinais analógicos exportados pelo SDTR. Esta técnica se baseia no método de *amostragem instantânea sincronizada*, comumente empregado em controladores digitais de

conversores eletrônicos. Conforme mostrado em [95] para o caso de VSC trifásicos com controle de corrente, o princípio de funcionamento deste método consiste em sincronizar o instante de amostragem com os pulsos PWM, de tal forma que o valor instantâneo de corrente medido represente o valor médio da onda de corrente durante tal período de chaveamento. Isto é obtido ao sincronizar o instante de amostragem com a metade do período On da chave superior do braço do conversor VSC [96].

Neste contexto, a abordagem proposta neste trabalho, representada graficamente pela Figura 3.4, é a seguinte: no DSP utilizado como controlador na bancada HIL é criado um pulso de sincronismo (“sinal de sincronismo” na figura) centralizado com o ponto médio da onda PWM de chaveamento; tal pulso digital serve como trigger para o módulo FPGA do SDTR, indicando-lhe o instante em que as variáveis devem ser salvas (escritas) em um registro de acesso direto de memória FIFO DMA [97]. Essas variáveis são posteriormente lidas pelo controlador PXI e exportadas pelo conversor D/A em um período (T_{ex}). Uma vez que T_{ex} é menor do que um período de chaveamento (que é de 100 μ s), o registro de memória FIFO DMA fica vazio antes da ocorrência do próximo pulso de sincronismo, quando é repetido o procedimento, evitando perda de dados.

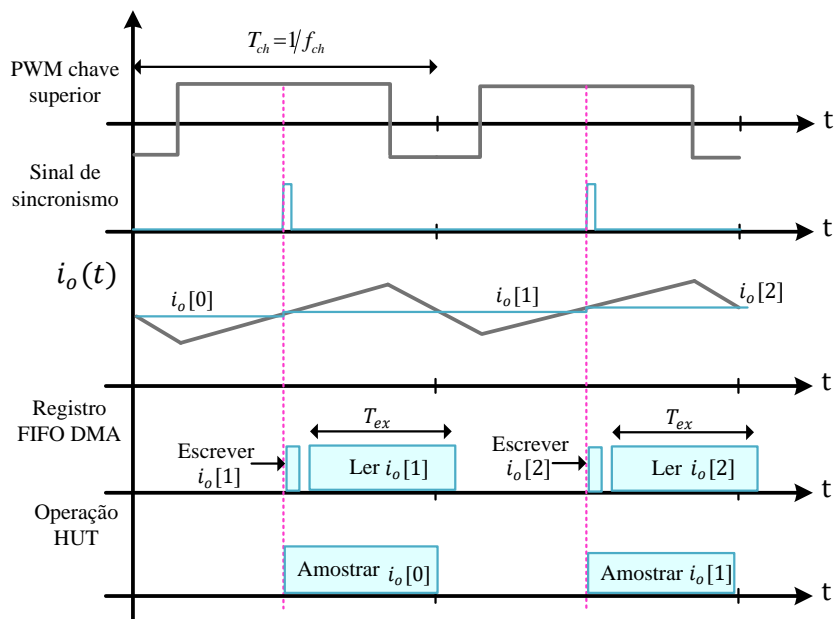


Figura 3.4 – Representação da segunda abordagem, que usa sincronismo entre o HUT e o SDTR. Considerando como exemplo o caso de um conversor de meia ponte.

A parte inferior da Figura 3.4 descreve a amostragem efetuada pelo HUT (DSP com controle embarcado) dentro do contexto da abordagem de adequação de sinais. No HUT é implementada a amostragem instantânea sincronizada, razão pela qual a figura mostra que o instante de amostragem do HUT também está em sincronismo com a metade do

pulso PWM. Entretanto, observa-se que existe um atraso equivalente a um período de chaveamento entre os sinais calculados pelo algoritmo do SDTR e os sinais disponíveis nos terminais analógicos, como consequência da abordagem de adequação proposta. Este atraso adicional prejudica a exatidão do SDTR para aplicações com controle em malha fechada, pelo que a relevância da inexatidão introduzida será avaliada com os casos de estudo apresentados na Seção 3.5.3.

Por outro lado, a análise de desempenho em regime permanente desta abordagem de adequação é mostrada na Seção 3.5.2.3.

3.4. Metodologia de programação do SDTR proposto

Dado que o SDTR proposto contém dois componentes programáveis (módulo FPGA e controlador PXI, ver Seção 3.1), a descrição da metodologia de programação das funcionalidades do SDTR, realizada nesta seção, apresenta cada função de forma individual, separando as tarefas realizadas pelo módulo FPGA das realizadas pelo controlador PXI.

Neste trabalho o controlador PXI é programado utilizando o *software* LabVIEW, enquanto que o módulo FPGA com o *software* LabVIEW FPGA, que tem como característica um maior nível de abstração do que a programação em linguagem de descrição de *hardware* (VHDL), uma vez que neste *software* o arquivo *bitstream* com as instruções das interconexões entre os componentes da FPGA é criado de forma automática, no processo de compilação.

3.4.1. Solução do sistema

A execução do algoritmo de solução do sistema, descrito na Seção 3.2, é realizada inteiramente no módulo FPGA. Esse algoritmo é implementado no *software* LabVIEW FPGA utilizando operações matemáticas básicas, uma multiplicação matricial (para executar o cálculo de $\mathbf{H}^{-1} \cdot \mathbf{b}[k]$) e estruturas do tipo *case*; tudo isso dentro de um laço cujo período de execução é controlado mediante um contador de *ticks* do relógio interno da FPGA (com frequência configurada em 60 MHz). Da programação efetuada merece destaque o método utilizado para implementar o algoritmo de cálculo da corrente em paralelo $j_s[k]$, que é realizado com a estrutura *case* de quatro possíveis estados da Figura 3.5.

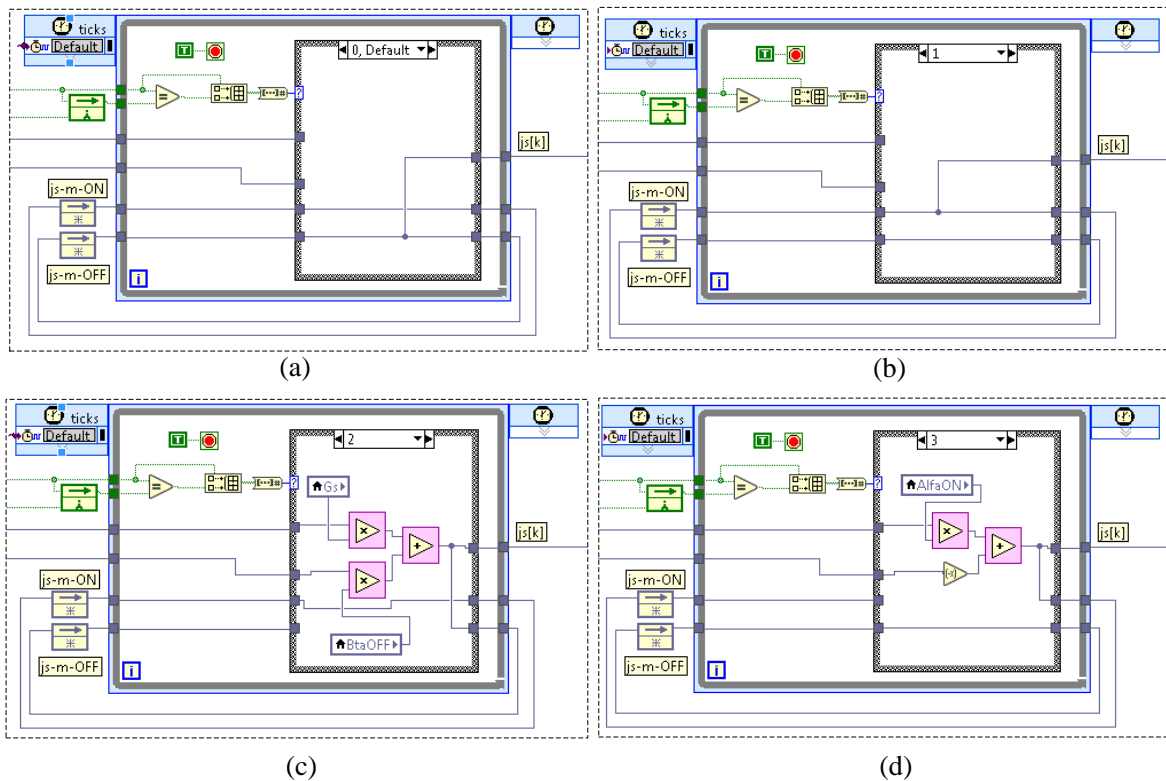


Figura 3.5 – Programação do cálculo da corrente histórica $j_s[k]$ em LabVIEW FPGA considerando o método G-ADC: (a) $S[k] = 0$ e $S[k - 1] = 1$; (b) $S[k] = 1$ e $S[k - 1] = 0$; (c) $S[k] = 0$ e $S[k - 1] = 0$; (d) $S[k] = 1$ e $S[k - 1] = 1$.

Salienta-se que a anterior figura é relativa ao caso de modelagem de chaves pelo método G-ADC, que é o mais complexo dos considerados nesta tese, os outros métodos (ADC, ADC-I e G-ADC-SI) foram implementados mediante versões simplificadas dos diagramas da Figura 3.5.

3.4.2. Conversão D/A

Conforme apresentado na Seção 3.3, uma abordagem de sincronismo entre o SDTR e o DSP é proposta nesta tese para realizar a adequação dos sinais a serem exportados pelo módulo D/A. Para isso, são requeridas programações tanto no módulo FPGA quanto no controlador do sistema PXI, como será descrito a seguir.

3.4.2.1. Programação no módulo FPGA

Para criar o sincronismo entre DSP e SDTR, descrito na Seção 3.3.1, um canal de entrada digital é acoplado em *hardware* ao pulso periódico fornecido pelo DSP. A partir da leitura desse canal digital, na frequência de 1 MHz, uma lógica simples que identifica a borda de subida desse sinal, para determinar o instante de cálculo no qual as variáveis de saída são armazenadas (“escritas”) no registro FIFO DMA, é programada. Nos outros

instantes de simulação nenhuma operação é realizada. A programação no *software* LabVIEW FPGA desta lógica é mostrada na Figura 3.6.

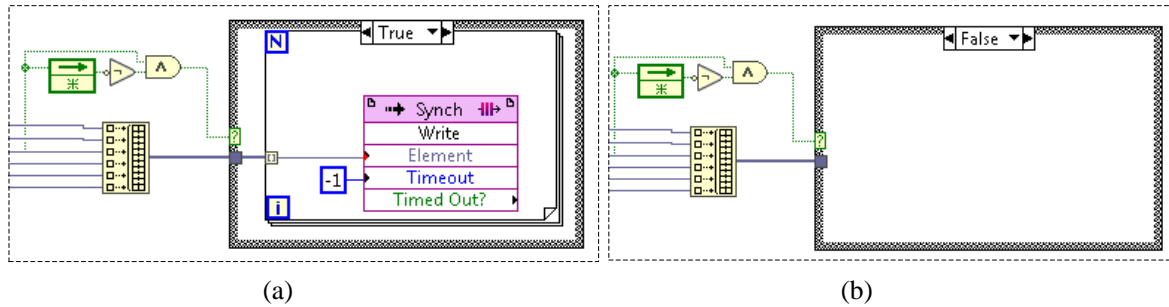


Figura 3.6 – Programação da lógica de sincronismo DSP-SDTR no módulo FPGA: (a) operação quando é identificada uma borda de subida no sinal de sincronismo; (b) operação nos outros instantes.

3.4.2.2. Programação no controlador do sistema PXI

No sincronismo projetado, os dados escritos no registro FIFO DMA pelo módulo FPGA são lidos pelo controlador do PXI e transferidos ao módulo D/A, que os exporta. Este procedimento é programado pela estrutura tipo *while* da Figura 3.7, no qual a operação de leitura unicamente é executada após uma bandeira digital indicar a existência de novos dados no registro FIFO, garantindo o desejado sincronismo. Os dados digitais lidos são ajustados em magnitude e *offset* de acordo com as características de entrada e saída do módulo D/A e do HUT (e de eventuais adequadores de sinais, caso estejam disponíveis) e transferidos para o registro do conversor D/A (“DAQmx” na figura). As variáveis ajustadas em magnitude e *offset* são referenciadas como “variáveis representativas” do sistema simulado, representadas simbolicamente pelo sobrescrito *D/A*, por exemplo, $i_a^{D/A}$.

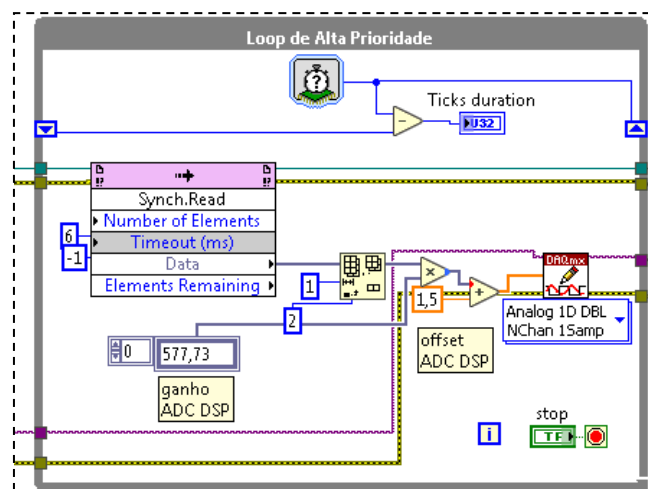


Figura 3.7 – Programação da lógica de sincronismo DSP-SDTR e exportação de sinais analógicos em LabVIEW.

3.4.3. HMI

Uma interface homem máquina foi programada no controlador do sistema PXI. Esta utiliza comunicação *Ethernet* para transmitir as variáveis de interesse a um computador. Neste trabalho, um “instrumento virtual” (VI) do *software* LabVIEW é utilizado como meio de interface, nele é monitorado o comportamento da simulação. A taxa de atualização dos dados apresentados nessa interface depende da latência do canal de comunicação Ethernet, não sendo em tempo real.

3.4.4. Armazenamento de formas de onda

Além do registro FIFO DMA utilizado para adequar os sinais a serem exportados, um segundo registro FIFO DMA é empregado no SDTR proposto. Este tem a função de armazenar as variáveis de interesse usando a resolução temporal do algoritmo de cálculo (1 μ s) e uma janela de um segundo de duração (o que representa um milhão de dados por variável armazenada). Os dados salvos neste registro FIFO DMA são lidos pelo controlador da PXI e transmitidos para o computador de interface, através da comunicação Ethernet, após termino da simulação, o que permite a análise *off-line* detalhada das ondas chaveadas. A visualização das variáveis de forma online, sujeita à latência da comunicação Ethernet, também pode ser realizada.

3.5. Validação do SDTR proposto

3.5.1. Descrição da bancada HIL

A bancada HIL utilizada neste trabalho para fins de validação do SDTR é ilustrada na fotografia da Figura 3.8. Essa bancada é composta pelo SDTR proposto e por um DSP, modelo TMS320F28337s da *Texas Instruments*, que é conectado em laço fechado através de sinais analógicos e discretos. Tomando como referência a estrutura conceitual de uma bancada HIL de controle da Figura 1.3, pode-se dizer que o DSP faz o papel do equipamento sob ensaio (HUT). Tal HUT comanda as chaves modeladas no SDTR através do estabelecimento de seus pulsos de chaveamento, o que é realizado, no caso de um controle em malha fechada, a partir da amostragem e tratamento matemático das variáveis representativas do sistema, estas exportadas como sinais analógicos pelo SDTR/FPGA. Conforme descrição da Seção 3.3.1, o pulso de sincronismo utilizado para realizar a adequação dos sinais a serem exportados também é gerado pelo DSP. A frequência de operação (amostragem e chaveamento) do DSP é de 10 kHz.

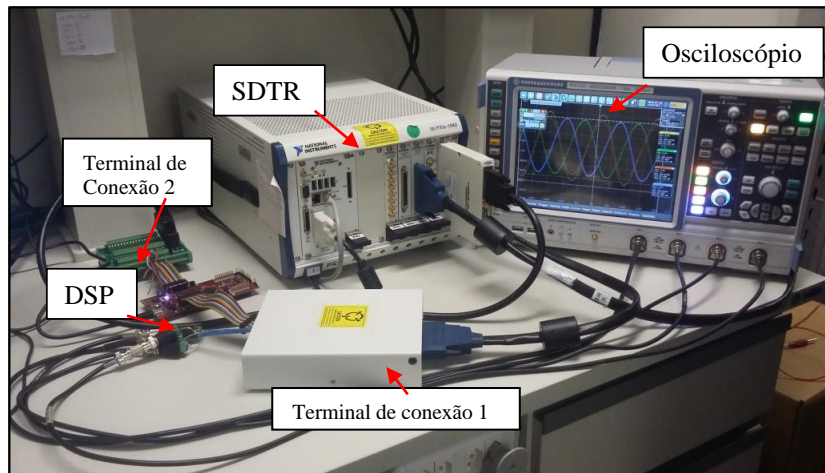


Figura 3.8 – Fotografia da bancada HIL. SDTR: Simulador Digital em Tempo Real desenvolvido; DSP: processador digital de sinais.

3.5.2. Simulação de conversor VSC operando em malha aberta

Neste ensaio foi reproduzida a simulação do circuito elétrico da Figura 2.10 (utilizado na Seção 2.3.1 para simulações *off-line*) na bancada HIL. Onde a modelagem do conversor VSC considera que a chave superior e inferior de cada ramo (S_1 e S_2 , S_3 e S_4 , S_5 e S_6 , respectivamente) operam de forma complementar, ou seja, no instante em que uma está fechada a outra está aberta. Assim, unicamente os três sinais de controle das chaves superiores são necessários para comandar a operação do conversor (S_1 , S_3 e S_5) e os sinais inferiores são interpretados como o negado do sinal superior (operação lógica NOT). Por outras palavras, não é considerado o tempo-morto comumente introduzido pelos *drivers* que estabelecem os pulsos de chaveamento em inversores reais [98].

Um quarto sinal S_{chav} é utilizado também para indicar os momentos nos quais todas as seis chaves estão em estado de alta impedância, ou seja, quando o chaveamento do conversor é interrompido. Esses sinais são estabelecidos pelo DSP mediante um controle em malha aberta que consiste (da mesma forma que na análise do Capítulo 2) em criar ondas moduladas PWM a partir de referência senoidais de 60 Hz e com amplitude de 0,85 pu na base da portadora triangular.

Este ensaio tem três objetivos: i) avaliar a precisão da bancada HIL em comparação aos resultados de simulação *off-line*, considerando a quantização da representação numérica por ponto fixo (FXP) do módulo FPGA e a assincronia entre os relógios da FPGA e do DSP; ii) comparar as características dos quatro métodos de representação de chaves utilizados e iii) avaliar o desempenho em regime permanente da abordagem para adequação dos sinais a serem exportados pelo SDTR.

3.5.2.1. Avaliação da precisão da bancada HIL

A modelagem do sistema elétrico com controle em malha aberta da Figura 2.10 foi realizada no SDTR/FPGA utilizando representação por unidade (pu) tomando bases de tensão e de corrente de 400 V e 500 A, respectivamente. Neste sentido, dado que o módulo FPGA utilizado tem a capacidade de realizar operações matriciais com números de até 32 bits em ponto fixo (FXP), as configurações escolhidas para as representações numéricas em ponto fixo da matriz e os vetores da operação matricial $\mathbf{x}[k] = \mathbf{H}^{-1} \cdot \mathbf{b}[k]$ são as seguintes:

Tabela 3.2– Configuração da representação numérica em ponto fixo.

Variável	Características
$\mathbf{b}[k]$	Número de bits: 32; Número de bits parte inteira: 3; faixa de representatividade: [-4, +4]; resolução 1,862e-9.
$\mathbf{x}[k]$	Número de bits: 32; Número de bits parte inteira: 3; faixa de representatividade: [-4, +4]; resolução 1,862e-9.
\mathbf{H}^{-1}	Número de bits: 32; Número de bits parte inteira: 9; faixa de representatividade: [-256, +256]; resolução 1,192e-7.

A avaliação numérica dos erros de exatidão dos resultados experimentais de simulação em tempo real para cada um dos métodos de representação de chaves é mostrada na Tabela 3.3. Nesta tabela, o indicador ε_x é calculado conforme (2.25), usando como *benchmark* os resultados de simulação *off-line* no software Simulink com modelagem das chaves como chave ideal (conforme definição da Seção 2.2.2.2). Observa-se que estes resultados são coerentes com os apresentados na Tabela 2.3 para o caso de comparação *off-line* do Capítulo 2, onde os métodos de representação G-ADC e ADC-I são os mais exatos. Entretanto, a comparação dessas duas tabelas indica que a bancada experimental introduz um erro de $\pm 1,15\%$ nas ondas de corrente, o que é consequência da operação assíncrona dos relógios do módulo FPGA e do DSP, bem como da quantização da representação em ponto fixo.

Tabela 3.3– Avaliação de exatidão das abordagens de representação de chaves.

Método (FXP)	Erro corrente CA (ε_x)
ADC tradicional	4,33%
G-ADC	1,21%
ADC-I (Proposta)	1,36%
G-ADC-SI (Proposta)	2,65%

Neste trabalho é considerado que as inexatidões dos métodos G-ADC e ADC-I estão dentro da margem de tolerância aceitável para a aplicação de validação de funcionamento de controladores embarcados.

3.5.2.2. Comparação do uso de recursos da FPGA para cada método de modelagem

Além da exatidão de representação, outras características importantes para um modelo utilizado para simulação em tempo real são seu tempo de execução e sua demanda computacional, atributos inversamente relacionados. Neste sentido, a Tabela 3.4 mostra os resultados da comparação das principais características de compilação do modelo elétrico da Figura 2.10 para execução no SDTR/FPGA desenvolvido, considerando todos métodos de modelagem chaveada utilizados neste trabalho. Destaca-se, neste sentido, que a metodologia de programação empregada (ver Seção 3.4) é tal que origina o mesmo tempo de execução para todos os métodos, resultante em 740 ns. Adicionalmente, em contraste com alguns simuladores comerciais, esta programação não está focada em diminuir o tempo de compilação, configuração que repercute em maior uso de recursos. O compilador utilizado é um servidor na nuvem de computação em paralelo da empresa National Instruments (*LabVIEW FPGA Compile Farm*).

A Tabela 3.4 mostra que o método G-ADC é aquele que apresenta maior tempo de compilação e que faz maior uso de recursos de *hardware*. Em especial no que se refere ao uso das unidades de DSP48s da FPGA, para o qual se observa um incremento de 58%¹² em comparação com os resultados para os métodos ADC e ADC-I. Por outras palavras, as técnicas ADC e ADC-I têm a capacidade de representar maior quantidade de componentes elétricos para o mesmo *hardware* do que a técnica G-ADC.

Tabela 3.4– Características de compilação do módulo FPGA PXIe-7965R.

Método	Tempo de compilação	Uso de recursos FPGA
ADC	27 min.	Slice Registers: 30,1%; Slice LUTs: 28,2% Block RAMs: 1,6%; DSP48s: 16,2%
G-ADC	35 min.	Slice Registers: 31,4%; Slice LUTs: 30,2% Block RAMs: 1,6%; DSP48s: 25,6%
ADC-I (Proposta)	29 min.	Slice Registers: 31,3%; Slice LUTs: 29,5% Block RAMs: 1,6%; DSP48s: 16,2%
G-ADC-SI (Proposta)	31 min.	Slice Registers: 30,0%; Slice LUTs: 28,7% Block RAMs: 1,6%; DSP48s: 25,6%

¹² Em termos da capacidade máxima da FPGA utilizada o incremento foi de 9,4%.

Considerando os resultados das duas tabelas anteriores, nesta tese escolheu-se empregar a técnica ADC-I (proposta) nos desenvolvimentos apresentados a partir deste ponto, uma vez que esta técnica oferece um nível de exatidão similar ao método G-ADC, porém com menor uso de recursos.

3.5.2.3. Desempenho da abordagem para adequação de sinais analógicos

A Figura 3.9 mostra o resultado experimental (medição com osciloscópio) para a onda representativa da corrente na fase a do lado CA do conversor ($i_a^{D/A}$) para o caso no qual nenhuma adequação é realizada na transferência de sinais digitais entre o módulo FPGA e o módulo D/A.

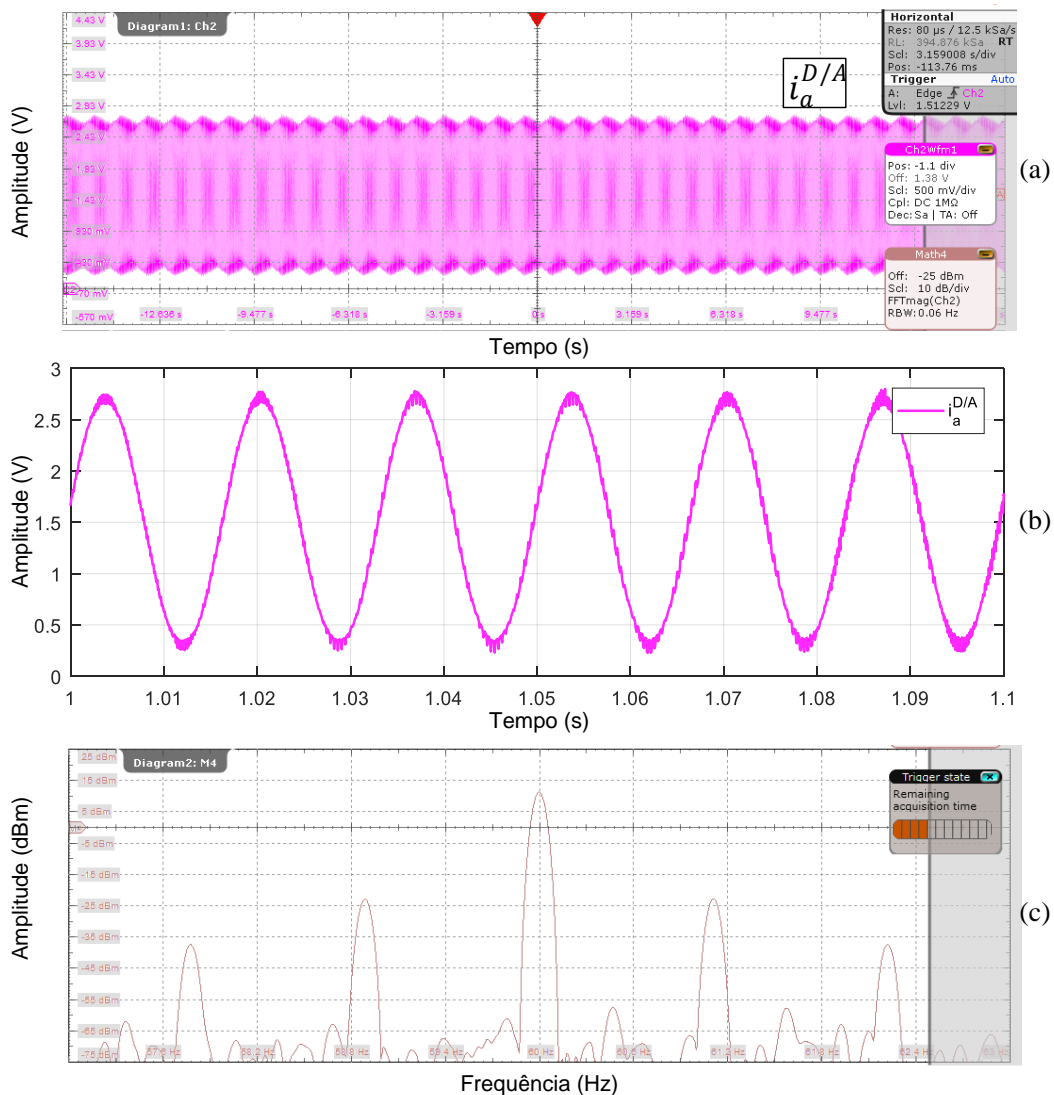


Figura 3.9 – Medição experimental de uma onda analógica de saída do SDTR/FPGA ($i_a^{D/A}$), sem o uso de adequação de sinais: (a) $i_a^{D/A}$; (b) zoom horizontal na onda $i_a^{D/A}$; (c) harmônicos próximos da frequência fundamental.

Conforme discussão da Seção 3.3, as características do *hardware* utilizado como SDTR originam a aparição de inter-harmônicos na onda resultante nos terminais do simulador, como pode ser verificado pelo diagrama FFT (*Fast Fourier Transformation*) da Figura 3.9(c). Este comportamento consiste na manifestação de erros de *Aliasing*, portanto, não é característico do sistema elétrico modelado e inabilitaria em um primeiro momento o uso do *hardware* empregado nesta tese como SDTR.

Por outro lado, o resultado experimental para o caso do uso da abordagem de adequação dos sinais analógicos através do sincronismo entre o DSP e o SDTR, proposta na Seção 3.3.1, é apresentado na Figura 3.10.

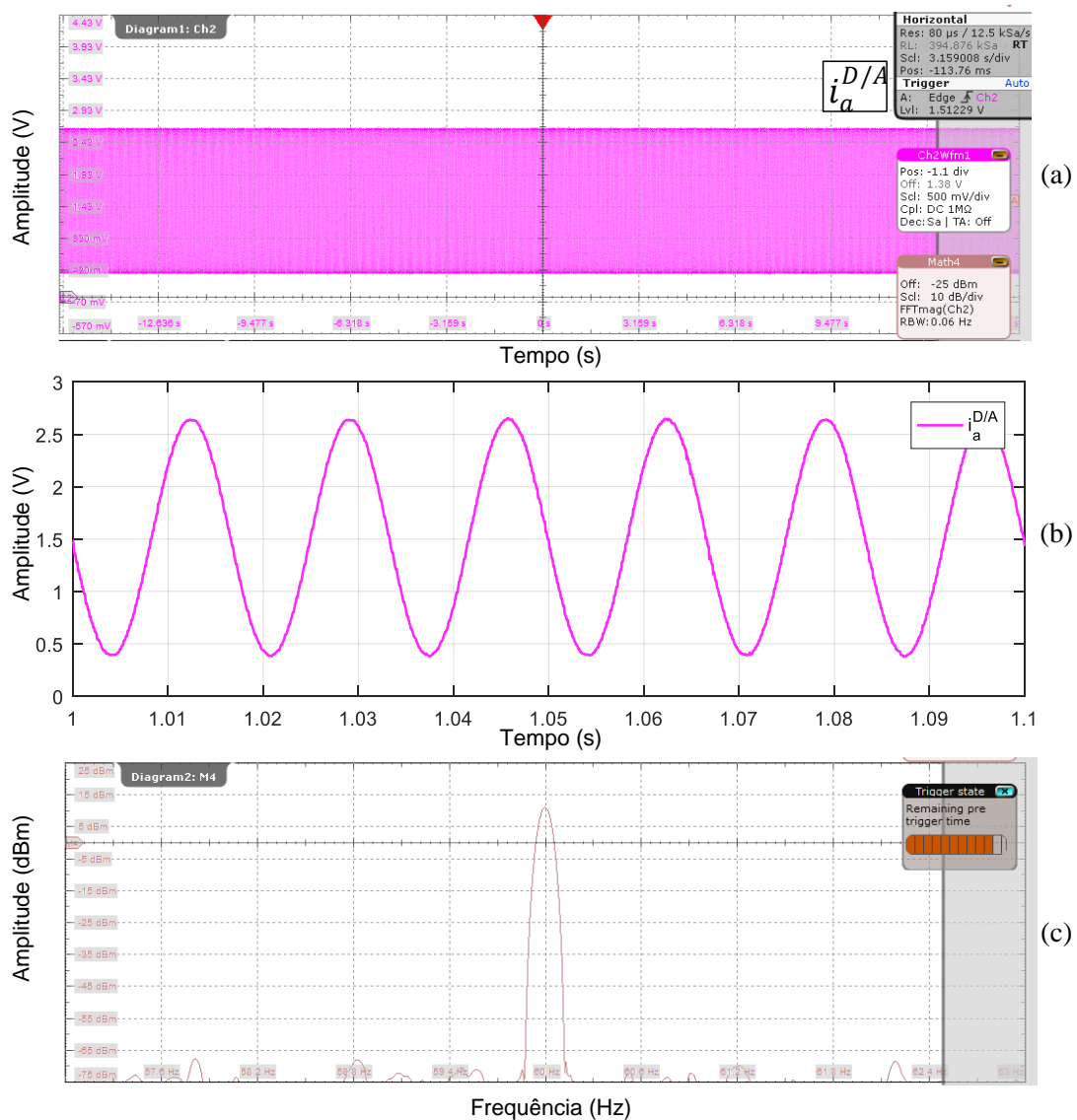


Figura 3.10 – Medição experimental de uma onda analógica de saída do SDTR/FPGA ($i_a^{D/A}$), com o uso da técnica de adequação por sincronismo: (a) $i_a^{D/A}$; (b) zoom horizontal na onda $i_a^{D/A}$; (c) harmônicos próximos da frequência fundamental.

Desta vez, observa-se um adequado comportamento em regime permanente da onda exportada que, conforme mostra a Figura 3.10(c), não contém valores significativos de inter-harmônicos em torno da frequência fundamental de 60 Hz. De fato, a Figura 3.10(b) mostra uma onda senoidal adequadamente filtrada, o que valida o uso desta estratégia. O efeito do atraso causado por esta abordagem de sincronismo, no desempenho dinâmico do SDTR proposto é analisado na Seção 3.5.4.

3.5.3. Simulação de conversor VSC operando em malha fechada

3.5.3.1. Sistema elétrico

No ensaio de operação do VSC com controle em malha fechada é utilizado o sistema elétrico da Figura 3.11 no qual um conversor trifásico de dois níveis com filtro indutivo nos terminais CA é conectado à rede elétrica.

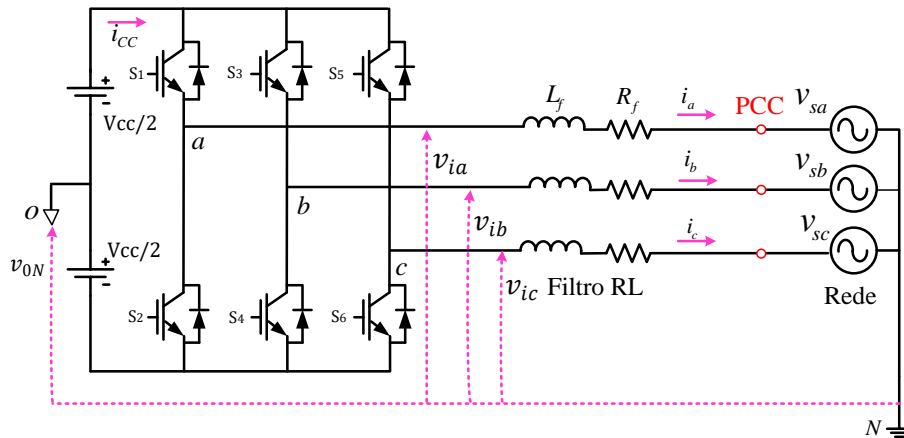


Figura 3.11 – Sistema elétrico utilizado na validação com ensaio HIL.

Por simplicidade é considerado inicialmente que o ponto de conexão com a rede é ideal, representado por um conjunto de três fontes senoidais balanceadas (sequência negativa e sequência zero nulas) com tensão eficaz fase-fase de 440 V e frequência de 60 Hz. Neste caso, a reatância indutiva equivalente do filtro, na frequência de 60 Hz, é especificada em 15% da impedância base, de forma a reduzir o conteúdo harmônico das correntes injetadas pelo conversor. O fator de qualidade deste filtro indutivo é de 25. Os outros parâmetros do sistema modelado são apresentados na Tabela 3.5.

Tabela 3.5– Parâmetros do sistema elétrico da Figura 3.11.

Grandeza	Símbolo	Valor	Grandeza	Símbolo	Valor
Tensão nominal elo CC	V_{CC}	800 V	Indutância filtro	L_f	308,12 μ H
Tensão fase-fase da rede	V_S	440 Vrms	Resistência filtro	R_f	4,65 m Ω
Potência nominal	S_n	250 kVA	Frequência de chaveamento	f_{ch}	10 kHz

Da mesma forma que como realizado para o caso da simulação do VSC operando em malha aberta, os sinais de chaveamento para as chaves inferiores (S_2, S_4 e S_6) são interpretados como o negado do sinal superior de cada ramo (S_1, S_3 e S_5 , respectivamente). Adicionalmente, para este caso de estudo empregam-se a mesma configuração da representação por ponto fixo das variáveis e o mesmo valor numérico para G_s dos utilizados na Seção 3.5.2.

3.5.3.2. Estrutura de controle

O controlador para o conversor da Figura 3.11 foi projetado para regular as correntes sintetizadas pelo VSC, de tal forma que as potências injetadas na rede sigam as potências ativa e reativa de referência (P_{ref} e Q_{ref}). Em termos gerais, esta estrutura básica de controle é um laço primário de um controlador para DER [99]. O controle implementado neste trabalho utiliza o referencial síncrono dq que se caracteriza por facilitar o atingimento de erro nulo em estado permanente, uma vez que idealmente as variáveis de controle são quantidades contínuas [81]. O uso de referencial síncrono dq é considerado adequado em aplicações que lidem com redes elétricas sem relevantes distorções ou desequilíbrios, em outros casos técnicas baseadas em controladores Proporcionais Ressonantes (PR) no referencial $\alpha\beta$ podem ser aplicadas [100].

Um diagrama esquemático da estrutura de controle implementada no DSP é mostrado na Figura 3.12. Nele, os sinais de tensão no ponto de conexão comum (PCC) (v_{sa}, v_{sb} e v_{sc}) são utilizados como entrada para um bloco de rastreamento de fase (PLL- *Phase-Locked Loop*) que determina o ângulo de sincronismo com a rede de distribuição θ_{PLL} . Esse ângulo é utilizado para transformar as grandezas do sistema para o referencial síncrono dq . Após os componentes v_{sd} e v_{sq} serem calculados, um bloco de controle é utilizado para estabelecer as correntes de referência necessárias para atingir os valores desejados de potência. O bloco encarregado de sintetizar as correntes de referência na saída do inversor é o denominado “controlador corrente” que realiza esta função mediante a determinação dos sinais de referência para o chaveamento PWM (*Pulse Width Modulation*). O chaveamento PWM tem a função final de estabelecer os pulsos de disparo das chaves do conversor VSC (sinais S_1, S_3 e S_5).

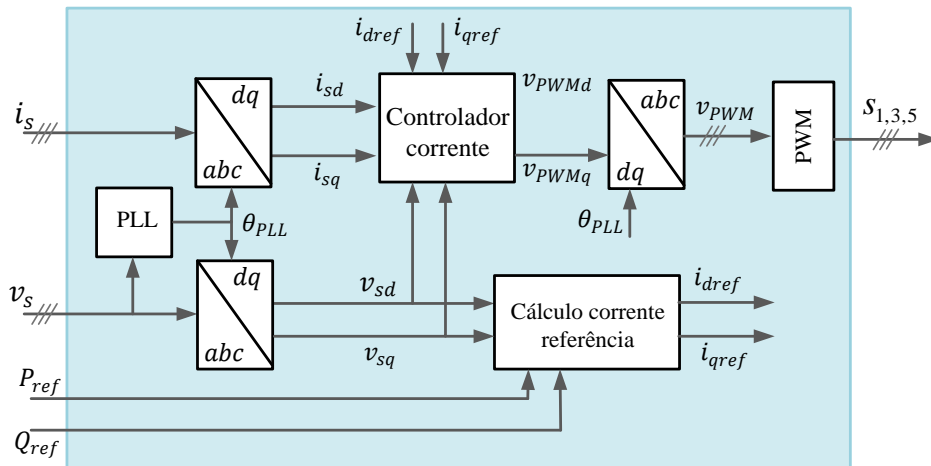


Figura 3.12 – Diagrama de blocos da estrutura de controle em malha fechada.

Uma descrição mais detalhada dos blocos de controle referidos na Figura 3.12 é realizada no Apêndice C. Onde é apresentado o desenvolvimento matemático que determina a escolha dos ganhos dos controladores Proporcional Integral (PI) que fazem parte desta estrutura de controle, apresentados na Tabela 3.6.

Tabela 3.6– Configuração dos ganhos dos controladores PI.

Controlador PI	K_P	K_I
PLL -Rastreamento de fase	70 (rad/s)/rad	2500 (rad/s)/(rad·s)
Controle de corrente	0,36 pu/pu	5,4 pu/(pu·s)

Programação do DSP

O esquema de controle descrito anteriormente foi embarcado no microcontrolador DSP da bancada HIL da Figura 3.8. A metodologia utilizada neste trabalho para realizar este procedimento emprega os *software* Simulink, onde é realizada a programação gráfica da estrutura de controle, e Code Composer Studio¹³ (CCS), onde é realizada a depuração das funções embarcadas. Foram necessárias as *toolboxes* “Simulink Coder” e “Embedded Coder” do *software* MATLAB para criar um arquivo executável (carregável no DSP,) a partir do modelo de Simulink [101]. Os periféricos do DSP utilizados para desenvolver a estrutura de controle do diagrama de blocos da Figura 3.12 são:

- (1) Interrupção de *hardware* gerada pelo periférico ePWM (*Enhanced Pulse Width Modulator*), utilizada para controlar a taxa de execução do algoritmo de controle, definida em 10 kHz.

¹³ Disponibilizado pelo fabricante do DSP.

- (2) Conversores A/D, utilizados para aquisição dos sinais analógicos de entrada, representativos das grandezas do sistema simulado. Cujas principais especificações são: faixa de operação (0 a 3 V), resolução (12 bits) e tipo de variável de saída (uint16) [102].
- (3) Periférico ePWM, que cria os sinais tipo TTL (*Transistor-Transistor Logic*) para comandar a operação do inversor modelado no SDTR.
- (4) Blocos de memória, para disponibilizar variáveis de controle no *software* de monitoramento CCS.

3.5.3.3. Resultados obtidos

A Figura 3.13 mostra os resultados do primeiro ensaio realizado, no qual foi aplicado um degrau na referência de potência ativa do controlador no instante de simulação $t = 0,312$ s, de forma que o valor de P_{ref} passa de 0,5 pu para 0,85 pu. A referência de potência reativa (Q_{ref}), por outra parte, permanece nula durante todo o período do ensaio. A Figura 3.13(a) mostra os resultados experimentais (formas de onda medidas pelo osciloscópio nos terminais analógicos do SDTR/FPGA proposto), enquanto que a Figura 3.13(b) corresponde aos resultados obtidos com o simulador *off-line* Simulink, considerando o mesmo evento e a mesma estrutura de controle (desta vez executada no próprio simulador *off-line*), para comparação.

Destaca-se que as formas de onda apresentadas na Figura 3.13(a) são tensões representativas das variáveis v_{sa} , i_a , i_b e i_c , adaptadas à faixa de operação do conversor A/D do DSP utilizado neste trabalho (0 a 3 V). Portanto, a amplitude das variáveis simuladas pode ser obtida subtraindo o nível CC de 1,5 V e multiplicando por um fator de escala (560/1,5 A/V no caso das ondas de corrente, e 400/1,5 V/V no caso da onda de tensão) nas formas de onda apresentadas ($v_{sa}^{D/A}$, $i_a^{D/A}$, $i_b^{D/A}$ e $i_c^{D/A}$). Por outro lado, na Figura 3.13(b) (resultados *off-line*) são utilizadas duas escalas verticais: tensão (direita) e corrente (esquerda), cujas linhas de grade foram escolhidas tal forma que permitissem uma comparação visual direta com os resultados experimentais da bancada HIL.

As formas de onda da Figura 3.13 mostram um adequado comportamento do controlador, que modifica em menos de meio ciclo da onda de tensão o ponto de operação do conversor VSC, mantendo um fator de potência próximo ao unitário antes e depois da mudança de referência.

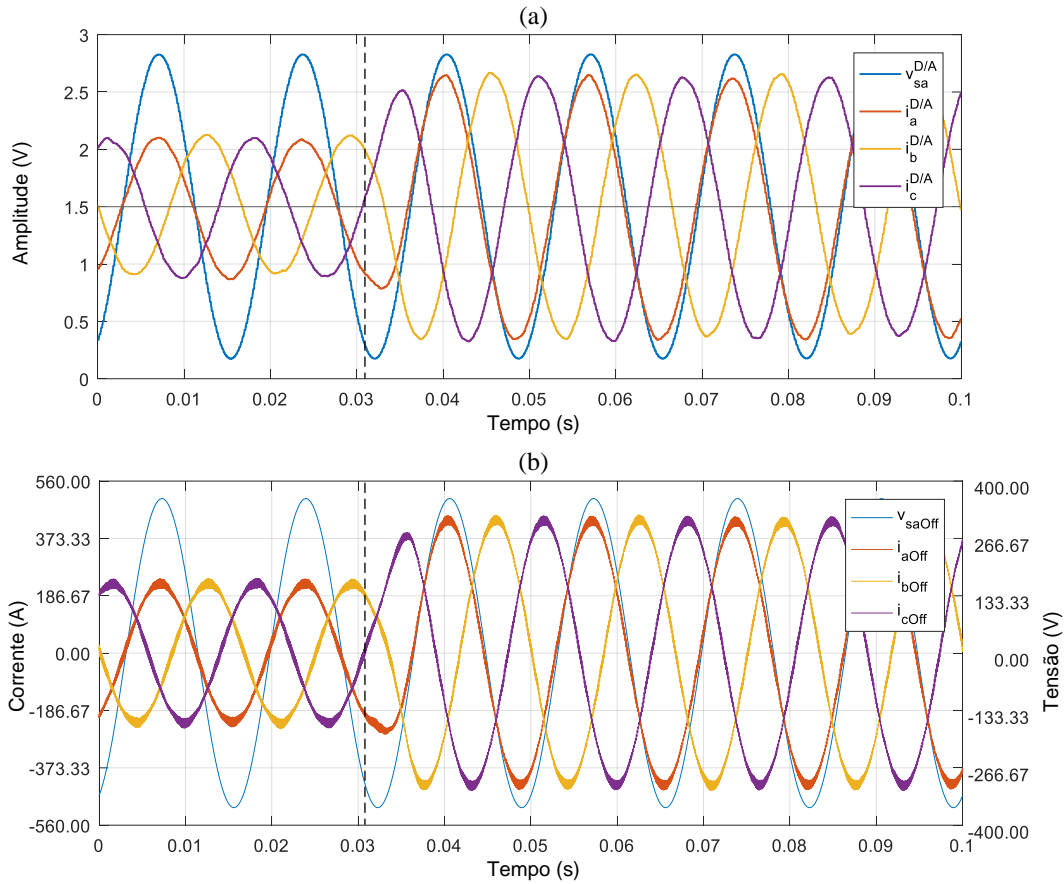


Figura 3.13 – Comportamento do VSC diante de uma mudança de referência de potência ativa: (a) resultados experimentais – chaves modeladas com a técnica ADC-I; (b) resultados de simulação *off-line*.

Para comparar numericamente as correntes de inversor dos resultados experimentais e dos resultados de simulação *off-line* é utilizada a expressão (3.1).

$$\varepsilon_{exp} = \left(\frac{rms(i_{aEXP} - i_{aOff}^A)}{rms(i_{aOff}^A)} + \frac{rms(i_{bEXP} - i_{bOff}^A)}{rms(i_{bOff}^A)} + \frac{rms(i_{cEXP} - i_{cOff}^A)}{rms(i_{cOff}^A)} \right) / 3 \quad (3.1)$$

Onde:

$$\begin{aligned} i_{aEXP} &= (i_a^{D/A} - 1,5) \cdot 560/1,5; \\ i_{bEXP} &= (i_b^{D/A} - 1,5) \cdot 560/1,5; \\ i_{cEXP} &= (i_c^{D/A} - 1,5) \cdot 560/1,5; \end{aligned} \quad (3.2)$$

e i_{aOff}^A , i_{bOff}^A , i_{cOff}^A são amostras de i_{aoff} , i_{boff} e i_{cOff} , respectivamente. Essas amostras são realizadas com frequência de 10 kHz e em sincronismo com o ponto meio da onda de chaveamento durante um período dos pulsos PWM. Esta amostragem se torna necessária uma vez que nas ondas de corrente experimentais não existe o *ripple* de chaveamento, dado o uso do conversor D/A com subamostragem e a técnica de adequação de sinais descrita na Seção 3.3.1.

A avaliação de (3.1) nas formas de onda da Figura 3.13 dá como resultado $\varepsilon_{exp} = 3,8\%$. Este índice de erro é maior do que o apresentado na Tabela 3.3 para o caso de validação em malha aberta como consequência dos erros de quantização dos canais de saída analógico do SDTR/FPGA, bem como pela operação independente em malha fechada dos dois sistemas, que origina que diferentes sinais de controle sejam aplicados em cada um dos modelos digitais (modelo em tempo real e modelo de simulação *off-line* no Simulink). Entretanto, nesta tese se considera admissível para fins de validação da operação do controlador embarcado.

Para observar as ondas chaveadas experimentais deste ensaio se faz necessário o registro FIFO DMA, descrito na Seção 3.4.4. Nele, as variáveis de saída são armazenadas com resolução temporal de $1 \mu s$. Isto permite analisar o detalhamento do *ripple* das ondas de corrente, conforme realizado pela visão ampliada da Figura 3.14.

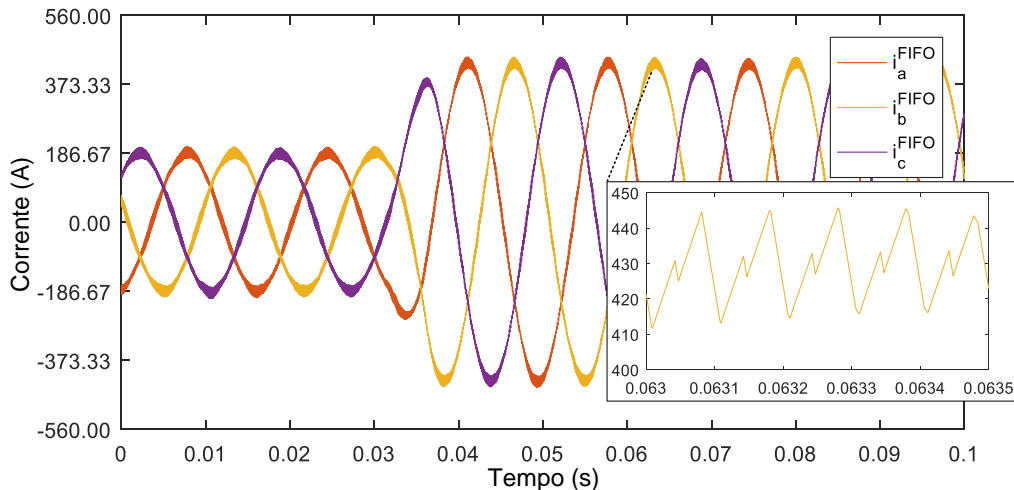


Figura 3.14 – Correntes do inversor armazenadas no registro FIFO DMA – ensaio de mudança de referência ativa. Chaves modeladas com a estratégia ADC-I.

O comportamento detalhado do controlador de corrente é observado na Figura 3.15 que mostra: (a) as correntes injetadas pelo conversor VSC utilizando o eixo referencial síncrono dq , (b) o erro de corrente, calculado como a diferença entre as correntes de referência e as correntes atuais, e (c) as saídas do bloco de controle de corrente (v_{PWMd} e v_{PWMq} no diagrama esquemático da Figura A.10). Observa-se que o controlador de corrente consegue atingir erro nulo em regime permanente e que responde de forma desacoplada (entre os componentes d e q) às variações nas potências de referência, conforme projetado no ajuste dos ganhos K_P e K_I .

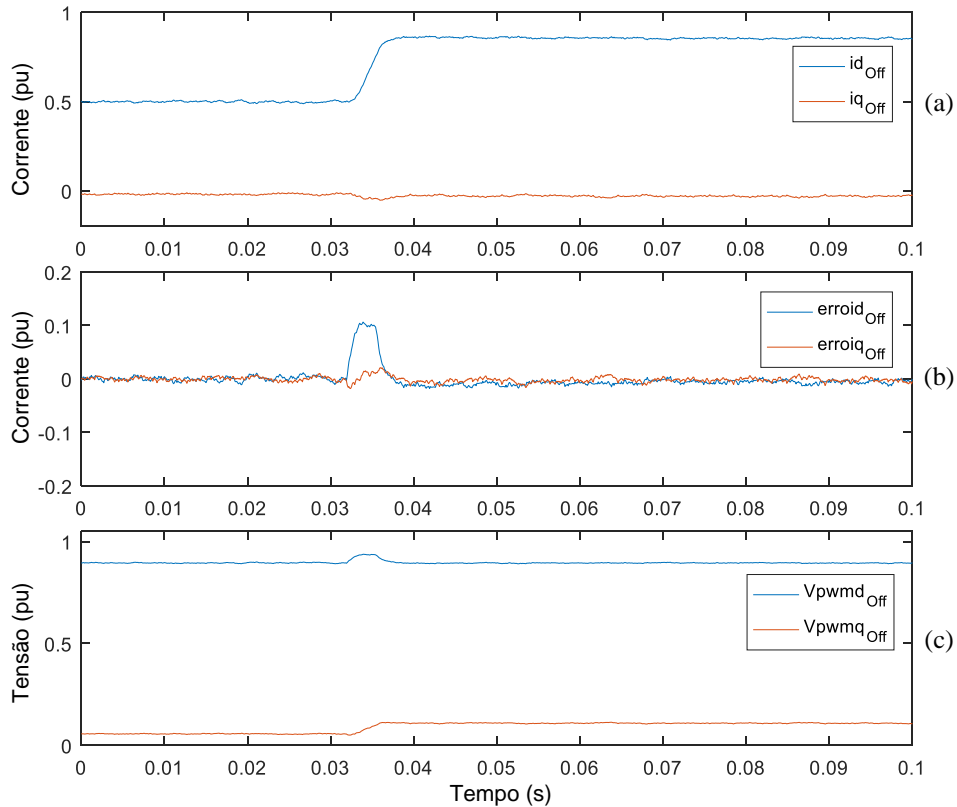


Figura 3.15 – Desempenho do controle interno de corrente - ensaio de mudança de referência ativa: (a) correntes i_d e i_q ; (b) erro de corrente; (c) saída do controlador de corrente.

A Figura 3.16 mostra o espectro harmônico de $i_a^{D/A}$, calculado pelo osciloscópio, nas redondezas da frequência fundamental. Observa-se que mesmo na operação em malha fechada, os inter-harmônicos resultantes mantêm amplitudes menores do que -55 dBm, validando, para condição de operação de regime permanente, a abordagem de sincronismo entre DSP e SDTR para adequação dos sinais analógicos exportados.

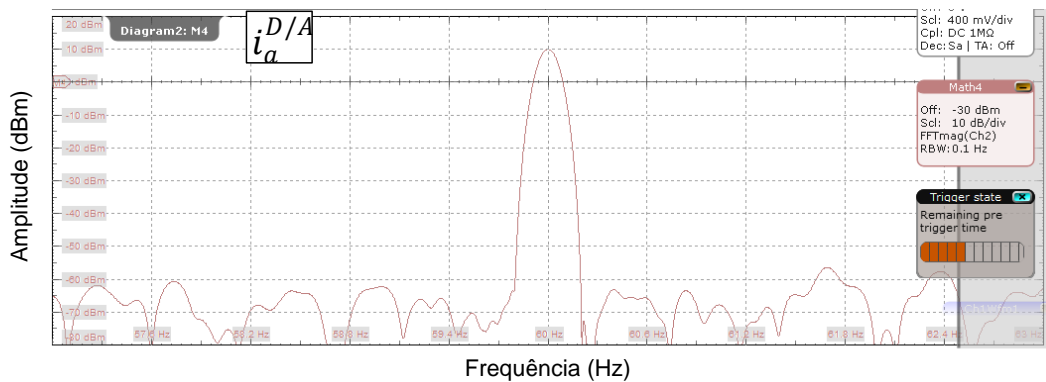


Figura 3.16 – Medição com osciloscópio dos sub-harmônicos de corrente próximos da frequência fundamental.

Utilizando novamente os dados armazenados no registro FIFO DMA é possível extrair as formas de onda das tensões produzidas num braço do conversor VSC. Neste sentido, a Figura 3.17 mostra a tensão v_{ia} durante uma janela de 1 ms. Assim como

ocorreu nos resultados apresentados no Capítulo 2, observa-se que o transitório numérico originado pelo método ADC-I após uma mudança de estado tem uma amplitude máxima de aproximadamente 3 V, o que é relativamente pequeno considerando a tensão do elo CC (800 V).

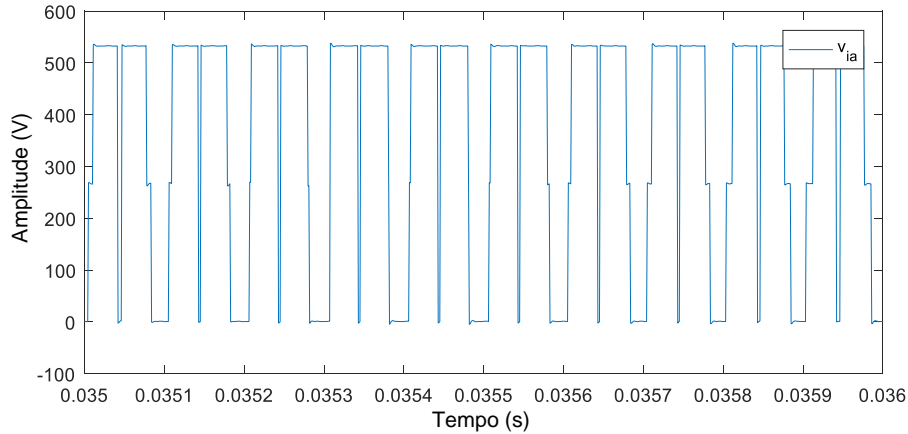


Figura 3.17 – Tensão v_{ia} durante o ensaio de mudança de referência ativa. Visão detalhada dos valores armazenados no registro FIFO DMA. Chaves modeladas pela estratégia ADC-I.

Finalmente, a Figura 3.18 apresenta as medições com osciloscópio das variáveis representativas do sistema ($v_{sa}^{D/A}$, $i_a^{D/A}$, $i_b^{D/A}$, $i_c^{D/A}$) quando o inversor é controlado para injetar potência ativa e reativa, neste caso $P_{ref} = 0,5$ pu e $Q_{ref} = 0,5$ pu, o que representa um fator de potência de 0,707 ($\varphi = 45^\circ$). Um ponto de operação com esta característica pode ser utilizado por DERs para contribuir no controle de tensão no PCC [103]. Observa-se novamente um comportamento satisfatório do controlador, uma vez que o ângulo de deslocamento medido foi de $44,7^\circ$.

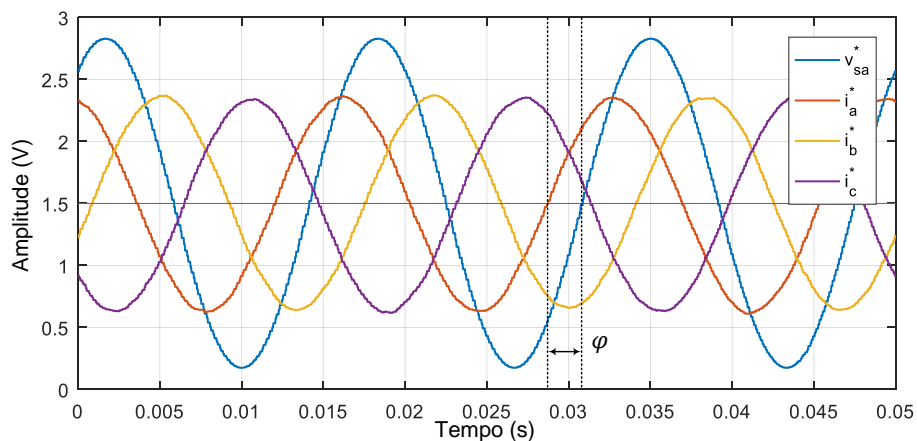


Figura 3.18 – Formas de onda de saída no SDTR. $P_{ref} = 0,5$ pu e $Q_{ref} = 0,5$ pu. Chaves modeladas com a estratégia ADC-I.

3.5.4. Caracterização dos limites de desempenho do SDTR/FPGA

Os resultados anteriores validaram o uso do SDTR/FPGA na simulação em tempo real de um inversor VSC trifásico de dois níveis conectado a um equivalente de rede com tensões trifásicas senoidais na frequência fundamental, o que, conforme apresentado, pode ser aplicado no desenvolvimento de uma bancada HIL focada em avaliar controladores embarcados. Entretanto, esta aplicação é limitada a certos parâmetros de simulação, conforme descrito nesta seção.

Número de chaves representáveis

Conforme mostrado na Tabela 3.1, a FPGA disponível para este desenvolvimento possui somente oito canais de entrada digital, dos quais cinco foram utilizados para a simulação do conversor VSC trifásico (3 para os pulsos PWM de chaveamento, 1 para indicar o estado Off das 6 chaves e 1 para efetuar a abordagem de adequação de sinais por sincronismo). Assim, o SDTR/FPGA está limitado a topologias com até 6 pares complementares de chaves, de forma que topologias de conversores mais complexas requereriam o uso de outro *hardware* de computação.

Frequência de chaveamento

O algoritmo de simulação programado no SDTR/FPGA não utiliza nenhum algoritmo de sincronização para lidar com os erros originados pelos eventos de chaveamento entre passos de amostragem (ITS), ver discussão da Seção 1.3. Assim, a exatidão deste algoritmo está limitada à existência de uma ampla relação entre o período de chaveamento e o passo de simulação. Em [28] é concluído que uma relação superior a 100 pu garante erros desprezíveis relacionados a ITS. Assim, uma vez que o tempo de execução da programação efetuada no módulo FPGA é de 740 ns, considera-se que o SDTR/FPGA consegue representar adequadamente conversores com frequências de chaveamento entre 0 e 15 kHz. Destaca-se que ajustes na programação do módulo FPGA podem reduzir seu tempo de execução, ampliando esta janela de operação.

Desempenho do SDTR/FPGA na presença de distorções harmônicas na rede

Na Seção 3.3 foi mencionado que a contrapartida da abordagem de adequação de sinais utilizada no SDTR/FPGA é a introdução de um atraso de um período de chaveamento para a exportação dos sinais analógicos. Neste sentido, a Figura 3.19

ilustra conceitualmente a interferência deste atraso na operação de um conversor com controle em malha fechada que utilize esses sinais analógicos como realimentação.

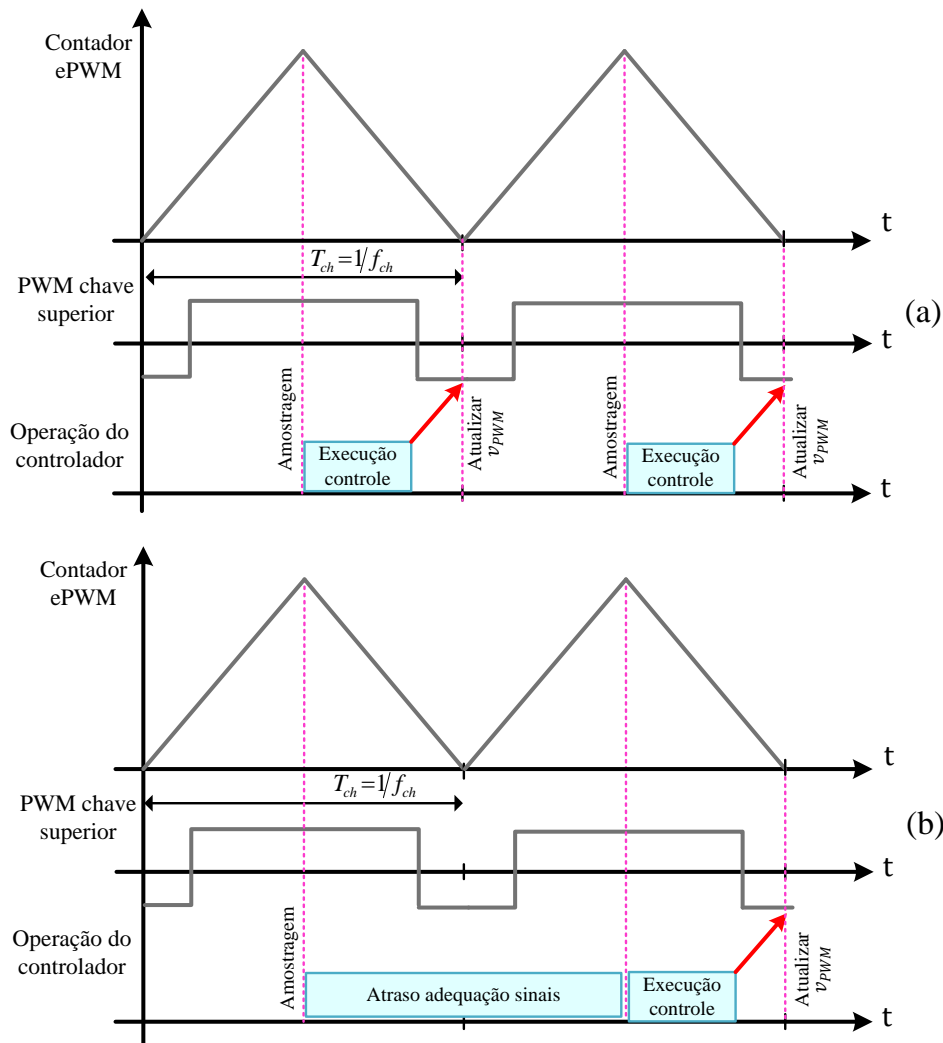


Figura 3.19 –Diagrama temporal da operação de um controle em malha fechada sincronizado com os pulsos PWM: (a) condição normal; (b) condição obtida com o SDTR/FPGA.

Na Figura 3.19(a) é mostrado que para o caso da condição normal com PWM de atualização única, o instante de amostragem está sincronizado com o valor de pico da portadora triangular do PWM (Contador ePWM) e o instante de atualização da tensão de referência para o comparador PWM (v_{PWM}) está sincronizado com o valor mínimo da portadora triangular¹⁴, o que origina o atraso inerente de um ciclo dos controladores digitais [104]. Por outro lado, a Figura 3.19(b) mostra que no caso do SDTR/FPGA o instante de atualização de v_{PWM} está atrasado mais um ciclo do que no caso normal. Isto afeta o desempenho do controlador interno de corrente, uma vez que o estado das

¹⁴ Isto é possível sempre que o tempo de execução da rotina de controle seja menor do que $1/(2 \cdot f_{ch})$, condição satisfeita pela estrutura de controle utilizada nesta tese.

variáveis do sistema no instante da aplicação do sinal de controle difere do estado utilizado para efetuar os cálculos do controlador em maior medida.

Neste sentido, entende-se que a influência deste atraso na exatidão da simulação depende do conteúdo harmônico do equivalente de rede elétrica modelado, uma vez que quanto maior for a frequência da distorção harmônica, mais relevante será a variação da onda de tensão durante o período de atraso. Simulações adicionais foram realizadas na bancada HIL com o intuito de verificar experimentalmente esta característica bem como identificar o limite de aplicabilidade do SDTR/FPGA em relação a esse aspecto. Nessas simulações foi considerado o sistema elétrico da Figura 3.11, porém com a introdução de distorções harmônicas individuais nas fontes de tensão equivalentes.

O controle utilizado nestas simulações também está baseado no esquema da Figura 3.12, porém neste caso os parâmetros de entrada para o controlador são as correntes de referência nos eixos direto e em quadratura (i_{dref} e i_{qref}). Por outras palavras, o controle utilizado nesta análise dispensa do bloco “cálculo corrente de referência”, cuja expressão matemática é descrita pela equação (A.10).

Nos ensaios realizados foram introduzidas distorções harmônicas individuais de tensão [105] nas fontes equivalentes do circuito modelado. Essas distorções têm amplitude de 7%; frequências de 180, 300 e 420 Hz e defasagem angular em relação à componente fundamental de 0, $\pi/2$ e π rad. A Tabela 3.7 mostra os resultados da avaliação de ε_{exp} , conforme definição de (3.1), para os ensaios realizados. Observa-se que o comportamento do SDTR/FPGA difere consideravelmente dos resultados de simulação *off-line* a partir do quinto harmônico.

Tabela 3.7– Resultados de avaliação do indicador ε_{exp} para o SDTR/FPGA diante distorções harmônicas na rede.

Fase Harmônico	0 rad	$\pi/2$ rad	π rad
3	3,47%	3,23%	3,41%
5	5,91%	6,05%	6,30%
7	9,49%	9,76%	9,40%

A Figura 3.20 mostra a comparação gráfica entre os resultados experimentais e os resultados de simulação *off-line* para um dos casos apresentados na tabela anterior. Observa-se que no caso experimental o controlador tem maior dificuldade para estabelecer as referências de corrente, de forma que acaba sintetizando umas formas de

onda de corrente consideravelmente distorcidas em relação com os resultados de simulação *off-line*, o que justifica o aumento no indicador de erro ε_{exp} .

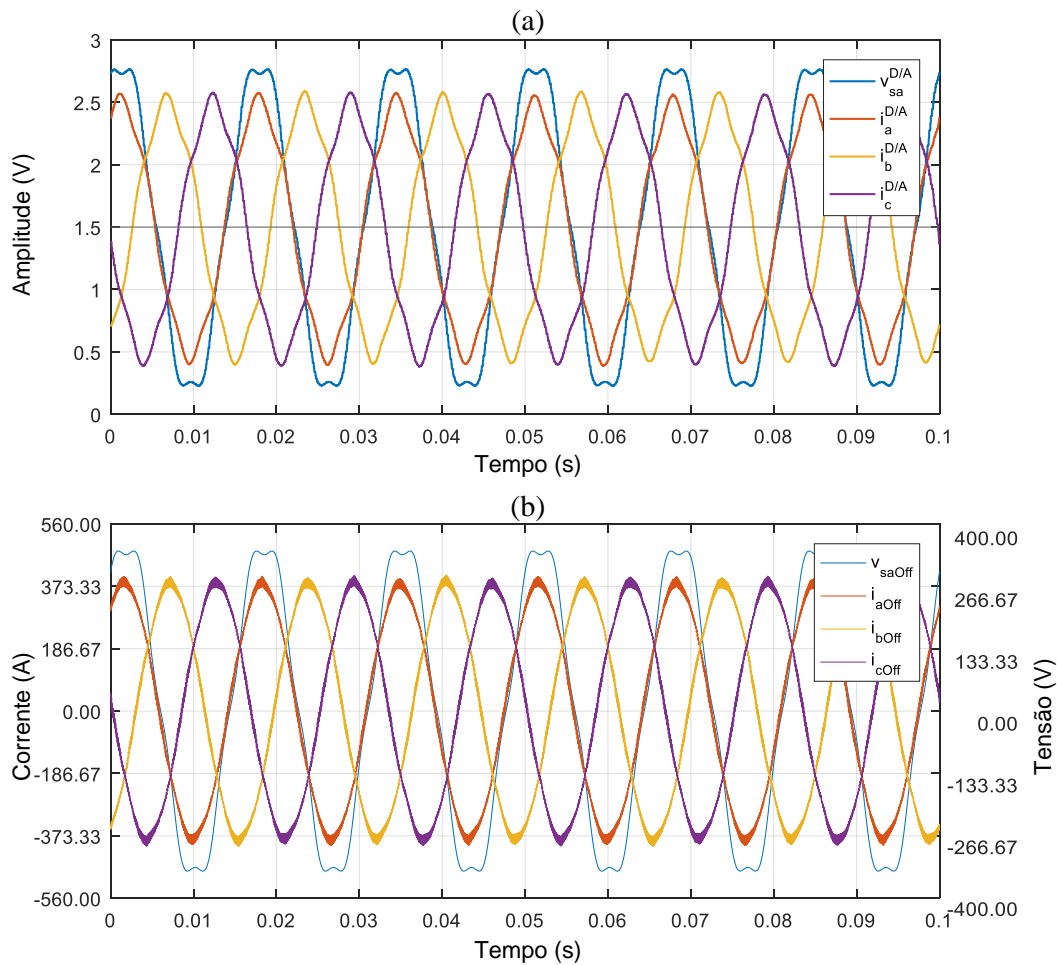


Figura 3.20 – Comportamento do VSC quando as fontes equivalentes têm distorção harmônica de 7% na frequência de 300 Hz, com defasagem de π rad: (a) resultados experimentais – chaves modeladas com a técnica ADC-I; (b) resultados de simulação *off-line*.

Assim, conclui-se desta análise que o SDTR/FPGA não pode ser utilizado, sem realizar modificações na estrutura de controle, em casos de estudo nos quais o equivalente de rede tenha distorções harmônicas relevantes em frequências superiores a 180 Hz. Entretanto, destaca-se que uma vez que a origem desta limitação é conhecida, técnicas especiais de controle focadas em abordar o problema do atraso adicional, poderiam ser utilizadas para superar esta restrição. Um exemplo deste tipo de técnicas é apresentado em [106], onde um observador baseado no filtro de Kalman é utilizado para compensar o atraso de dois ciclos de um controlador digital de um conversor VSC.

Capítulo 4

Desenvolvimento do SDTR integrado

Este capítulo apresenta o desenvolvimento do SDTR integrado, plataforma que conjuga, mediante uma co-simulação multitaxa, o SDTR/FPGA, proposto no capítulo anterior, e um SDTR comercial baseado em processadores Intel. Esta combinação de recursos permitiu constituir um equipamento com maior versatilidade, largura de banda e capacidade de simulação das que as que seriam conseguidas com o uso independente de cada simulador. Exemplos de aplicação desta ferramenta para estudos focados na integração de DER são apresentados. A originalidade deste desenvolvimento está na aplicação do método de interface de Transformador Ideal e da estratégia de modelagem de chaves ADC-I para integrar duas plataformas de *hardware* de diferentes fabricantes na referida co-simulação multitaxa.

4.1. Fundamentação teórica

4.1.1. Simulação em tempo real de sistemas elétricos com processadores tipo CPUs

A simulação digital de um sistema elétrico é essencialmente a solução numérica passo a passo das equações diferenciais que determinam seu comportamento. Assim, em algoritmos de solução com passo de simulação (Δt) fixo, Δt é um parâmetro que define a largura de banda dos transitórios que podem ser representados satisfatoriamente. Quanto menor for Δt , fenômenos de maior frequência poderão ser reproduzidos [5], [40], [70]. Em estudos de simulação *off-line* não existe uma limitação intrínseca para o mínimo valor de Δt , uma vez que o tempo de execução (t_e) de cada passo de simulação (que é função da complexidade dos modelos dos componentes) pode, em princípio, ser ordens de magnitude maior do que Δt . Na simulação em tempo real, em contraste, deve existir sincronismo entre Δt e t_e (ver Seção 1.1). Desta forma, a “potência de computação” requerida para solucionar um modelo digital em tempo real, definida como a multiplicação da complexidade (ou tamanho do sistema) pelo número de passos

de simulação por segundo [7], deve ser menor do que a capacidade do SDTR utilizado [4], [49].

A Figura 4.1, adaptada de [7], ilustra as relações típicas entre a complexidade do modelo digital (número de nós/barras) e o passo de simulação para diferentes aplicações em tempo real. Destaca-se que os eventos EMT representados nesta figura são os de frente de onda lenta [5], uma vez que os transitórios mais rápidos, como os relacionados com eventos de chaveamento em subestações isoladas a gás ou sobretensões originadas por descargas atmosféricas estão geralmente fora do foco das simulações em tempo real. Assim, a figura mostra que a simulação EMT de sistemas de potência pode utilizar um Δt comparativamente maior do que o requerido para simular modelos chaveados de conversores, o que para uma capacidade de computação constante permite representar um sistema relativamente maior. Neste sentido, uma prática comum é a de selecionar um passo de simulação em torno de $50 \mu s$ para tal aplicação, o que segundo várias referências permite representar adequadamente transitórios com frequências de até 2 kHz [6], [33], [49]. Ou seja, uma relação de 10 amostras por ciclo do componente harmônico de maior frequência costuma se considerar suficiente¹⁵.

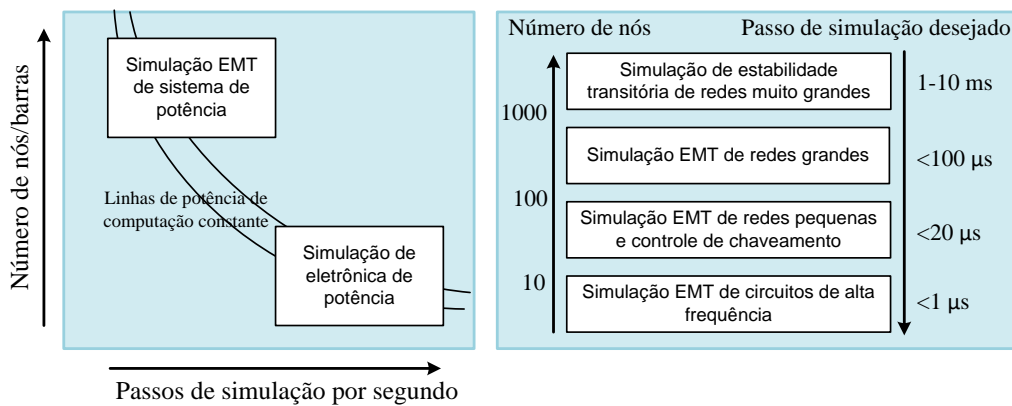


Figura 4.1 –Ilustração de requisitos de computação e de passo de simulação para diferentes tipos de aplicações. Adaptada de [7]. EMT: Transitório Eletromagnético.

Em contraste com o discutido na Seção 1.4 para a simulação em tempo real de modelos chaveados de conversores, um passo de execução da ordem de $50 \mu s$ não é limitante para processadores tipo CPU. De fato, os SDTR comerciais focados nessas aplicações costumam utilizar *racks* de processadores CPU *multicore* como *hardware* de

¹⁵ Em contraste com o teorema de Nyquist, no qual é exigida uma relação de pelo menos 2 amostras por ciclo do componente harmônico de maior frequência.

cálculo [6]–[8], [10]. Esses dispositivos são preferidos, sobre módulos FPGA, por sua maior flexibilidade, facilidade de programação e escalabilidade.

Um recurso tipicamente utilizado nos SDTR baseados em CPU para permitir a computação em paralelo (e, assim, representar sistemas elétricos complexos) é o de dividir o sistema elétrico em subsistemas, aproveitando os atrasos naturais de propagação das ondas eletromagnéticas nas linhas de transmissão. O modelo de linha de Bergeron, descrito no Apêndice D, é uma das principais abordagens utilizadas para tal propósito [40].

Neste contexto, entretanto, é observado o interesse dos fabricantes de SDTR comerciais na proposição e uso de estratégias focadas em aumentar a potência de computação de seus equipamentos sem recorrer à separação do sistema elétrico. Exemplos disso são o algoritmo de solução ARTEMiS-SSN da OPAL-RT que combina os métodos de solução por espaço de estados e de análise nodal [107], [108] e o mais recente simulador da RTDS que permite simular redes com centenas de nós trifásicos sem efetuar desacoplamento do sistema [109].

4.1.2. Co-simulação multitaxa

Entende-se por co-simulação à solução numérica de um sistema utilizando duas ou mais ferramentas de simulação de forma cooperativa. Nesta técnica, cada simulador tem seu próprio algoritmo de solução, executa independentemente a solução de seu modelo (que representa uma parte do sistema global) e interage com o outro simulador dinamicamente através do intercâmbio de variáveis de acoplamento [110]. Quando os subsistemas são resolvidos com passos diferentes, esta co-simulação recebe a denominação de multitaxa. Neste tipo de co-simulação, representado esquematicamente pela Figura 4.2, existe um instante periódico de intercâmbio de informações (Δt_{com}), múltiplo dos passos de solução (Δt_1 e Δt_2).

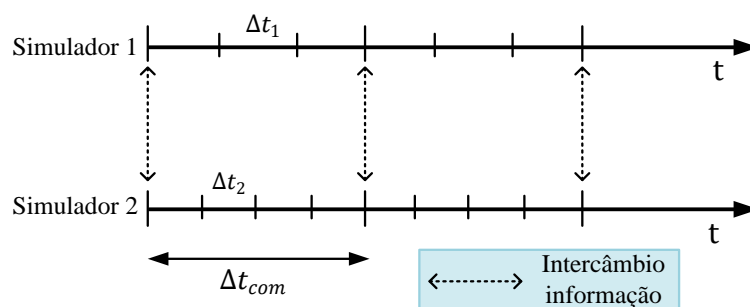


Figura 4.2 – Representação dos passos de simulação e de intercâmbio de informações numa co-simulação multitaxa.

Dado que numa co-simulação existem intervalos discretos de intercâmbio de informações entre os dois simuladores, o acoplamento ideal entre dois subsistemas, representado pela Figura 4.3(a) e equação (4.1), não é possível. Assim, técnicas focadas em quebrar esse laço algébrico, mediante a inclusão de dinâmicas artificiais à equação de acoplamento ideal têm sido propostas [110]. Entre elas, uma das alternativas de mais simples implementação é a de adicionar um atraso de Δt_{com} nos sinais de entrada de cada subsistema (que constituem as saídas do outro subsistema), conforme mostrado pela Figura 4.3(b) e equação (4.2) [111]. A aplicação deste método em sistemas elétricos recebe o nome de interface de transformador ideal [112].

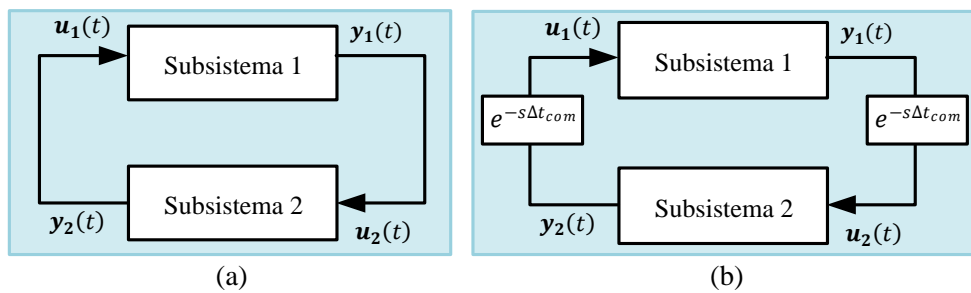


Figura 4.3 – Laço algébrico entre dois subsistemas: (a) caso de acoplamento ideal; (b) acoplamento mediante atraso.

$$\mathbf{u}_1(t) = \mathbf{y}_2(t) \quad \text{e} \quad \mathbf{u}_2(t) = \mathbf{y}_1(t) \quad (4.1)$$

$$\mathbf{u}_1(t) = \mathbf{y}_2(t - \Delta t_{com}) \quad \text{e} \quad \mathbf{u}_2(t) = \mathbf{y}_1(t - \Delta t_{com}) \quad (4.2)$$

A técnica de co-simulação multitaxa tem sido aplicada para estudos nos quais modelos chaveados de conversores são conectados a sistemas elétricos de potência detalhados [5], [7], [38], [57], [113]. Estas aplicações partem da premissa de que apesar da dinâmica interna de alta velocidade dos conversores, sua interação com o restante do sistema pode ser estudada ignorando os transitórios mais rápidos. Assim, o princípio dessas aplicações é o de simular os conversores chaveados com um passo de simulação “menor” (conforme discussões das Seções 1.3 e 1.4), e utilizar um passo “maior” para simular o restante do sistema de potência. Considerando a Figura 4.1, pode ser interpretado que essa abordagem tem como vantagem um requisito de potência de computação menor do que seria necessário para simular todo o sistema elétrico com o passo de simulação menor.

4.2. Descrição do SDTR integrado

4.2.1. Plataformas de *hardware* utilizadas

O SDTR integrado é formado por dois componentes: a) o SDTR/FPGA, proposto e validado no Capítulo 3 e b) um SDTR comercial, modelo OPAL-RT 5600. O DSP utilizado no Capítulo 3 como HUT é também acoplado ao SDTR integrado, criando uma bancada HIL como a representada esquematicamente pela Figura 4.4.

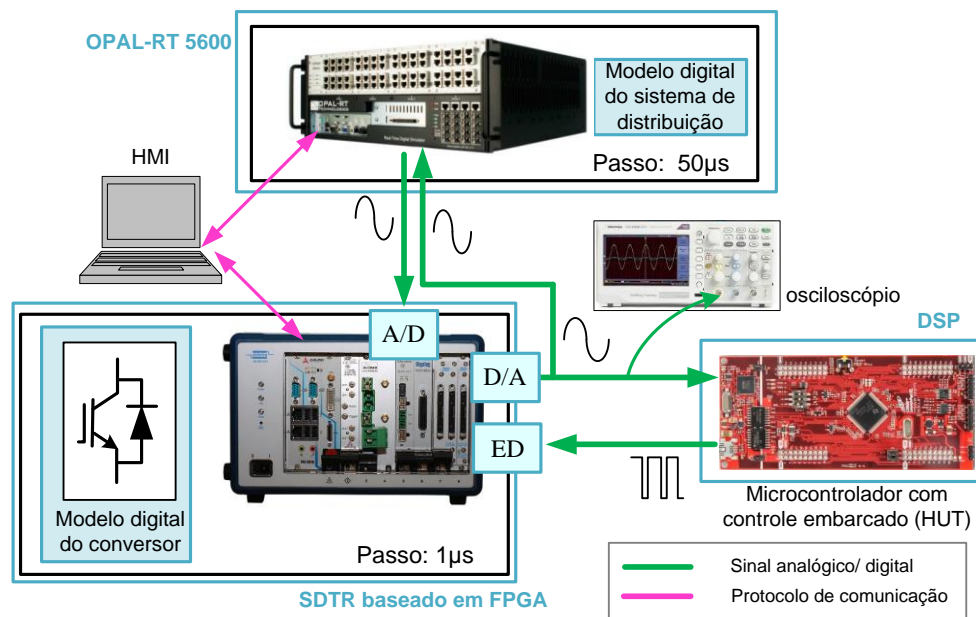


Figura 4.4 – Representação da bancada HIL para o SDTR integrado. SDTR: Simulador Digital em Tempo-Real; D/A: Conversor digital/analógico; ED: Entrada Digital; A/D: Conversor analógico/digital; HUT: Equipamento sobre ensaio; HMI: Interface homem-máquina.

As características mais relevantes do SDTR comercial são:

- (1) Dois núcleos ativos de processamento Intel Xeon E5-2667 com sistema operativo Linux em tempo real;
- (2) Integração numérica através de um método de quinta ordem denominado ARTEMiS Art5 [108];
- (3) Periféricos de entrada digital (OP53353 de 32 canais), saída digital (OP5360 de 32 canais), entrada analógica (OP5340 de 16 canais) e saída analógica (OP5330 de 16 canais);
- (4) Em contrapartida, este equipamento carece de módulo FPGA apropriado para simulação de modelos chaveados de conversores eletrônicos.

A programação do SDTR comercial é realizada de forma gráfica através do software Simulink. O software RT-LAB, desenvolvido pelo fabricante do SDTR, realiza a compilação do modelo desenvolvido em Simulink, de forma a criar um arquivo executável que roda nos processadores do RTDS.

4.2.2. Interface de acoplamento

O estudo de técnicas de acoplamento entre os subsistemas de uma co-simulação multitaxa tem recebido muita atenção, onde as principais abordagens que têm sido propostas são as de interface de transformador ideal [112], [114], equivalente multi-área de Thevenin [115] e técnicas baseadas em modelos de linha de parâmetros distribuídos [57], [116]. Essa área de pesquisa, contudo, está fora do foco desta tese, na qual é aplicado o método de transformador ideal, dada sua facilidade de implementação e sua maior robustez diante do ruído e inexatidão do meio físico utilizado para realizar a troca de informações entre os SDTR (canais analógicos)¹⁶.

O método de transformador ideal pode ser explicado com o diagrama da Figura 4.5, onde o Ponto De Separação (PDS) representa a localização onde é efetuada a separação do sistema original. Conforme apresentado, os subsistemas resultantes são acoplados na simulação através de um par de fontes de tensão e de corrente, cujos valores instantâneos são determinados pelo outro subsistema no instante anterior de comunicação (Δt_{com}), conforme representa o atraso de (4.2).

A corrente chaveada i_2 na Figura 4.5(b) contém harmônicos que não são representáveis com o passo de simulação adotado no SDTR comercial. Neste sentido, entretanto, a adequação de sinais apresentada na Seção 3.3.1, baseada no sincronismo entre o DSP e o módulo FPGA permite exportar uma onda de corrente com reduzida interferência devido ao fenômeno de *Aliasing*. Assim, o valor escolhido neste trabalho para o passo de troca de informações entre simuladores (Δt_{com}) é o equivalente a um período de chaveamento PWM, ou seja, 100 μ s.

¹⁶ Nesta tese não foi empregado um protocolo digital para a transferência de dados entre os SDTRs como consequência da indisponibilidade de um módulo periférico adicional, capaz de implementar tal protocolo, no sistema PXI utilizado como SDTR/FPGA.

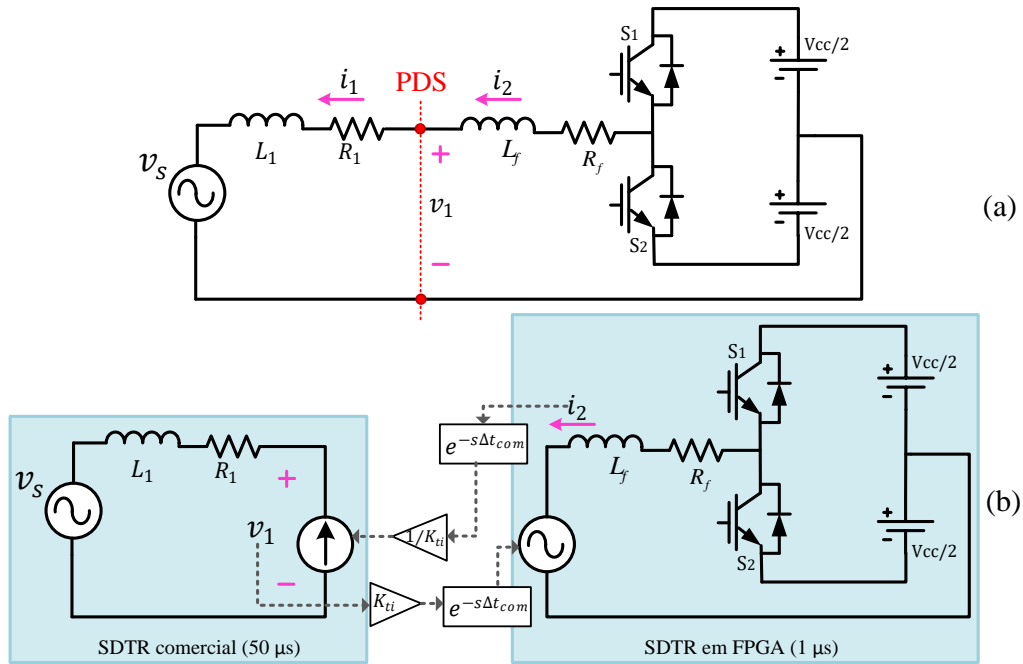


Figura 4.5 – Representação do método de interface de transformador ideal: (a) sistema elétrico original; (b) subsistemas acoplados. K_{ti} : Constante de transformação. PDS: Ponto de separação.

4.3. Simulações e discussões

4.3.1. Caso I: conexão de VSC com equivalente de rede

O primeiro caso de estudo consiste na co-simulação do sistema elétrico descrito pela Figura 4.6, no qual um conversor VSC de dois níveis é modelado no SDTR/FPGA (com Δt_1 de 1 μ s), enquanto que o modelo do equivalente de rede no qual esse conversor é acoplado simula-se no SDTR comercial (com Δt_2 de 50 μ s).

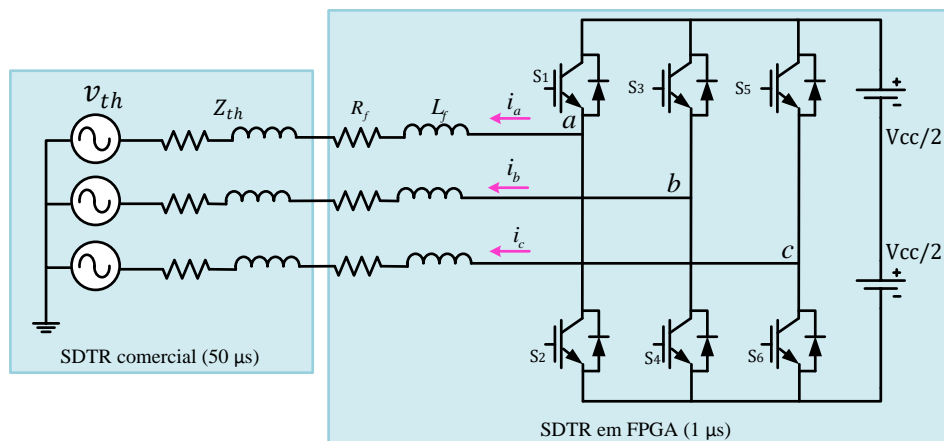


Figura 4.6 – Diagrama do sistema de co-simulação do caso I.

Nesta análise é considerado que este conversor VSC, cujos parâmetros são os mesmos apresentados na Tabela 3.5, representa um inversor fotovoltaico conectado à

rede. A tensão nominal da fonte equivalente é de 440 Vrms fase-fase, o valor da impedância de Thevenin foi escolhido de tal forma que a relação $|Z_{th}|/|Z_f|$ seja de 0,1. O que corresponde, utilizando a definição de [117], a uma relação de curto circuito (SCR – *Short Circuit Ratio*) entre a rede equivalente e o conversor VSC de 67.

O DSP da bancada HIL da Figura 4.4 controla o modelo do VSC com uma rotina embarcada de controle que está baseada naquela apresentada na Seção 3.5.3.2 e Apêndice C. Efetivamente, a única diferença no algoritmo de controle utilizado nos ensaios deste capítulo, em relação ao empregado na validação em malha fechada do Capítulo 3, é a modificação da fórmula para cálculo das correntes de referência, não sendo utilizada a equação (A.10) mas a expressão que segue:

$$\begin{bmatrix} i_{dref} \\ i_{qref} \end{bmatrix} = \begin{cases} \frac{1}{v_{sd}^2 + v_{sq}^2} \cdot \begin{bmatrix} v_{sd} & v_{sq} \\ v_{sq} & -v_{sd} \end{bmatrix} \times \begin{bmatrix} P_{ref} \\ Q_{ref} \end{bmatrix} & \text{se } (v_{sd}^2 + v_{sq}^2) > 0,64 \text{ pu} \\ \begin{matrix} 0 \text{ pu} \\ 0,5 \text{ pu} \end{matrix} & \text{se } (v_{sd}^2 + v_{sq}^2) \leq 0,64 \text{ pu.} \end{cases} \quad (4.3)$$

A equação (4.3) tem como objetivo mudar as referências de corrente (i_{dref} e i_{qref}) de tal forma que o conversor produza potência reativa na condição de um evento de afundamento de tensão, o que em redes predominantemente indutivas contribui com a redução do grau de afundamento no ponto de conexão do conversor. A escolha do limite de tensão a partir do qual esta função simplificada é executada (0,64 pu) se fundamenta na Seção 7 (Requisitos de suportabilidade a subtensões decorrentes de faltas na rede) da norma técnica brasileira ABNT NBR 16149 para sistemas fotovoltaicos [118]. Destaca-se que não é objetivo desta tese o estudo e/ou aplicação de técnicas de controle que satisfaçam integralmente os requisitos da norma citada.

O primeiro ensaio dinâmico realizado neste caso de estudo consiste na aplicação de um afundamento trifásico de tensão nos terminais da fonte equivalente modelada no SDTR comercial (v_{th}) na Figura 4.6. Esse afundamento tem amplitude de 0,3 pu (tensão residual de 0,7 pu) e duração de 100 ms. As referências de potência para o controlador embarcado no DSP, durante toda a simulação em tempo-real, são: $P_{ref} = 0,7$ pu e $Q_{ref} = 0$ pu. Essas referências caracterizam as condições iniciais do sistema simulado antes do evento de afundamento, bem como o ponto de operação pós-evento.

A Figura 4.7(a) mostra os resultados experimentais (sinais exportados pelo SDTR/FPGA no ponto de medição do osciloscópio, conforme Figura 4.4) durante o referido evento. Estas são ondas analógicas representativas do comportamento das

variáveis do sistema, adaptadas à faixa de operação do conversor A/D do DSP utilizado, para as quais aplicam os fatores de conversão de 560/1,5 A/V nas ondas de corrente, e 400/1,5 V/V na onda de tensão. Observa-se que as correntes injetadas pelo modelo do VSC se encontram em fase com a tensão da rede antes do afundamento, bem como instantes após o término do evento, ocorrendo o retorno para as condições de regime permanente (estabelecidas pelas referências de potência P_{ref} e Q_{ref}). Durante o afundamento, as correntes injetadas pelo VSC passam a estar adiantadas das tensões da rede por um ângulo próximo a $\pi/2$, conforme desejado pela função simplificada de (4.3). O atingimento deste ponto temporário de operação, entretanto, ocorre de forma gradual, com uma constante de tempo próxima a 90 ms.

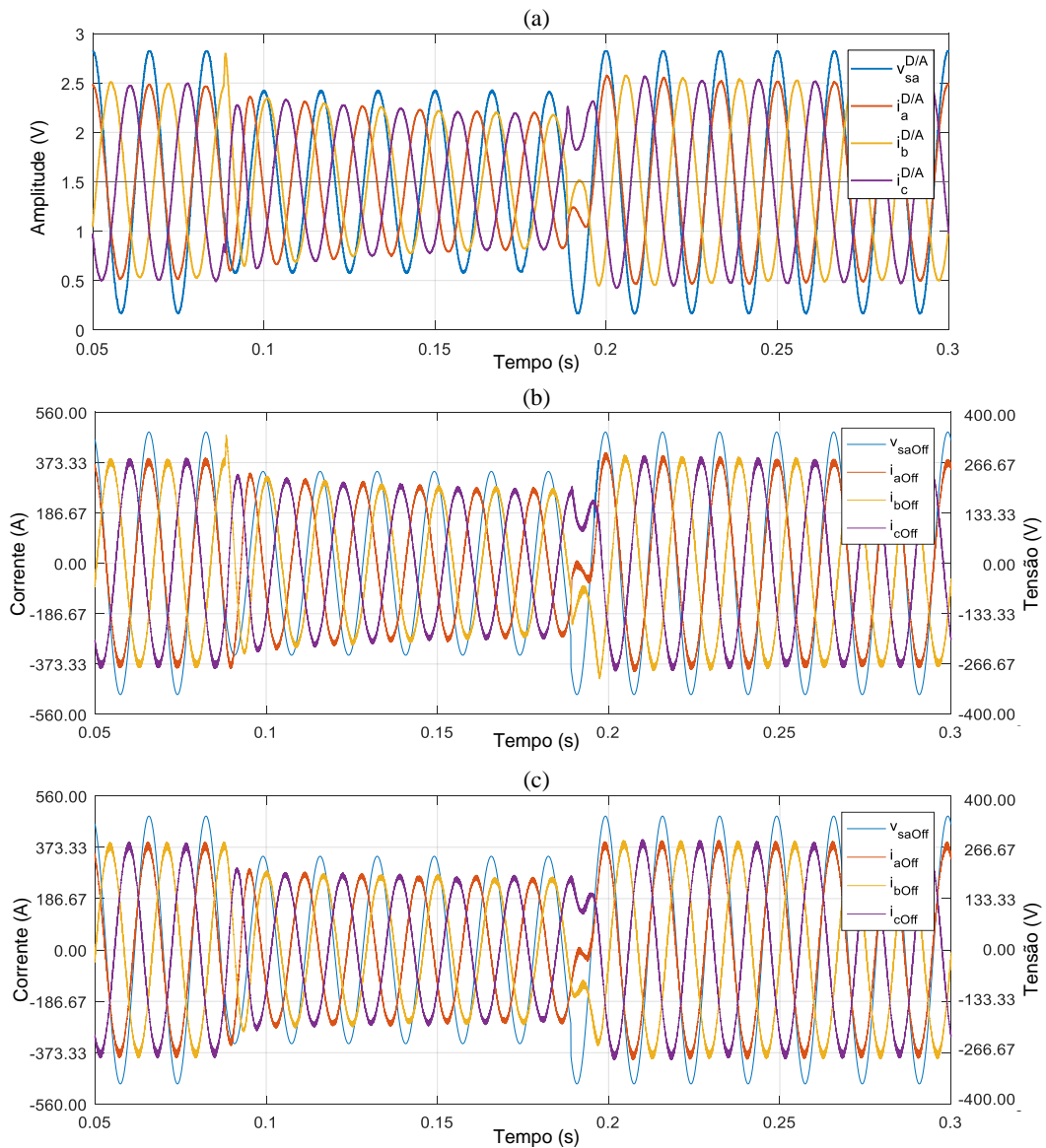


Figura 4.7 – Correntes injetadas pelo modelo do VSC durante o evento de afundamento: (a) resultados experimentais da bancada HIL; (b) resultados de simulação *off-line* com representação dos atrasos da bancada HIL; (c) resultados de simulação *off-line* sem atrasos.

Os resultados experimentais foram comparados com resultados de simulações *off-line* no Simulink. Nas simulações *off-line*, o sistema elétrico da Figura 4.6 é solucionado utilizando um único passo de simulação de 1 μ s, uma vez que o mesmo hardware (um computador pessoal com sistema operativo Windows rodando Simulink) realiza a solução do sistema e a execução da mesma rotina de controle embarcada no DSP para o caso experimental. Para esta comparação dois modelos *off-line* foram criados, no primeiro deles foram introduzidos atrasos nos sinais de realimentação para o controlador, de forma a representar o atraso existente na bancada HIL experimental (como consequência do método de sincronismo do SDTR/FPGA para exportação de sinais e da interface de transformador ideal na co-simulação), os resultados deste primeiro modelo são mostrados na Figura 4.7(b). No segundo modelo *off-line*, cujos resultados são apresentados na Figura 4.7(c), não são incluídos atrasos nos sinais de controle, representando assim o caso ideal.

Dado que as duas figuras dos resultados de simulação *off-line* utilizam linhas de grade que coincidem com a escala equivalente das ondas da Figura 4.7(a), é possível realizar uma comparação visual entre os resultados experimentais e de simulação *off-line*. Assim, observa-se que os conjuntos de formas de onda das Figura 4.7(a) e Figura 4.7(b) são bastante similares, existindo em ambos os casos uma sobrecorrente temporária no instante de afundamento, bem como uma dinâmica amortecida para atingir as referências de corrente durante o evento. A comparação numérica entre estas correntes trifásicas, utilizando (3.1), dá como resultado um indicador de erro ε_{exp} de 7,87%, que se considera satisfatório. Destaca-se que a avaliação de ε_{exp} em condições de operação de regime permanente, por exemplo após a remoção do afundamento, resulta em 4,1%.

Nos resultados da Figura 4.7(c), por outro lado, observa-se uma melhor resposta do controlador de corrente durante o evento de afundamento, uma vez que não existem sobrecorrentes transitórias e se evidencia um atingimento mais rápido das referências durante o amortecimento. A comparação destes resultados de simulação *off-line* com os resultados experimentais dá como resultado um indicador ε_{exp} de 9,90%, que se considera válido para examinar as características de funcionamento do HUT.

O segundo ensaio efetuado com este modelo de sistema elétrico consiste na mudança instantânea da frequência da fonte equivalente v_{th} . Com base nos requisitos da norma técnica brasileira para inversores fotovoltaicos, a frequência dessa fonte foi modificada

de 60 Hz para 57,5 Hz [118]. Neste contexto, a Figura 4.8(a) mostra os resultados experimentais, enquanto que a Figura 4.8(b) mostra os resultados de simulação *off-line* sem introdução de atrasos nos sinais de controle. Observam-se um conjunto de formas de onda bastante similares, para o qual o indicador de erro ε_{exp} é igual a 7,2%. Para o caso da comparação entre os resultados experimentais e os de simulação *off-line* com atrasos (formas de onda não apresentadas por simplicidade) ε_{exp} é igual a 6,8%.

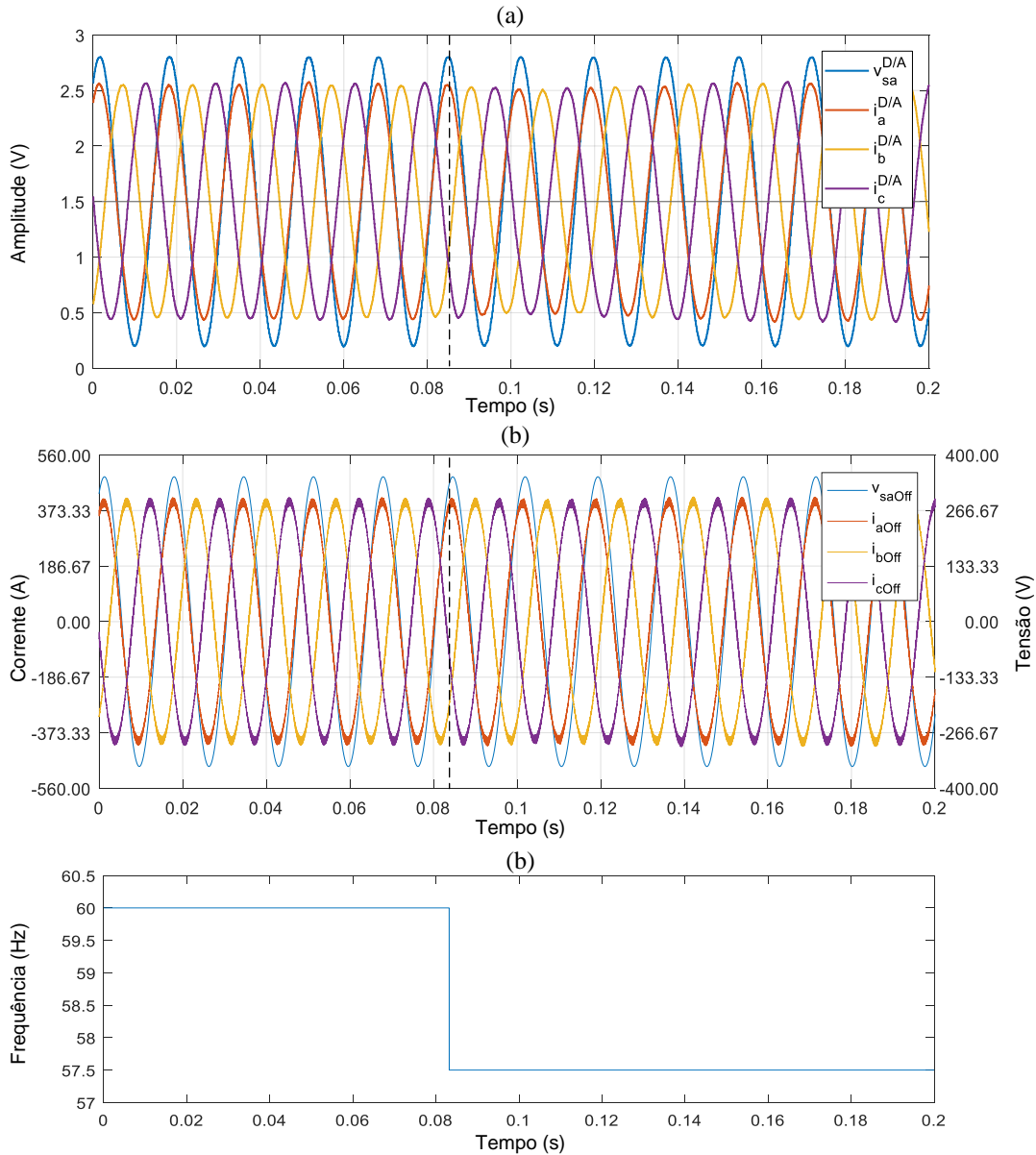


Figura 4.8 – Correntes injetadas pelo modelo do VSC e tensão da fase a no PCC durante o evento de degrau de frequência: (a) resultados da bancada HIL, medidos com osciloscópio; (b) resultados de simulação *off-line* sem atrasos. Potência de referência de 0,85 pu.

Uma visão ampliada da figura anterior é mostrada na Figura 4.9. Destaca-se que nesta figura as medições de corrente foram ajustadas à escala de valor real, para facilitar a comparação visual.

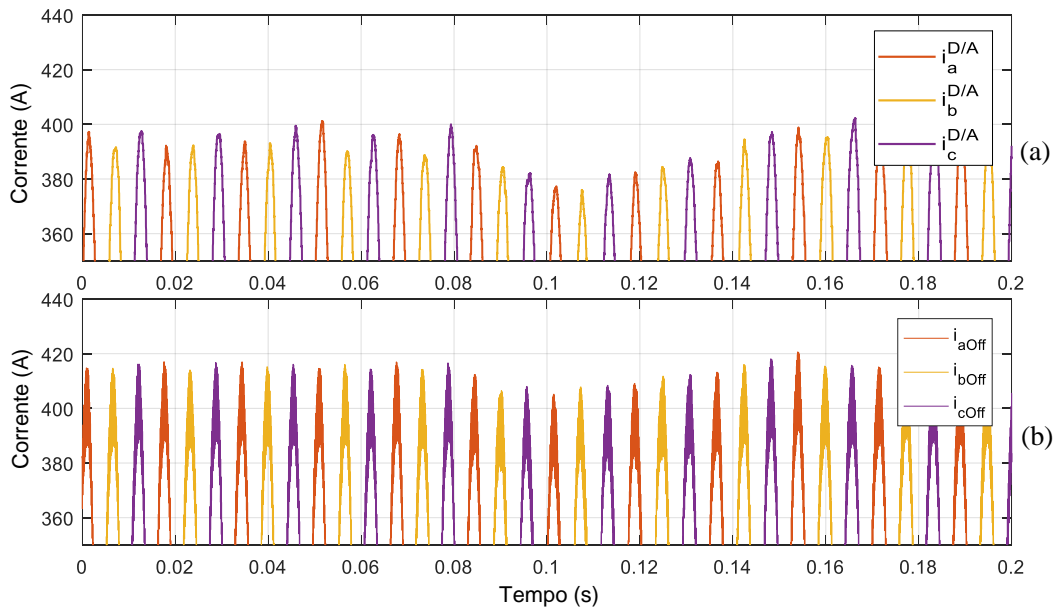


Figura 4.9 – Zoom no eixo vertical nas ondas de corrente da Figura 4.8.

4.3.2. Caso II: conexão com rede *benchmark* de distribuição

Com o intuito de exemplificar a aplicação do SDTR integrado na representação de sistemas de potência com maior complexidade, no segundo caso de estudo o mesmo conversor VSC fotovoltaico utilizado anteriormente (simulado no SDTR/FPGA) é acoplado ao sistema de distribuição *benchmark* IEEE 34 barras (simulado no SDTR comercial). Segundo o documento de referência deste *benchmark* [119], esse sistema elétrico está baseado numa rede real do Arizona (USA) com tensão nominal de 24,9 kV caracterizada por longas distâncias, dois reguladores de tensão, um transformador OLTC (*on load tap changer*) para uma curta seção de 4,16 kV, cargas desequilibradas e capacitores *shunt*. Conforme descreve a Figura 4.10, neste trabalho o barramento 830 é escolhido como o ponto de conexão do VSC. O ganho utilizado no método de interface de transformador ideal (K_{ti}) é ajustado para permitir a conexão deste conversor numa rede de maior nível de tensão nominal.

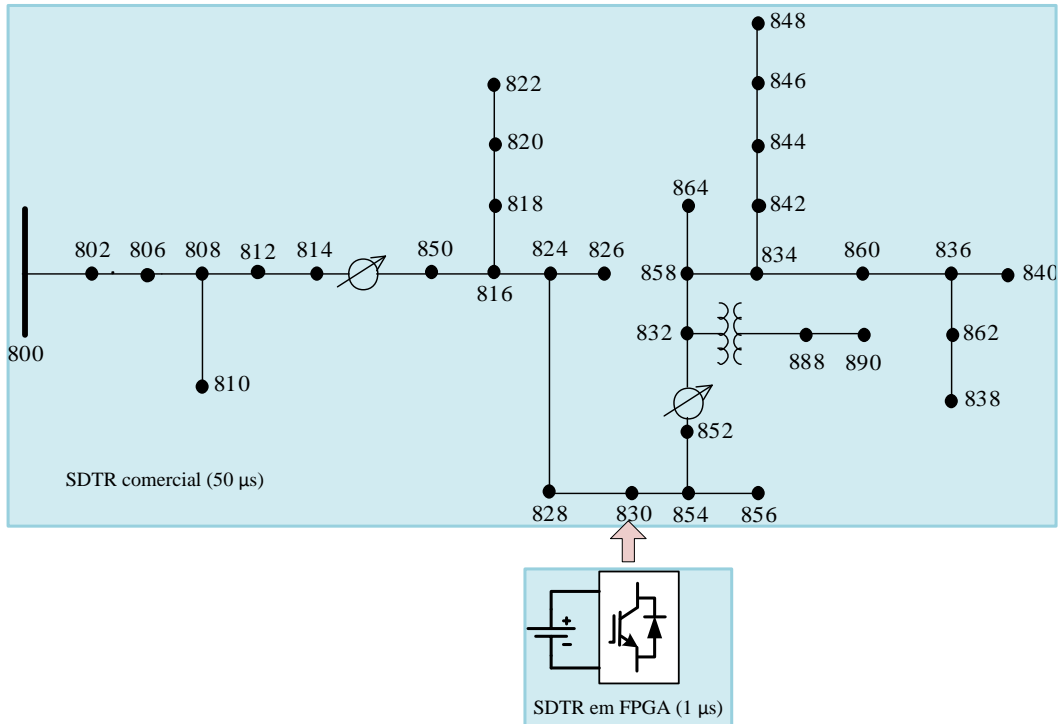


Figura 4.10 – Diagrama unifilar do sistema de co-simulação do caso de estudo II, baseado no *benchmark* IEEE 34 barras.

O modelo do sistema *benchmark* IEEE 34 barras programado nesta tese considera todas as características descritas em [119] exceto os reguladores automáticos de tensão, que são representados como transformadores com relações de transformação fixas, ajustadas para obter um perfil de tensão em regime permanente similar ao apresentado no documento de referência. O tempo de compilação (transformação do código desenvolvido em Simulink num arquivo executável em tempo real) deste modelo no SDTR comercial foi de 5 min.; enquanto que seu tempo de execução médio é de 42 μ s, portanto o passo de solução adotado para esta parte do modelo (Δt_1) foi de 50 μ s.

A Figura 4.11(a) mostra as formas de onda representativas da tensão da fase *a* do barramento 830 ($v_{830a}^{D/A}$) e das três correntes injetadas pelo modelo do VSC fotovoltaico ($i_{VSCa}^{D/A}$, $i_{VSCb}^{D/A}$ e $i_{VSCc}^{D/A}$) durante uma variação da potência ativa de referência de 0 para 1 pu. Observa-se um comportamento adequado do controlador embarcado, onde as referências são atingidas em menos de meio ciclo da frequência fundamental. A Figura 4.11(b), por sua parte, mostra as correntes eficazes injetadas pelo VSC durante a referida mudança de referência. Destaca-se que para o cálculo desses valores eficazes é utilizada uma janela deslizante cujo período é equivalente a três ciclos da onda fundamental de 60 Hz, o que justifica o atraso mostrado para atingir o valor de 1 pu.

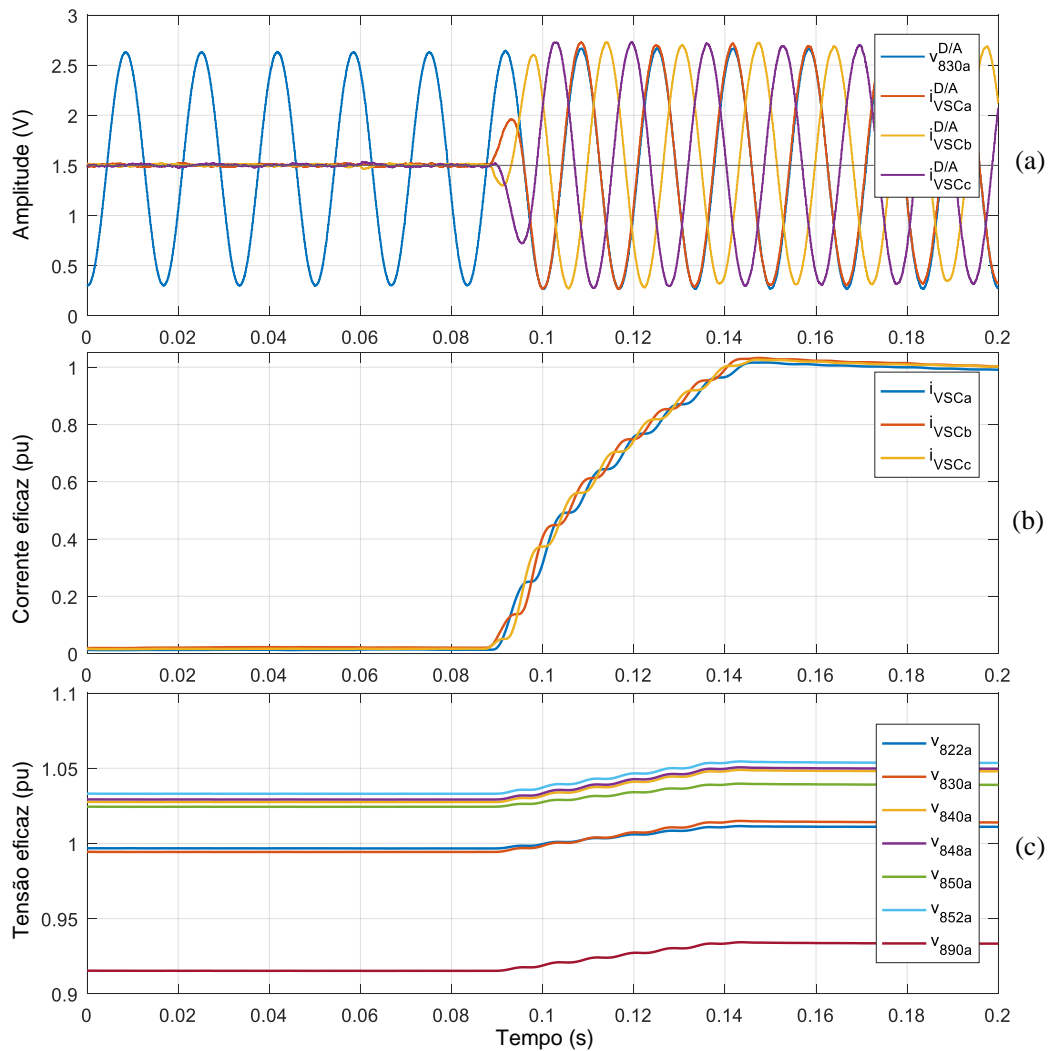


Figura 4.11 – Resultado da simulação de aumento de geração pelo VSC: (a) ondas representativas medidas pelo osciloscópio; (b) valor eficaz das correntes injetadas; (c) valor eficaz de algumas tensões na rede.

Ao correlacionar a Figura 4.11(b) com a Figura 4.11(c) observa-se que a injeção de potência ativa na rede modifica o perfil de tensão do sistema, ocasionando que em algumas das barras apresentadas o limite de 1,05 pu seja ultrapassado. Este é um problema conhecido da penetração de geração distribuída nas redes de distribuição. Neste sentido, o SDTR integrado poderia ser utilizado em futuros estudos detalhados, focados em avaliar a interação entre os controladores dos reguladores de tensão e o controlador do conversor, de forma a encontrar a maneira mais apropriada do ponto de vista técnico/econômico para evitar tensões fora das margens definidas pelos procedimentos de rede.

O segundo ensaio realizado com este sistema é a aplicação de um curto circuito trifásico no barramento 814. Neste contexto, se destaca novamente que não é objetivo deste trabalho implementar no HUT uma técnica de controle o suficientemente robusta

para suportar afundamentos de tensão de até 100% de profundidade, como exigido, pelo período de 200 ms, pela norma técnica brasileira [118]. Assim, teve-se o cuidado de selecionar uma impedância de curto circuito tal que originasse um afundamento de tensão de aproximadamente 40%.

A Figura 4.12(a) mostra as tensões no barramento 830 (ponto de conexão do VSC) durante o referido evento. Como principal característica destas ondas se destaca os transitórios EMT com pequena constante de tempo, originados logo após a aplicação e remoção do curto circuito. Esse comportamento dinâmico constitui a resposta natural dos elementos armazenadores de energia do sistema *benchmark* diante de tais eventos e caracteriza o valor agregado do SDTR integrado e, numa visão mais geral, dos ensaios HIL. Isto porque ensaios normativos tradicionalmente utilizam equivalentes de rede que, como observado na seção anterior, são incapazes de representar essas dinâmicas que podem chegar a comprometer a operação do VSC.

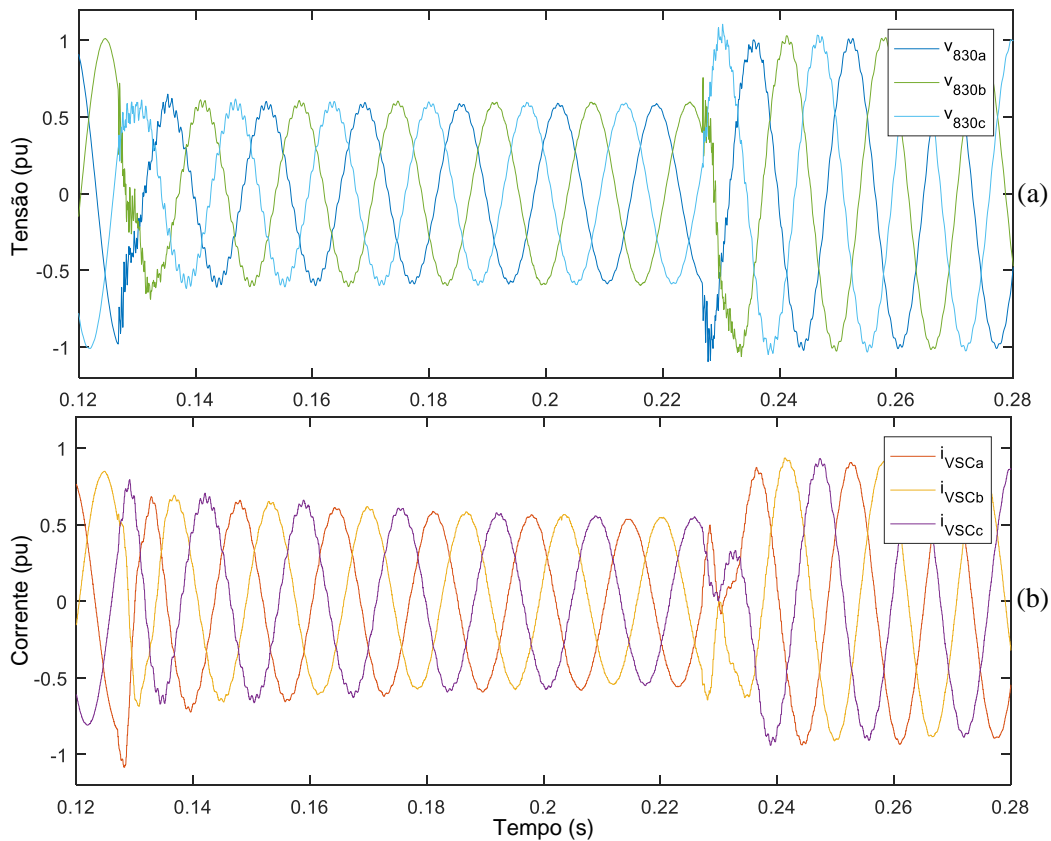


Figura 4.12 – Resultados da bancada de co-simulação multitaxa durante o evento de curto-circuito na barra 814: (a) tensões no barramento 830; (b) correntes injetadas pelo VSC.

De forma similar aos resultados obtidos no caso de estudo da seção anterior, as ondas de corrente da Figura 4.12(b) mostram um transitório amortecido para chegar a seus valores de referência durante o afundamento. Outra característica que merece destaque

são as oscilações harmônicas de pequena amplitude originadas pela interação do sistema de controle do VSC com o restante da rede, que surgem após os eventos de transientes eletromagnéticos e são atenuadas em cerca de 60 ms.

De forma a validar os resultados da bancada HIL de co-simulação multitaxa, foi realizada uma simulação *off-line* onde o sistema de distribuição *benchmark* IEEE 34 barras, o modelo de conversor VSC fotovoltaico e seu controlador são simulados no ambiente Simulink, utilizando um passo de solução fixo de 1 μ s. De forma similar ao realizado na seção anterior, um atraso foi introduzido nos sinais de realimentação para o controlador do VSC. Assim, a Figura 4.13 mostra as tensões e correntes na barra 830 para o caso da simulação *off-line*, para comparação com os resultados experimentais da Figura 4.12. Pode ser notado que na Figura 4.13 também existem oscilações harmônicas após os transitórios EMT, contudo as mesmas são atenuadas de forma mais rápida, o que caracteriza a principal diferença entre estes resultados e os obtidos na bancada HIL. A avaliação de (3.1) dá como resultado indicadores de erro de 9,8% para o caso das tensões ($\varepsilon_{exp-v830}$) e 10,8% para o caso das correntes ($\varepsilon_{exp-iVSC}$).

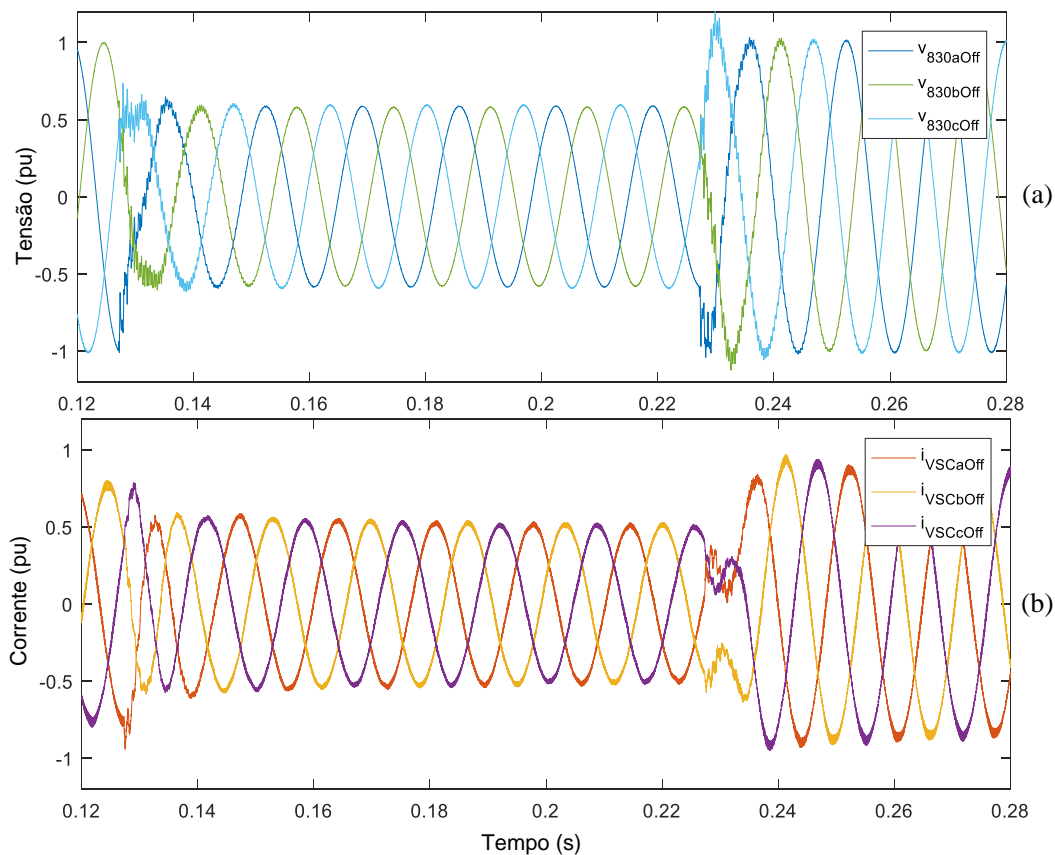


Figura 4.13 – Resultados de simulação *off-line* durante o evento de curto-circuito na barra 814: (a) tensões no barramento 830; (b) correntes injetadas pelo VSC.

Salienta-se que a simulação *off-line* apresenta certas idealidades em comparação com a bancada HIL experimental, entre elas, tratamento digital sem interferência de quantização dos periféricos de E/S e passo de simulação de 1 μ s para todo o sistema. Destaca-se, ainda, que a execução desta simulação por um período de 0,7 s num computador pessoal com processador Intel Core i7 e sistema operativo Windows 10 teve um tempo de execução próximo de 6 min.

Para finalizar a análise deste evento, a Figura 4.14 mostra as medições com osciloscópio dos sinais analógicos exportados pelo SDTR/FPGA. Esta figura utiliza um período tal que permite mostrar o atingimento da condição de operação pré-falta depois de finalizados os eventos. O adequado funcionamento do controle simplificado de deslocamento de fase e injeção de potência reativa durante o afundamento é também apresentado.

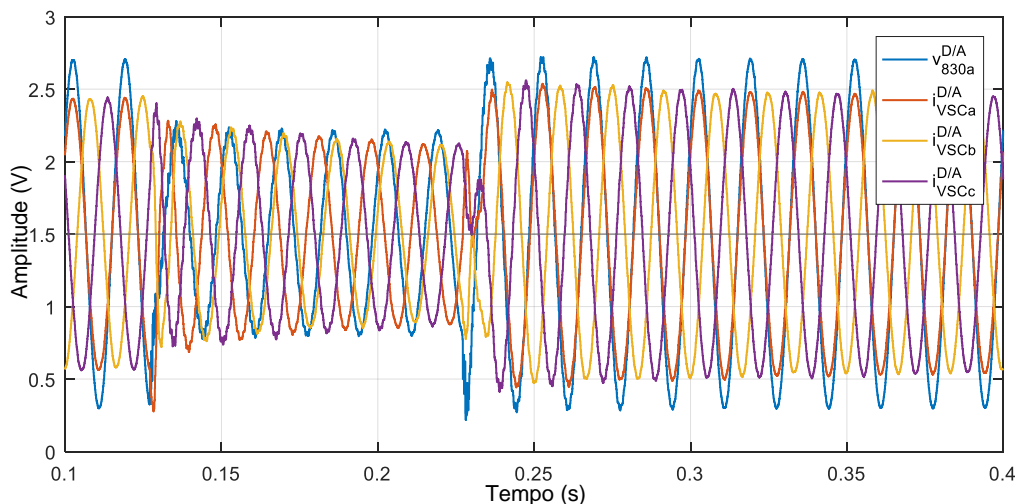


Figura 4.14 – Ondas exportadas pelo SDTR/FPGA durante o evento.

4.3.3. Discussão

Os casos de estudo apresentados evidenciaram a utilidade do SDTR integrado na avaliação de dispositivos com controle embarcado, bem como a aplicabilidade da estratégia de modelagem de chaves ADC-I numa co-simulação multitaxa. Acredita-se que desenvolvimentos nessa linha de pesquisa sejam importantes no contexto atual e futuro, no qual o interesse de avaliar a integração de DER fundamentados em conversores vem se tornando crescente em função da maior penetração de tais equipamentos nas redes de distribuição e do surgimento de sistemas centralizados de controle. Outra área na qual a tecnologia de co-simulação multitaxa em tempo real pode ser aplicada é, por exemplo, na avaliação de controladores de conversores para sistemas

de transmissão em corrente contínua (HVDC), em especial do tipo MMC (*Modular Multilevel Converter*), dado seu maior número de componentes e, conseqüentemente, complexidade para representação em tempo real.

Apesar do mencionado anteriormente, destaca-se que os resultados obtidos pelo SDTR integrado incluem comportamentos dinâmicos artificiais, conseqüência dos atrasos introduzidos pela própria ferramenta. Entretanto, as comparações realizadas entre os resultados experimentais da bancada HIL e os determinados por um simulador *off-line* indicaram que as imprecisões originadas por esses componentes dinâmicos são inferiores a 11% para todos os ensaios realizados. A implicação destes erros na fidelidade da modelagem é um aspecto que merece uma avaliação específica considerando o sistema simulado, e que depende também do ponto elétrico escolhido para efetuar a separação dos subsistemas com a interface de transformador ideal [61], [120]. Nos casos de simulação apresentados, entretanto, observou-se que a inexatidão da bancada HIL não comprometeu a avaliação de conformidade do HUT (controlador) com relação aos requisitos da norma. Portanto, se considera que, desde um ponto de vista normativo, o uso do SDTR integrado é válido.

Os atrasos introduzidos pelo SDTR integrado são resultado das limitações dos componentes de *hardware* utilizados para seu desenvolvimento. As limitações mais relevantes são a interface analógica assíncrona entre o SDTR/FPGA e o SDTR/CPU, cuja baixa exatidão impede o uso de algoritmos de acoplamentos mais complexos do que o de transformador ideal, e o atraso inserido pela abordagem de sincronismo entre o DSP e SDTR/FPGA como conseqüência das restrições do conversor D/A.

Entretanto, as restrições acima mencionadas não impedem que a metodologia apresentada seja estendida para aplicação em sistemas de maior porte e com resultados mais próximos exatos, uma vez superadas as limitações das interfaces.

Capítulo 5

Considerações finais

5.1. Conclusões

Neste trabalho foi proposta e validada uma técnica complementar para modelagem chaveada de conversores controláveis para simulação em tempo real em FPGA. Essa técnica, denominada ADC-I e baseada no método G-ADC, se fundamenta no uso de um algoritmo de inicialização para determinar o valor da corrente histórica do modelo ADC da chave, de forma a diminuir a amplitude do transitório numérico originado após uma mudança de estado. Conforme mostrado pelas validações *off-line* e em tempo real, a técnica proposta oferece um nível de exatidão similar ao do método G-ADC (ou seja, erro de corrente inferior a 0,5% nos resultados *off-line* para um VSC trifásico), porém com menor uso dos recursos de *hardware* do módulo FPGA utilizado como núcleo computacional. Assim, a técnica ADC-I serve para ter maior capacidade de representação na simulação em tempo real em FPGA de sistemas elétricos com conversores eletrônicos.

Dois simuladores digitais em tempo real foram desenvolvidos nesta tese. O primeiro deles, que utiliza um módulo FPGA genérico como núcleo computacional, foi elaborado utilizando um algoritmo de solução baseado na técnica de Análise Nodal Modificada, representação numérica em ponto fixo e passo de simulação de 1 μ s. Este SDTR/FPGA tem como limitação, em relação com os SDTR baseados em plataformas genéricas previamente reportados na literatura, a indisponibilidade de canais de saída analógicos na frequência de cálculo. Esta característica motivou a proposição de uma abordagem de adequação de sinais baseada no sincronismo entre o simulador e os pulsos de chaveamento do controlador.

Os ensaios experimentais em malha aberta e em malha fechada mostraram um funcionamento adequado dessa abordagem de adequação de sinais, uma vez que as ondas medidas tiveram componentes harmônicos com amplitudes inferiores a -55 dBm. Os indicadores de erro dos resultados experimentais, definidos em relação com os resultados *off-line*, para a operação do SDTR/FPGA na frequência fundamental foram

inferiores a 1,5% para o caso em malha aberta e a 4% em malha fechada o que se considera satisfatório para fins práticos. Assim o SDTR/FPGA serve para simular em tempo real sistemas elétricos contendo conversores eletrônicos e redes simplificadas de distribuição. A principal aplicação deste simulador é a sua integração em bancadas HIL para avaliação de controladores embarcados. O uso do SDTR/FPGA, entretanto, está limitado à representação de até 6 pares complementares de chaves, como consequência do número de canais digitais, e à representação de harmônicos até a terceira ordem, como consequência do atraso introduzido pela abordagem de adequação de sinais. Por último, uma vez que o SDTR/FPGA não utiliza nenhum algoritmo de sincronização de chaveamento para lidar com os ITS, a frequência de chaveamento está limitada à faixa entre 0 e 15 kHz.

O segundo SDTR desenvolvido nesta tese, denominado SDTR integrado, consiste numa plataforma de co-simulação multitaxa onde o SDTR/FPGA interage em tempo real com um SDTR comercial fundamentado em processadores tipo CPU e com passo de simulação de 50 μ s. Este SDTR combina as capacidades de modelagem das duas ferramentas de forma que serve para simular em tempo real sistemas elétricos contendo conversores eletrônicos e redes detalhadas de distribuição, conforme verificado experimentalmente pela modelagem e simulação de um modelo da rede *benchmark* IEEE 34 barras. Além das limitações do SDTR/FPGA, o SDTR integrado tem como desvantagem a introdução de dinâmicas artificiais após eventos eletromagnéticos, conforme discutido no Capítulo 4. Entretanto, constatou-se que, desde o ponto de avaliação de conformidade do controlador externo, essas inexactidões não invalidam o uso da plataforma desenvolvida para o estudo da integração de DER em redes de distribuição.

5.2. Principais atividades futuras

Neste trabalho foram desenvolvidas duas ferramentas de simulação em tempo real aplicáveis à avaliação de controladores embarcados de conversores eletrônicos. Essas ferramentas passam a estar disponíveis para futuros estudos de caso. Exemplo de pesquisas que podem ser executadas com o SDTR integrado são as relacionadas com a análise de desempenho do funções ancilares por parte de DERs, característica que é reconhecida como importante no contexto das redes elétricas futuras ou *smart grids*. Adicionalmente, a metodologia apresentada para desenvolvimento do SDTR/FPGA

genérico pode motivar futuros trabalhos similares, em especial considerando a principal vantagem da técnica proposta para modelagem de chaves (ADC-I) que é o menor requisito de recursos de computação do que métodos de similares níveis de exatidão.

Apesar de suas limitações de *hardware*, o SDTR/FPGA desenvolvido neste trabalho foi utilizado de forma bem-sucedida numa bancada HIL onde uma rotina de controle de corrente no referencial $d-q$, embarcada em um microprocessador, foi testada. Entretanto, o atraso originado pela abordagem de adequação de sinais através de sincronismo entre o DSP e o SDTR repercute no desempenho dinâmico do sistema modelado. Assim, recomenda-se como atividade futura comparar os resultados da bancada HIL desenvolvida com os obtidos por um SDTR no qual o uso desta abordagem de adequação não seja requerido.

Com relação ao SDTR integrado, pode-se aprimorar a exatidão desta plataforma através do uso de outro meio físico de interface (por exemplo, um protocolo digital através de fibra ótica) e de algoritmos de acoplamento mais avançados.

Referências Bibliográficas

- [1] A. Isaacs, “Simulation Technology: The Evolution of the Power System Network [History],” *IEEE Power Energy Mag.*, vol. 15, no. 4, pp. 88–102, 2017.
- [2] M. Giesselmann, “Computer simulation of power electronics and motor drives,” in *Power Electronics Handbook*, Elsevier, 2011, pp. 1249–1273.
- [3] P. Kundur, *Power System Stability and Control*, 1°. Toronto, USA: McGraw-Hill Inc., 1994.
- [4] X. Guillaud *et al.*, “Applications of Real-Time Simulation Technologies in Power and Energy Systems,” *IEEE Power Energy Technol. Syst. J.*, vol. 2, no. 3, pp. 103–115, 2015.
- [5] J. Mahseredjian, V. Dinavahi, and J. A. Martinez, “Simulation Tools for Electromagnetic Transients in Power Systems: Overview and Challenges,” *IEEE Trans. Power Deliv.*, vol. 24, no. 3, pp. 1657–1669, 2009.
- [6] R. Kuffel, J. Giesbrecht, T. Maguire, R. P. Wierckx, and P. McLaren, “RTDS-a fully digital power system simulator operating in real time,” in *Proceedings 1995 International Conference on Energy Management and Power Delivery EMPD '95*, 1995, vol. 2, pp. 498–503 vol.2.
- [7] M. D. O. Faruque *et al.*, “Real-time simulation technologies for power systems design, testing, and analysis,” *IEEE Power Energy Technol. Syst. J.*, vol. 2, no. 2, pp. 63–73, 2015.
- [8] G. F. Lauss, M. O. Faruque, K. Schoder, C. Dufour, A. Viehweider, and J. Langston, “Characteristics and Design of Power Hardware-in-the-Loop Simulations for Electrical Power Systems,” *IEEE Trans. Ind. Electron.*, vol. 63, no. 1, pp. 406–417, Jan. 2016.
- [9] M. Roitman, E. H. Watanabe, and F. J. Lyra, “Power systems analog simulation enhancement using a new programmable electronic model,” *IEEE Trans. Power Syst.*, vol. 4, no. 1, pp. 286–292, 1989.
- [10] C. Dufour, V. Jalili-Marandi, J. Belanger, and L. Snider, “Power system simulation algorithms for parallel computer architectures,” in *2012 IEEE Power and Energy Society General Meeting*, 2012, pp. 1–6.
- [11] M. O. O. Faruque and V. Dinavahi, “Hardware-in-the-loop simulation of power

- electronic systems using adaptive discretization,” *IEEE Trans. Ind. Electron.*, vol. 57, no. 4, pp. 1146–1158, 2010.
- [12] M. Reiter, M. Wehr, and D. Abel, “Built-in HiL simulator: A concept for faster prototyping of navigation- and communication-based control systems,” in *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 2016, pp. 1363–1369.
- [13] E. de Jong *et al.*, “European White Book on Real-Time Power Hardware-in-the-loop testing,” 2012.
- [14] F. Katiraei, C. Sun, and B. Enayati, “No Inverter Left Behind: Protection, Controls, and Testing for High Penetrations of PV Inverters on Distribution Systems,” *IEEE Power Energy Mag.*, vol. 13, no. 2, pp. 43–49, 2015.
- [15] B. Lu, X. Wu, H. Figueroa, and A. Monti, “A Low-Cost Real-Time Hardware-in-the-Loop Testing Approach of Power Electronics Controls,” *IEEE Trans. Ind. Electron.*, vol. 54, no. 2, pp. 919–931, 2007.
- [16] W. Li, G. Joos, and J. Belanger, “Real-Time Simulation of a Wind Turbine Generator Coupled With a Battery Supercapacitor Energy Storage System,” *IEEE Trans. Ind. Electron.*, vol. 57, no. 4, pp. 1137–1145, 2010.
- [17] D. Maclay, “Simulation gets into the loop,” *IEE Rev.*, vol. 43, no. 3, pp. 109–112, 1997.
- [18] K. Ou *et al.*, “MMC-HVDC Simulation and Testing Based on Real-Time Digital Simulator and Physical Control System,” *IEEE J. Emerg. Sel. Top. Power Electron.*, vol. 2, no. 4, pp. 1109–1116, 2014.
- [19] Y. Liu *et al.*, “Controller hardware-in-the-loop validation for a 10 MVA ETO-based STATCOM for wind farm application,” in *2009 IEEE Energy Conversion Congress and Exposition*, 2009, pp. 1398–1403.
- [20] A. Sharma, S. C. Srivastava, and S. Chakrabarti, “Testing and Validation of Power System Dynamic State Estimators Using Real Time Digital Simulator (RTDS),” *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 2338–2347, 2016.
- [21] S. Vamsidhar and B. G. Fernandes, “Hardware-in-the-loop simulation based design and experimental evaluation of DTC strategies,” in *2004 IEEE 35th Annual Power Electronics Specialists Conference (IEEE Cat. No.04CH37551)*, 2004, vol. 5, pp. 3615–3621 Vol.5.
- [22] M. Rezkallah, A. Hamadi, A. Chandra, and B. Singh, “Real-Time HIL Implementation of Sliding Mode Control for Standalone System Based on PV

- Array Without Using Dumpload,” *IEEE Trans. Sustain. Energy*, vol. 6, no. 4, pp. 1389–1398, 2015.
- [23] R. A. Ferreira, “Controle de Microrredes CC baseado em droop adaptativo de tensão - simulação em tempo real com control-hardware-in-loop,” Universidade Federal de Juiz de Fora, 2015.
- [24] R. Uluski *et al.*, “Microgrid Controller Design, Implementation, and Deployment: A Journey from Conception to Implementation at the Philadelphia Navy Yard,” *IEEE Power Energy Mag.*, vol. 15, no. 4, pp. 50–62, 2017.
- [25] H. Saad, C. Dufour, J. Mahseredjian, S. Denetiere, and S. Nguéfeu, “Real time simulation of MMCs using the state-space nodal approach,” in *Proceedings of the IPST*, 2013, vol. 13, pp. 18–20.
- [26] ANEEL, “Edital de Leilão N°007/2008. Anexo 6F-CC- Lote LF-CC. Transmissão associada à integração das usinas do Rio Madeira. Conversoras do bipolo 2 - SE coletora Porto Velho e SE Araraquara 2. Características e requisitos técnicos básicos das instalações de trans.” Agência Nacional de Energia Elétrica, pp. 473–593, 2008.
- [27] N. Mohan, T. M. Undeland, and W. P. Robbins, *Power Electronics Converters, Applications, and Design*, 2nd ed. New York: John Willey & Sons, Inc, 1995.
- [28] M. Matar and R. Iravani, “FPGA Implementation of the Power Electronic Converter Model for Real-Time Simulation of Electromagnetic Transients,” *IEEE Trans. Power Deliv.*, vol. 25, no. 2, pp. 852–860, 2010.
- [29] PSCAD, “EMTDC Users Guide V4.6.0.” Manitoba Hydro International Ltd., 2017.
- [30] V. R. Dinavahi, R. Iravani, and R. Bonert, “Real-time digital simulation of power electronic apparatus interfaced with digital controllers,” *Power Deliv. IEEE Trans.*, vol. 16, no. 4, pp. 775–781, 2001.
- [31] F. Dicler *et al.*, “DSP implementation of a real-time boost converter simulator handling inter-step switching delay,” in *CBA Congresso Brasileiro de Automática*, 2018.
- [32] C. Dufour, S. Cense, T. Ould-Bachir, L. Grégoire, and J. Bélanger, “General-purpose reconfigurable low-latency electric circuit and motor drive solver on FPGA,” in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, 2012, pp. 3073–3081.
- [33] M. Lemaire, P. Sicard, and J. Belanger, “Prototyping and Testing Power

- Electronics Systems Using Controller Hardware-In-the-Loop (HIL) and Power Hardware-In-the-Loop (PHIL) Simulations,” in *2015 IEEE Vehicle Power and Propulsion Conference (VPPC)*, 2015, pp. 1–6.
- [34] R. Razzaghi, “Fast simulation of electromagnetic transients in power systems: numerical solvers and their coupling with electromagnetic time reversal process,” *École polytechnique fédérale de Lausanne*, 2016.
- [35] M. S. Vekic, S. U. Grabic, D. P. Majstorovic, I. L. Celanovic, N. L. Celanovic, and V. A. Katic, “Ultralow Latency HIL Platform for Rapid Development of Complex Power Electronics Systems,” *IEEE Trans. Power Electron.*, vol. 27, no. 11, pp. 4436–4444, Nov. 2012.
- [36] Y. Chen and V. Dinavahi, “FPGA-Based Real-Time EMTP,” *IEEE Trans. Power Deliv.*, vol. 24, no. 2, pp. 892–902, 2009.
- [37] A. Hadizadeh, M. Hashemi, M. Labbaf, and M. Parniani, “A Matrix-Inversion Technique for FPGA-Based Real-Time EMT Simulation of Power Converters,” *IEEE Trans. Ind. Electron.*, vol. 66, no. 2, pp. 1224–1234, 2019.
- [38] G. G. Parma and V. Dinavahi, “Real-Time Digital Hardware Simulation of Power Electronics and Drives,” *IEEE Trans. Power Deliv.*, vol. 22, no. 2, pp. 1235–1246, 2007.
- [39] A. Myaing and V. Dinavahi, “FPGA-Based Real-Time Emulation of Power Electronic Systems With Detailed Representation of Device Characteristics,” *IEEE Trans. Ind. Electron.*, vol. 58, no. 1, pp. 358–368, Jan. 2011.
- [40] M. M. A. Matar Bayoumi, “An FPGA-Based Real-Time Simulator for the Analysis of Electromagnetic Transients in Electrical Power Systems,” PhD thesis, University of Toronto, 2009.
- [41] M. Dagbagi, A. Hemdani, L. Idkhajine, M. W. Naouar, E. Monmasson, and I. Slama-Belkhodja, “ADC-Based Embedded Real-Time Simulator of a Power Converter Implemented in a Low-Cost FPGA: Application to a Fault-Tolerant Control of a Grid-Connected Voltage-Source Rectifier,” *IEEE Trans. Ind. Electron.*, vol. 63, no. 2, pp. 1179–1190, 2016.
- [42] Ó. Lucía, I. Urriza, L. A. Barragan, D. Navarro, Ó. Jimenez, and J. M. Burdio, “Real-Time FPGA-Based Hardware-in-the-Loop Simulation Test Bench Applied to Multiple-Output Power Converters,” *IEEE Trans. Ind. Appl.*, vol. 47, no. 2, pp. 853–860, 2011.
- [43] RTDS Technologies, “MMC Modelling,” Winnipeg, Canadá, 2015.

- [44] Typhoon HIL, “Find the right HIL size for you,” *HIL portfolio overview*, 2019. [Online]. Available: <https://www.typhoon-hil.com/products/hil-portfolio-overview>. [Accessed: 05-Aug-2019].
- [45] C. Villegas and V. Lenzi, “HIL testing and controls for power electronics made easy with Speedgoat real-time solutions,” *IEEE PES IAS Workshop*. Turgi, 2019.
- [46] T. Maguire, S. Elimban, E. Tara, and Y. Zhang, “Predicting Switch ON/OFF Statuses in Real Time Electromagnetic Transients Simulations with Voltage Source Converters,” in *2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2)*, 2018, pp. 1–7.
- [47] T. Maguire and J. Giesbrecht, “Small time-step (< 2us) VSC model for the real time digital simulator,” in *International Conference on Power Systems Transients*, 2005.
- [48] T. Ould-Bachir, H. F. Blanchette, and K. Al-Haddad, “A Network Tearing Technique for FPGA-Based Real-Time Simulation of Power Converters,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3409–3418, 2015.
- [49] J. Belanger, P. Venne, and J.-N. Paquin, “The what, where and why of real-time simulation,” *Planet Rt*, vol. 1, no. 1, pp. 25–29, 2010.
- [50] P. Pejovic and D. Maksimovic, “A method for fast time-domain simulation of networks with switches,” *IEEE Trans. Power Electron.*, vol. 9, no. 4, pp. 449–456, Jul. 1994.
- [51] K. Wang, J. Xu, G. Li, N. Tai, A. Tong, and J. Hou, “A Generalized Associated Discrete Circuit Model of Power Converters in Real-Time Simulation,” *IEEE Trans. Power Electron.*, vol. 34, no. 3, pp. 2220–2233, 2019.
- [52] C. Dufour, “Method and system for reducing power losses and state-overshoots in simulators for switched power electronic circuit,” US 2016/0314230 A1, 2016.
- [53] R. Razzaghi, C. Foti, M. Paolone, and F. Rachidi, “A novel method for the optimal parameter selection of discrete-time switch model,” in *Proceedings of the 10th International Conference on Power Systems Transients (IPST 2013)*, 2013, no. EPFL-CONF-188222, pp. 1–7.
- [54] X. Guo, J. Yuan, Y. Tang, and X. You, “Hardware in the Loop Real-Time Simulation for the Associated Discrete Circuit Modeling Optimization Method of Power Converters,” *Energies*, vol. 11, no. 11, p. 3237, 2018.
- [55] Q. Mu, J. Liang, X. Zhou, Y. Li, and X. Zhang, “Improved ADC Model of Voltage-Source Converters in DC Grids,” *IEEE Trans. Power Electron.*, vol. 29,

- no. 11, pp. 5738–5748, Nov. 2014.
- [56] P. Li, Z. Wang, C. Wang, X. Fu, H. Yu, and L. Wang, “Synchronisation mechanism and interfaces design of multi-FPGA-based real-time simulator for microgrids,” *IET Gener. Transm. Distrib.*, vol. 11, no. 12, pp. 3088–3096, 2017.
- [57] C. Dufour, W. Li, X. Xiao, J.-N. Paquin, and J. Bélanger, “Fault studies of MMC-HVDC links using FPGA and CPU on a real-time simulator with iteration capability,” in *2017 11th IEEE International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG)*, 2017, pp. 550–555.
- [58] R. Meka, M. Sloderbeck, M. O. Faruque, J. Langston, M. Steurer, and L. S. DeBrunner, “FPGA model of a high-frequency power electronic converter in an RTDS power system co-simulation,” in *2013 IEEE Electric Ship Technologies Symposium (ESTS)*, 2013, pp. 71–75.
- [59] J. von Appen, M. Braun, T. Stetz, K. Diwold, and D. Geibel, “Time in the sun: the challenge of high PV penetration in the German electric grid,” *IEEE Power Energy Mag.*, vol. 3, no. 2, pp. 55–64, 2013.
- [60] T. Stetz, F. Marten, and M. Braun, “Improved low voltage grid-integration of photovoltaic systems in Germany,” *Sustain. Energy, IEEE Trans.*, vol. 4, no. 2, pp. 534–542, 2013.
- [61] P. C. Kotsampopoulos, F. Lehfuss, G. F. Lauss, B. Bletterie, and N. D. Hatziargyriou, “The Limitations of Digital Simulation and the Advantages of PHIL Testing in Studying Distributed Generation Provision of Ancillary Services,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 9, pp. 5502–5515, 2015.
- [62] R. Brundlinger *et al.*, “Lab tests: Verifying that smart grid power converters are truly smart,” *IEEE Power Energy Mag.*, vol. 13, no. 2, pp. 30–42, 2015.
- [63] Cepel, “Cepel se prepara para ser referência em pesquisas voltadas a redes elétricas inteligentes,” *Detalhe: Notícias*, 2019. [Online]. Available: http://www.cepel.br/pt_br/sala-de-imprensa/noticias/cepel-se-prepara-para-ser-referencia-em-pesquisas-voltadas-a-redes-eletricas-inteligentes.htm. [Accessed: 08-Aug-2019].
- [64] R. Razzaghi, M. Paolone, and F. Rachidi, “A general purpose FPGA-based real-time simulator for power systems applications,” in *IEEE PES ISGT Europe 2013*, 2013, pp. 1–5.
- [65] F. R. F. de Sá, A. C. Oliveira, and W. L. A. Neves, “Simulação de transitórios

- eletromagnéticos em tempo real via FPGA,” in *CBA Congresso Brasileiro de Automática*, 2018.
- [66] E. de M. Fernandes, D. R. Huller, A. C. Oliveira, M. B. de R. Corrêa, and W. R. N. Santos, “Simulador em tempo real para motor síncrono com ímã permanente baseado em dispositivos lógicos programáveis,” *Rev. Eletrônica Potência*, vol. 20, no. 3, pp. 244–253, 2015.
- [67] J. F. Araújo, H. M. Pedrosa, P. G. Barbosa, and M. C. B. P. Rodriguez, “Simulação em tempo real ‘Hardware-in-the-loop’ de elementos do sistema de propulsão de um veículo elétrico,” in *XXI Congresso Brasileiro de Automática*, 2016.
- [68] K. Ogata, *Ingeniería de control moderna*. Pearson Educación, 2003.
- [69] MathWorks, “How Simscape electrical specialized power systems software works,” *Simulation and analysis*, 2019. [Online]. Available: <https://la.mathworks.com/help/physmod/sps/powersys/ug/how-simpowersystemssoftware-works.html>. [Accessed: 07-Aug-2019].
- [70] H. W. Dommel, “Digital computer solution of electromagnetic transients in single-and multiphase networks,” *IEEE Trans. power Appar. Syst.*, no. 4, pp. 388–399, 1969.
- [71] Chung-Wen Ho, A. Ruehli, and P. Brennan, “The modified nodal approach to network analysis,” *IEEE Trans. Circuits Syst.*, vol. 22, no. 6, pp. 504–509, 1975.
- [72] S. Y. R. Hui and S. Morrall, “Generalised associated discrete circuit model for switching devices,” *IEE Proc. - Sci. Meas. Technol.*, vol. 141, no. 1, pp. 57–64, Jan. 1994.
- [73] M. N. O. Sadiku, *Numerical techniques in electromagnetics*. CRC press, 2000.
- [74] K. J. Astrom and B. Wittenmark, *Computer-Controlled Systems, 3rd edition*, 3rd editio. Prentice Hall, Inc, 1997.
- [75] H. W. Dommel, *EMTP theory book*. Microtran Power System Analysis Corporation, 1996.
- [76] J. R. Marti and J. Lin, “Suppression of numerical oscillations in the EMTP power systems,” *IEEE Trans. Power Syst.*, vol. 4, no. 2, pp. 739–747, 1989.
- [77] Kuang Sheng, B. W. Williams, and S. J. Finney, “A review of IGBT models,” *IEEE Trans. Power Electron.*, vol. 15, no. 6, pp. 1250–1266, Nov. 2000.
- [78] A. R. Hefner Jr and D. L. Blackburn, “An analytical model for the steady-state and transient characteristics of the power insulated-gate bipolar transistor,” *Solid.*

- State. Electron.*, vol. 31, no. 10, pp. 1513–1532, 1988.
- [79] R. Kraus and K. Hoffmann, “An analytical model of IGBTs with low emitter efficiency,” in *[1993] Proceedings of the 5th International Symposium on Power Semiconductor Devices and ICs*, 1993, pp. 30–34.
- [80] M. Dagbagi, “FPGA-Based embedded real time simulation of electrical systems,” L’Ecole Nationale d’Infénierus de Tunis, 2015.
- [81] A. Yazdani and R. Iravani, *Voltage-sourced converters in power systems: modeling, control, and applications*. Wiley Online Library, 2010.
- [82] MathWorks, “Ideal Switch,” *Documentation - Power Electronics*, 2018. [Online]. Available:
<https://la.mathworks.com/help/physmod/sps/powersys/ref/idealswitch.html>.
 [Accessed: 30-Oct-2018].
- [83] S. Bacha, I. Munteanu, A. I. Bratcu, and others, “Power electronic converters modeling and control,” *Adv. texts Control Signal Process.*, vol. 454, p. 454, 2014.
- [84] J. G. Kassakian, M. F. Schlecht, and G. C. Verghese, *Principles of power electronics*. Graphis, 2000.
- [85] J. Jeon *et al.*, “Development of Hardware In-the-Loop Simulation System for Testing Operation and Control Functions of Microgrid,” *IEEE Trans. Power Electron.*, vol. 25, no. 12, pp. 2919–2929, 2010.
- [86] I. Etxeberria-Otadui, V. Manzo, S. Bacha, and F. Baltés, “Generalized average modelling of FACTS for real time simulation in ARENE,” in *IEEE 2002 28th Annual Conference of the Industrial Electronics Society. IECON 02*, 2002, vol. 2, pp. 864–869 vol.2.
- [87] M. Chiarandini, “LU Factorization,” *Linear and Integer Programming*. [Online]. Available: <https://imada.sdu.dk/~marco/Teaching/AY2014-2015/DM554/Slides/dm554-lu.pdf>. [Accessed: 09-Sep-2019].
- [88] B. MacCleery, O. Trescases, M. Mujagic, D. M. Bohls, O. Stepanov, and G. Fick, “A new platform and methodology for system-level design of next-generation FPGA-based digital SMPS,” in *2012 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2012, pp. 1599–1606.
- [89] NI, “O que é o PXI?,” *Suporte/ Tutoriais*, 2014. [Online]. Available: <http://www.ni.com/tutorial/4811/pt/>. [Accessed: 30-Nov-2018].
- [90] W. Ji, K. Wang, G. Li, J. Zhuang, S. Li, and J. Li, “A real-time simulation algorithm for power electronics circuit considering multiple switching events and

- its application on PXI platform,” in *2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC)*, 2016, pp. 1–6.
- [91] Y. Liao, X. Shi, C. Fu, and J. Meng, “Hardware in-the-loop simulation system based on NI-PXI for operation and control of microgrid,” in *2014 9th IEEE Conference on Industrial Electronics and Applications*, 2014, pp. 1366–1370.
- [92] Z. Zhang, R. Schurhuber, L. Fickert, X. Liu, Q. Chen, and Y. Zhang, “Hardware-in-the-loop Based Grid Compatibility Test for Power Electronics Interface,” in *2019 20th International Scientific Conference on Electric Power Engineering (EPE)*, 2019, pp. 1–6.
- [93] OPAL-RT, “e-FPGASIM - Pragmatic Real-Time Simulation on FPGA for Modern Electronic Systems,” *OPAL-RT Technologies*, 2017. [Online]. Available: <https://www.opal-rt.com/systems-efpgasim/>. [Accessed: 14-Sep-2017].
- [94] W. Kester and A. D. I. Engineeri, *Data conversion handbook*. Newnes, 2005.
- [95] V. Blasko, V. Kaura, and W. Niewiadomski, “Sampling of discontinuous voltage and current signals in electrical drives: A system approach,” in *IAS’97. Conference Record of the 1997 IEEE Industry Applications Conference Thirty-Second IAS Annual Meeting*, 1997, vol. 1, pp. 682–689.
- [96] S. Buso, P. Mattavelli, L. Rossetto, and G. Spiazzi, “Simple digital control improving dynamic performance of power factor preregulators,” *IEEE Trans. Power Electron.*, vol. 13, no. 5, pp. 814–823, 1998.
- [97] National Instruments, “Transferring data using direct memory access (FPGA module),” *LabVIEW 2016 FPGA Module Help*, 2016. [Online]. Available: http://zone.ni.com/reference/en-XX/help/371599L-01/lvfpgaconcepts/fpga_dma_communication/. [Accessed: 20-Aug-2019].
- [98] S. Buso and P. Mattavelli, “Digital control in power electronics,” *Lect. Power Electron.*, vol. 1, no. 1, pp. 1–158, 2006.
- [99] J. M. Guerrero, J. C. Vasquez, J. Matas, L. G. de Vicuna, and M. Castilla, “Hierarchical Control of Droop-Controlled AC and DC Microgrids- A General Approach Toward Standardization,” *IEEE Trans. Ind. Electron.*, vol. 58, no. 1, pp. 158–172, Jan. 2011.
- [100] M. Schiesser, S. Wasterlain, M. Marchesoni, and M. Carpita, “A simplified design strategy for multi-resonant current control of a grid-connected voltage source inverter with an LCL filter,” *energies*, vol. 11, no. 3, p. 609, 2018.
- [101] MathWorks, “Generate C and C++ code optimized for embedded systems,”

- Embedded Coder*, 2019. [Online]. Available: <https://la.mathworks.com/products/embedded-coder.html>. [Accessed: 30-Apr-2019].
- [102] Texas Instruments, “LAUNCHXL-F28377S overview,” *SPRUI25D User’s Guide*, 2019. [Online]. Available: <http://www.ti.com/lit/ug/sprui25d/sprui25d.pdf>. [Accessed: 03-May-2019].
- [103] S. Deshmukh, B. Natarajan, and A. Pahwa, “Voltage/VAR control in distribution networks via reactive power injection through distributed generators,” *Smart Grid, IEEE Trans.*, vol. 3, no. 3, pp. 1226–1234, 2012.
- [104] L. Harnefors, X. Wang, A. G. Yepes, and F. Blaabjerg, “Passivity-based stability assessment of grid-connected VSCs - An overview,” *IEEE J. Emerg. Sel. Top. Power Electron.*, vol. 4, no. 1, pp. 116–125, 2015.
- [105] ANEEL, “Módulo 8 - Qualidade de Energia Elétrica. Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional -PRODIST.” Agência Nacional de Energia Elétrica, 2018.
- [106] K. Niyomsatian, P. Vanassche, R. V. Sabariego, and J. Gyselinck, “A comparison of virtual circuit-based control designs for half-bridge converters with LCL output filters,” in *2017 19th European Conference on Power Electronics and Applications (EPE’17 ECCE Europe)*, 2017, p. P--1.
- [107] H. Hooshyar, L. Vanfretti, and C. Dufour, “Delay-free parallelization for real-time simulation of a large active distribution grid model,” in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, 2016, pp. 6278–6284.
- [108] C. Dufour, J. Mahseredjian, J. Bélanger, and J. L. Naredo, “An advanced real-time electro-magnetic simulator for power systems with a simultaneous state-space nodal solver,” in *Transmission and Distribution Conference and Exposition: Latin America (T&D-LA), 2010 IEEE/PES*, 2010, pp. 349–358.
- [109] M. Dyck and O. Nzimako, “Real-time simulation of large distribution networks with distributed energy resources,” *CIREN - Open Access Proc. J.*, vol. 2017, no. 1, pp. 1402–1405, 2017.
- [110] P. Palensky, A. A. Van Der Meer, C. D. Lopez, A. Joseph, and K. Pan, “Cosimulation of Intelligent Power Systems: Fundamentals, Software Architecture, Numerics, and Coupling,” *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 34–50, 2017.

- [111] F. Jia, X. Cai, Y. Lou, and Z. Li, “Interfacing technique and hardware-in-loop simulation of real-time co-simulation platform for wind energy conversion system,” *IET Gener. Transm. Distrib.*, vol. 11, no. 12, pp. 3030–3038, 2017.
- [112] S. Oh and S. Chae, “A co-simulation framework for power system analysis,” *Energies*, vol. 9, no. 3, p. 131, 2016.
- [113] D. Shu, X. Xie, Q. Jiang, G. Guo, and K. Wang, “A Multirate EMT Co-Simulation of Large AC and MMC-Based MTDC Systems,” *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 1252–1263, 2018.
- [114] R. Kuffel *et al.*, “Expanding an Analogue HVDC Simulator’s Modelling Capability Using a Real-Time Digital Simulator (RTDS),” in *ICDS '95. First International Conference on Digital Power System Simulators*, 1995, pp. 199-.
- [115] F. A. Moreira, J. R. Mart, L. C. Zanetta, Jr., and L. R. Linares, “Multirate Simulations With Simultaneous-Solution Using Direct Integration Methods in a Partitioned Network Environment,” *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 53, no. 12, pp. 2765–2778, 2006.
- [116] T. Maguire, I. K. Park, and P. Forsyth, “An interface technique for multi-rate EMT simulations,” in *International Conference on Power Systems Transients 2015 (IPST 2015)*, 2015.
- [117] NERC, “Short-Circuit Modeling and System Strength,” Atlanta, 2018.
- [118] ABNT, “ABNT NBR 16149 - Sistemas fotovoltaicos (FV) - Características da interface de conexão com a rede elétrica de distribuição.” Associação Brasileira de Normas Técnicas, 2013.
- [119] IEEE PES AMPS DSAS Test Feeder Working Group, “1992 Test Feeder Cases,” *Resources*, 2017. [Online]. Available: <http://sites.ieee.org/pes-testfeeders/resources/>. [Accessed: 06-Sep-2019].
- [120] O. Tremblay, H. Fortin-Blanchette, R. Gagnon, and Y. Brissette, “Contribution to stability analysis of power hardware-in-the-loop simulators,” *IET Gener. Transm. Distrib.*, vol. 11, no. 12, pp. 3073–3079, 2017.
- [121] L. G. B. Rolim, D. R. da Costa, and M. Aredes, “Analysis and software implementation of a robust synchronizing PLL circuit based on the pq theory,” *IEEE Trans. Ind. Electron.*, vol. 53, no. 6, pp. 1919–1926, 2006.
- [122] H. Akagi, E. H. Watanabe, and M. Aredes, *Instantaneous Power Theory and Applications to Power Conditioning*. 2007.
- [123] M. H. Rashid, *Power electronics handbook*. Butterworth-Heinemann, 2017.

- [124] R. Teodorescu, M. Liserre, and P. Rodriguez, *Grid converters for photovoltaic and wind power systems*, vol. 29. John Wiley & Sons, 2011.
- [125] W. Leonhard, *Control of Electrical Drives*, 3°. Springer-Verlag, 2001.
- [126] A. R. Ofoli and M. R. Altimania, “Real-time digital simulator testbed using eMEGASim for wind power plants,” in *2017 IEEE Industry Applications Society Annual Meeting*, 2017, pp. 1–9.

Apêndices

A. Análise numérica das perdas de potência virtuais no circuito discreto associado de um indutor

A Figura A.1 mostra um circuito contínuo simplificado, no qual uma fonte de tensão está conectada a um indutor puro, e sua representação discreta associada. Neste apêndice são avaliados os fluxos de potência ativa no circuito discreto, considerando que a condutância G_L é um elemento consumidor de potência. Dois métodos de integração numérica são utilizados nesta análise: Euler regressivo e Trapezoidal.

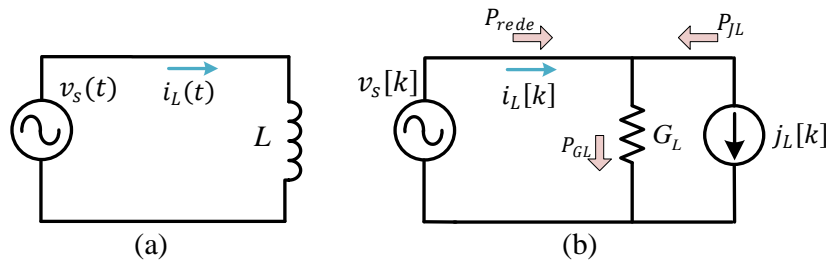


Figura A.1 – Circuito simplificado considerado na análise de perdas virtuais: (a) circuito contínuo, (b) circuito discreto associado.

No circuito acima, a fonte de tensão é definida pela expressão:

$$v_s(t) = \sqrt{2} \cdot \cos(120 \cdot \pi \cdot t), \quad (\text{A.1})$$

o que implica que a expressão analítica para a corrente $i_L(t)$, considerando condições iniciais nulas, vem dada por:

$$i_{L-ana}(t) = \frac{\sqrt{2}}{L \cdot 120 \cdot \pi} \cdot \sin(120 \cdot \pi \cdot t). \quad (\text{A.2})$$

Na análise a seguir é realizada uma varredura nos demais parâmetros do circuito da Figura A.1 (indutância e passo de simulação), de forma a avaliar numericamente os fluxos de potência ativa no circuito discreto associado (P_{rede}, P_{GL}, P_{JL}). Estas potências foram determinadas como o valor médio da potência instantânea, considerado em todos os casos um período de 50 ms. G_L e $j_L[k]$ são determinados, para cada método de integração, conforme especificado pela Tabela 2.1.

Método Euler regressivo

A Figura A.2 mostra os fluxos de potência ativa, definidos na Figura A.1(b), em função do passo de solução com a reatância indutiva constante em 1 pu. Observa-se que a potência P_{GL} tem um comportamento linear com pente positiva. Um aspecto interessante do resultado apresentado é que metade desta potência é fornecida pela rede enquanto que a outra metade é fornecida pela fonte de corrente em paralelo do circuito discreto associado (P_{JL}). Assim, as perdas de potência virtuais equivalentes da representação discreta pelo método de Euler regressivo são iguais à metade da potência consumida por G_L .

Na figura é apresentada, adicionalmente, uma quarta variável (P_{erro}), que é determinada a partir de:

$$\sum_{k=1}^T (i_L[k] - i_{L-ana}(k)) \cdot v_s[k] / T. \quad (A.3)$$

P_{erro} corresponde à potência ativa originada pela inexactidão da resposta discreta, em contraste com a potência analítica. Que, conforme observado, coincide com as perdas de potência virtuais P_{rede} . Por outras palavras, o erro de exatidão do método de integração Euler regressivo pode ser representado em termos das perdas de potência virtuais resultantes.

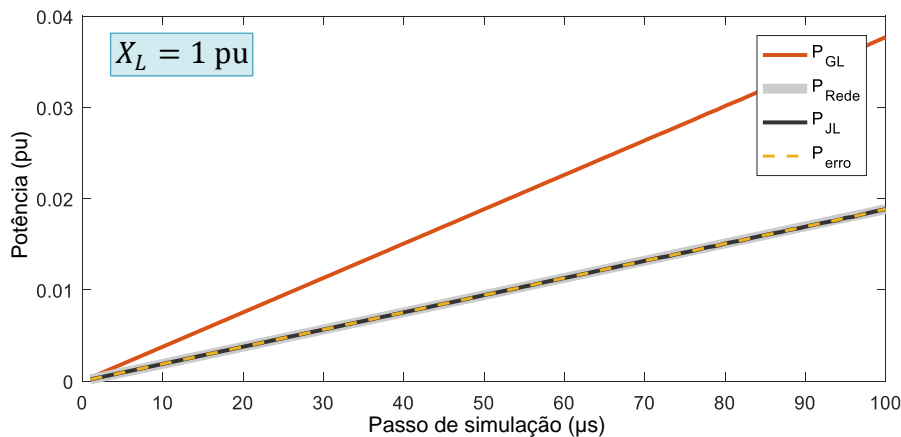


Figura A.2 – Fluxos de potência ativa em função do passo de solução para o método de integração Euler regressivo.

A Figura A.3 apresenta as mesmas variáveis, porém em função do valor da indutância. As formas de onda apresentadas desta vez são não-lineares, porém mantêm a mesma relação discutida anteriormente, na qual metade da potência consumida pela condutância é fornecida pela rede, enquanto que a outra metade pela fonte de corrente do circuito discreto associado do indutor.

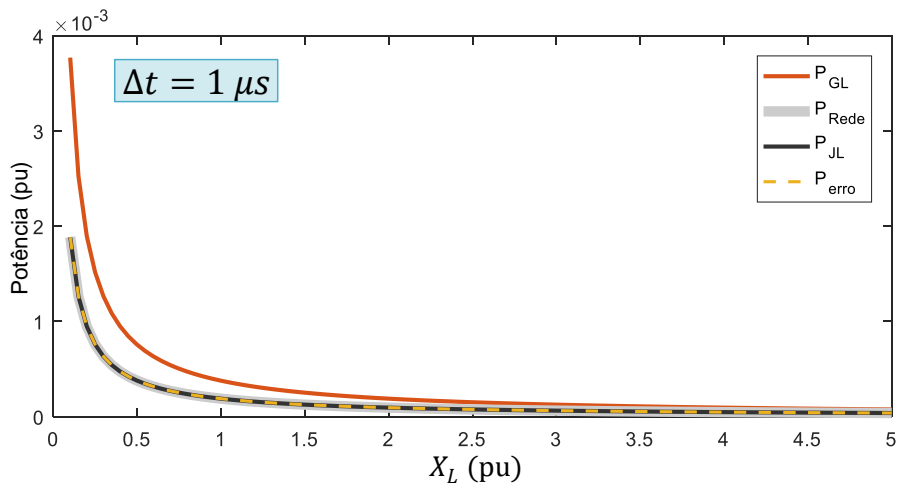


Figura A.3 – Fluxos de potência ativa em função da reatância indutiva para o método de integração Euler regressivo.

Os resultados apresentados coincidem com a análise apresentada na Seção 2.1.2.1, na qual se destaca a característica amortecedora do método Euler regressivo, que, em contrapartida, está associada a perdas de potência virtuais.

Método Trapezoidal

As Figura A.4 e Figura A.5 reproduzem as análises anteriores para o caso do método de integração Trapezoidal. Observa-se que as formas de onda de P_{GL} permanecem com os mesmos formatos obtidos no caso anterior, porém com valores absolutos menores, uma vez que o método Trapezoidal é mais exato do que o método Euler regressivo. A principal característica destes resultados é que toda a potência requerida por G_L é fornecida pela fonte de corrente $j_L[k]$, o que se traduz em perdas de potência virtuais nulas com este método, como mostram as formas de onda P_{rede} e P_{erro} .

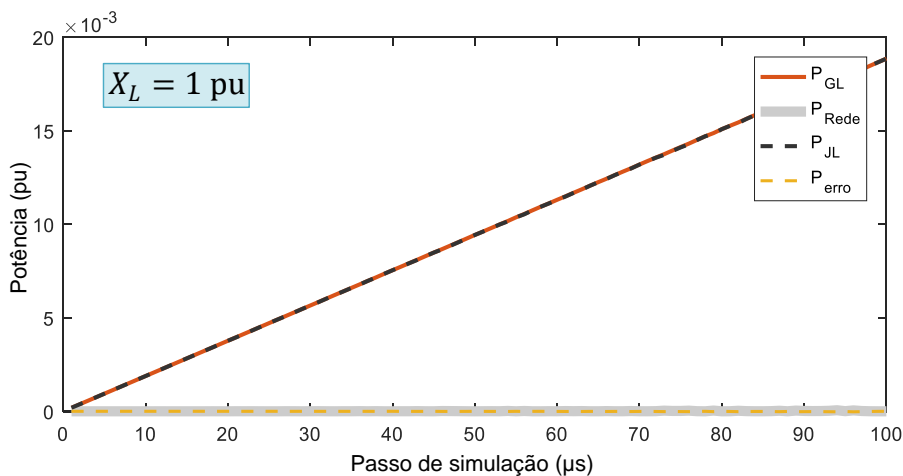


Figura A.4 – Fluxos de potência ativa em função do passo de solução para o método de integração Trapezoidal.

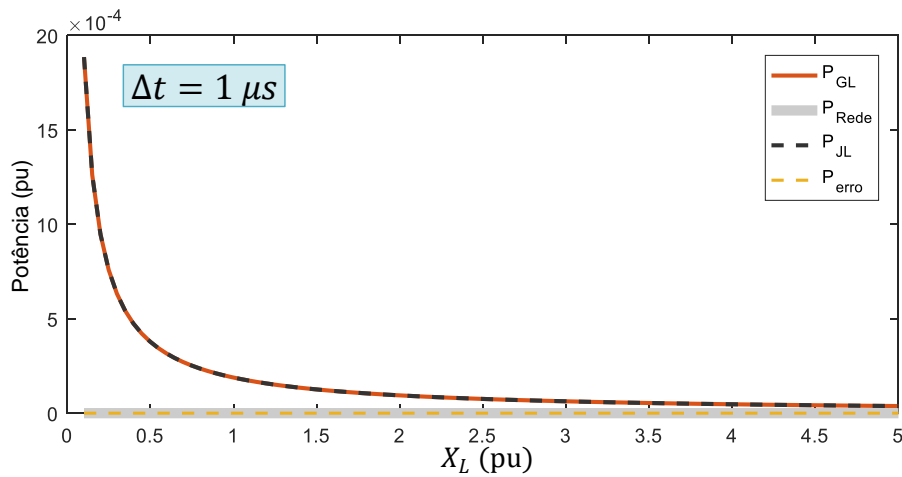


Figura A.5 – Fluxos de potência ativa em função da reatância indutiva para o método de integração Trapezoidal.

Em [75] é realizada uma análise do erro de exatidão do método Trapezoidal para simulação de sistemas elétricos, no qual esse erro é interpretado fisicamente como uma capacitância parasita em paralelo com o indutor. Por outras palavras, embora este método não origine perdas de potência ativa, ocasiona modificações na potência reativa resultante, em comparação com o resultado analítico.

B. Determinação das multiplicações matriciais para solução do sistema

Neste apêndice são detalhadas as matrizes e vetores das multiplicações matriciais empregadas para solucionar, no tempo discreto, os sistemas representados no SDTR/FPGA. Esses componentes resultam da aplicação das leis de Kirchhoff no circuito elétrico associado da rede modelada, uma vez utilizado o algoritmo de Dommel para discretização dos elementos armazenadores de energia.

Em resumo, os sistemas elétricos simulados no SDTR/FPGA consistem em dois circuitos: i) VSC conectado a carga resistiva (Figura 2.10) e ii) VSC conectado a uma fonte equivalente (Figura 3.11). Ambos circuitos podem ser representados pelo circuito discreto associado da Figura A.6, para o qual deve ser considerado que os valores da condutância equivalente G_{eq} e da fonte j_{eq-x} (com $x = a, b, c$) diferem em função da existência ou não da carga resistiva; e que o valor das fontes equivalentes é zero no caso da carga resistiva ($v_{sa}[k] = v_{sb}[k] = v_{sc}[k] = 0$).

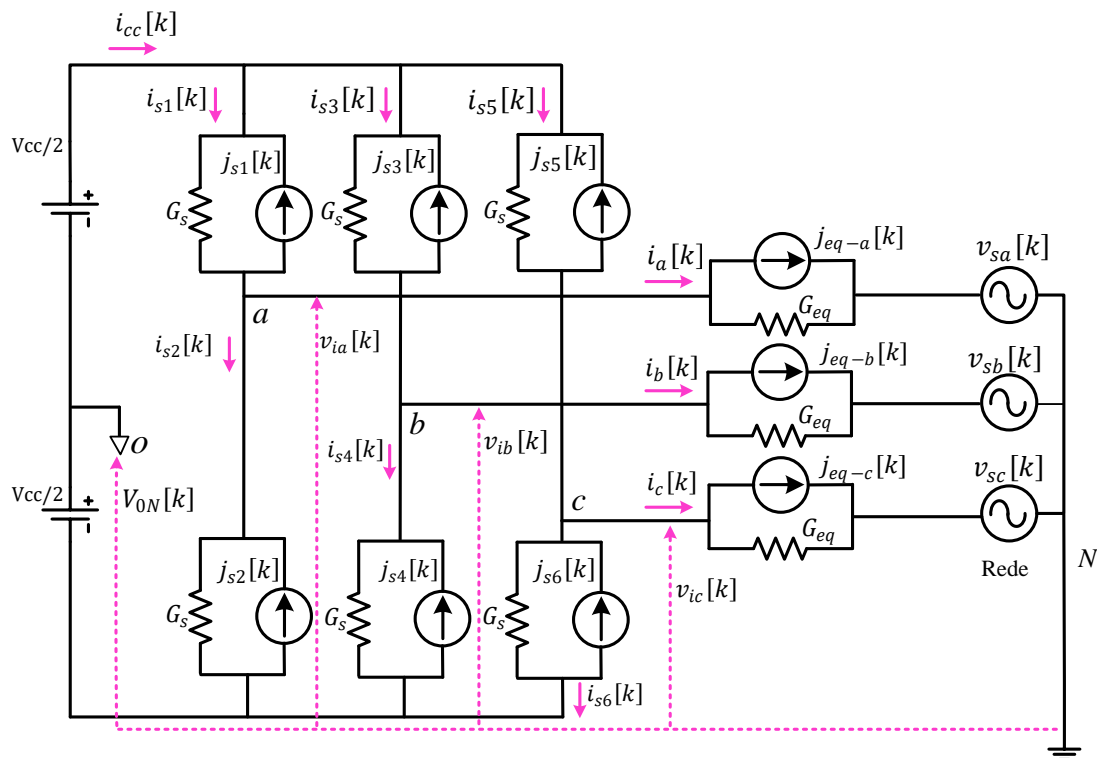


Figura A.6 – Circuito discreto associado considerado nas simulações efetuadas.

Conforme discussão do Capítulo 2, a equação da solução do sistema pelo método de Análise Nodal Modificada é:

$$\mathbf{H} \mathbf{x}[k] = \mathbf{b}[k]. \quad (\text{A.4})$$

Onde, para o caso do sistema do conversor VSC conectado a carga resistiva (utilizado nas simulações com controle em laço aberto), os componentes desta equação são:

$$\mathbf{H} = \begin{bmatrix} -1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & -G_s & 0 & 0 & 0 & 0 & 0 & G_s \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -G_s & 0 & 0 & 0 & 0 & G_s \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -G_s & 0 & 0 & 0 & G_s \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & G_s & 0 & 0 & 0 & 0 & 0 & -G_s \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & G_s & 0 & 0 & 0 & 0 & -G_s \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & G_s & 0 & 0 & 0 & -G_s \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & G_{eq1} & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & G_{eq1} & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & G_{eq1} & 0 & 0 & -1 & 0 \\ 1 & 0 & -1 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix};$$

$$\mathbf{x}[k] = \begin{bmatrix} i_{cc}[k] \\ i_{s1}[k] \\ i_{s2}[k] \\ i_{s3}[k] \\ i_{s4}[k] \\ i_{s5}[k] \\ i_{s6}[k] \\ v_{ia}[k] \\ v_{ib}[k] \\ v_{ic}[k] \\ i_a[k] \\ i_b[k] \\ i_c[k] \\ V_{0N}[k] \end{bmatrix} \quad \text{e} \quad \mathbf{b}[k] = \begin{bmatrix} 0 \\ -V_{cc}/2 \cdot G_s + j_{s1}[k] \\ -V_{cc}/2 \cdot G_s + j_{s3}[k] \\ -V_{cc}/2 \cdot G_s + j_{s5}[k] \\ 0 \\ 0 \\ 0 \\ -V_{cc}/2 \cdot G_s + j_{s2}[k] \\ -V_{cc}/2 \cdot G_s + j_{s4}[k] \\ -V_{cc}/2 \cdot G_s + j_{s6}[k] \\ -j_{eq1-a} \\ -j_{eq1-b} \\ -j_{eq1-c} \\ 0 \end{bmatrix}.$$

Com:

$$j_{eq1-abc}[k] = \frac{i_{abc}[k-1]}{(R_f + R_c) \cdot G_{L_f} + 1} \quad ; \quad G_{eq1} = \frac{G_{L_f}}{G_{L_f} \cdot (R_f + R_c) + 1};$$

e $j_{s1-6}[k]$ calculado conforme discussão da Seção 3.2, segundo o tipo de modelagem de chave empregada.

Para o caso do sistema do conversor VSC conectado à rede (utilizado nas simulações com controle em laço fechado), os componentes de (A.4) são:

$$\mathbf{H} = \begin{bmatrix} -1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & -G_s & 0 & 0 & 0 & 0 & 0 & G_s \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -G_s & 0 & 0 & 0 & 0 & G_s \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -G_s & 0 & 0 & 0 & G_s \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & G_s & 0 & 0 & 0 & 0 & 0 & -G_s \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & G_s & 0 & 0 & 0 & 0 & -G_s \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & G_s & 0 & 0 & 0 & -G_s \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & G_{eq2} & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & G_{eq2} & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & G_{eq2} & 0 & 0 & -1 & 0 \\ 1 & 0 & -1 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix};$$

$$\mathbf{x}[k] = \begin{bmatrix} i_{cc}[k] \\ i_{s1}[k] \\ i_{s2}[k] \\ i_{s3}[k] \\ i_{s4}[k] \\ i_{s5}[k] \\ i_{s6}[k] \\ v_{ia}[k] \\ v_{ib}[k] \\ v_{ic}[k] \\ i_a[k] \\ i_b[k] \\ i_c[k] \\ V_{ON}[k] \end{bmatrix} \text{ e } \mathbf{b}[k] = \begin{bmatrix} 0 \\ -V_{cc}/2 \cdot G_s + j_{s1}[k] \\ -V_{cc}/2 \cdot G_s + j_{s3}[k] \\ -V_{cc}/2 \cdot G_s + j_{s5}[k] \\ 0 \\ 0 \\ 0 \\ -V_{cc}/2 \cdot G_s + j_{s2}[k] \\ -V_{cc}/2 \cdot G_s + j_{s4}[k] \\ -V_{cc}/2 \cdot G_s + j_{s6}[k] \\ -j_{eq2-a}[k] + v_{sa}[k] \cdot G_{eq2} \\ -j_{eq2-b}[k] + v_{sb}[k] \cdot G_{eq2} \\ -j_{eq2-c}[k] + v_{sc}[k] \cdot G_{eq2} \\ 0 \end{bmatrix}.$$

Com:

$$j_{eq2-abc}[k] = \frac{i_{abc}[k-1]}{R_f \cdot G_{Lf} + 1}; \quad G_{eq2} = \frac{G_{Lf}}{G_{Lf} \cdot R + 1};$$

e j_{s1-6} calculado conforme discussão da Seção 3.2, segundo o tipo de modelagem de chave empregada.

C. Descrição dos blocos de controle em malha fechada

Neste apêndice é realizada uma sucinta descrição dos blocos de controle em malha fechada do diagrama esquemático da Figura 3.12.

PLL (*Phase-Locked Loop*)

O PLL é utilizado para sincronizar a operação do conversor com a fase angular da tensão da rede elétrica no ponto de conexão comum (PCC). Dentre as diversas configurações existentes, no presente trabalho foi empregada a estrutura descrita pela Figura A.7 [121].

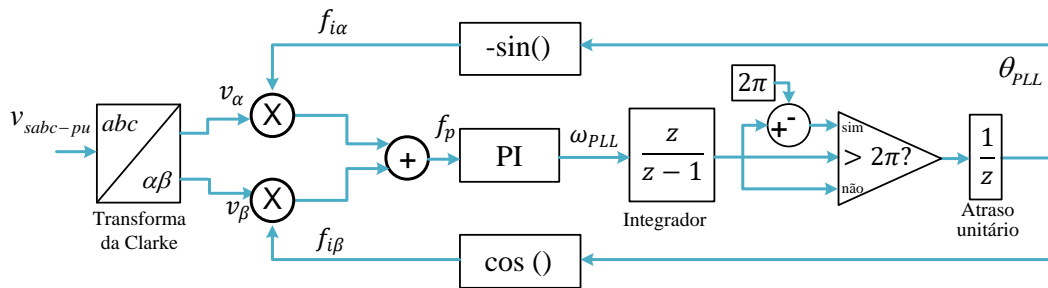


Figura A.7 – Diagrama de blocos do PLL utilizado neste trabalho.

Neste PLL as tensões da rede elétrica são representadas mediante seus componentes α e β , utilizando a transformada de Clarke [122]. O erro de fase entre estas tensões e as saídas do oscilador controlado ($f_{i\alpha}$ e $f_{i\beta}$ na figura), f_p , é calculado como a soma dos produtos dos componentes individuais, ou seja:

$$f_p = v_\alpha \cdot f_{i\alpha} + v_\beta \cdot f_{i\beta}. \quad (\text{A.5})$$

Considerando que o PLL apresentado utiliza como entrada as tensões no PCC em pu, f_p pode ser expresso como:

$$f_p = \cos(\theta_{in} - \theta_{PLL}), \quad (\text{A.6})$$

onde θ_{in} é o ângulo do vetor espacial das tensões da rede e θ_{PLL} o ângulo de saída do PLL. Dado que f_p é a entrada para um controlador Proporcional Integral – PI, seu valor em estado estável tende a zero, o que significa que, em regime permanente, o ângulo θ_{in} vai estar adiantado 90° do ângulo θ_{PLL} . Na parte final do diagrama da Figura A.7 é mostrado um integrador, discretizado pelo método de integração numérica Euler regressivo, que transforma a frequência angular de saída do controlador PI no ângulo θ_{PLL} . Existe ainda uma lógica de *reset*, para evitar *overflow*. O atraso unitário mostrado

no diagrama é necessário para quebrar o laço algébrico gerado pelo integrador. A expressão resultante $(1/(z - 1))$ equivale a um integrador discretizado pelo método Euler progressivo.

O ajuste dos ganhos do controlador PI segue a metodologia apresentada Rolim et al. [121], que se baseia na análise do comportamento linearizado do PLL, descrito pelo diagrama simplificado da Figura A.8. Assim, mesmo se tratando de um PLL discreto, o ajuste dos ganhos é realizado através de uma análise em tempo contínuo. Na Figura A.8, ω_{in} é denominada frequência central da saída do controlador PI, que em regime permanente deve ser igual à frequência das tensões no PCC, ou seja, $(\omega_{PLL} = \omega_{in})$.

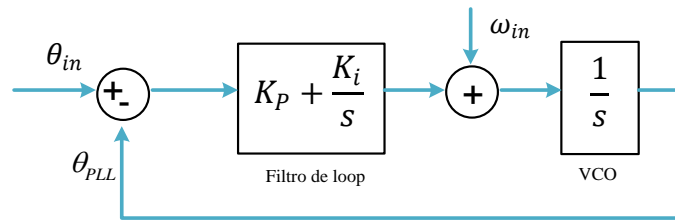


Figura A.8 – Diagrama do comportamento linearizado para pequenos sinais do PLL utilizado.

A função de transferência resultante para a representação de pequenos sinais do diagrama anterior vem dada por:

$$H(s) = \frac{\theta_{in}(s)}{\theta_{PLL}(s)} = \frac{K_P \cdot s + K_I}{s^2 + K_P \cdot s + K_I}, \quad (A.7)$$

que pode ser reescrita da forma:

$$H(s) = \frac{2 \cdot \xi \cdot \omega_n \cdot s + \omega_n^2}{s^2 + 2 \cdot \xi \cdot \omega_n \cdot s + \omega_n^2}, \quad (A.8)$$

onde $\omega_n = \sqrt{K_I}$ e $\xi = K_P/2 \cdot \sqrt{K_I}$. Neste trabalho foram escolhidos os ganhos $K_P = 70$ [(rad/s)/rad] e $K_I = 2500$ [(rad/s)/(rad·s)] de forma a ter uma frequência de corte (ω_n) de 50 rad/s e um coeficiente de amortecimento (ξ) de 0,7.

Para verificar o desempenho dinâmico deste PLL foi realizada uma simulação *offline* no *software* Simulink®. Nessa simulação é aplicado um deslocamento instantâneo de fase de 180° em um conjunto de tensões trifásicas equilibradas. Na Figura A.9(a) é mostrada a frequência rastreada pelo PLL (ω_{PLL}) durante tal evento de deslocamento de fase (que acontece no instante $t=0$ s). Observa-se que logo após o evento a frequência rastreada sai do seu valor nominal de 120π rad/s para descrever um transitório amortecido com duração aproximada de 200 ms, período a partir do qual ω_{PLL} volta a valores em torno da frequência do sistema. A Figura A.9(b) mostra o comportamento do

ângulo de saída do PLL (θ_{PLL}) durante o evento. Como desejado, em regime permanente, este sinal se encontra atrasado $\pi/2$ rad do cruze por zero da onda v_{sa} .

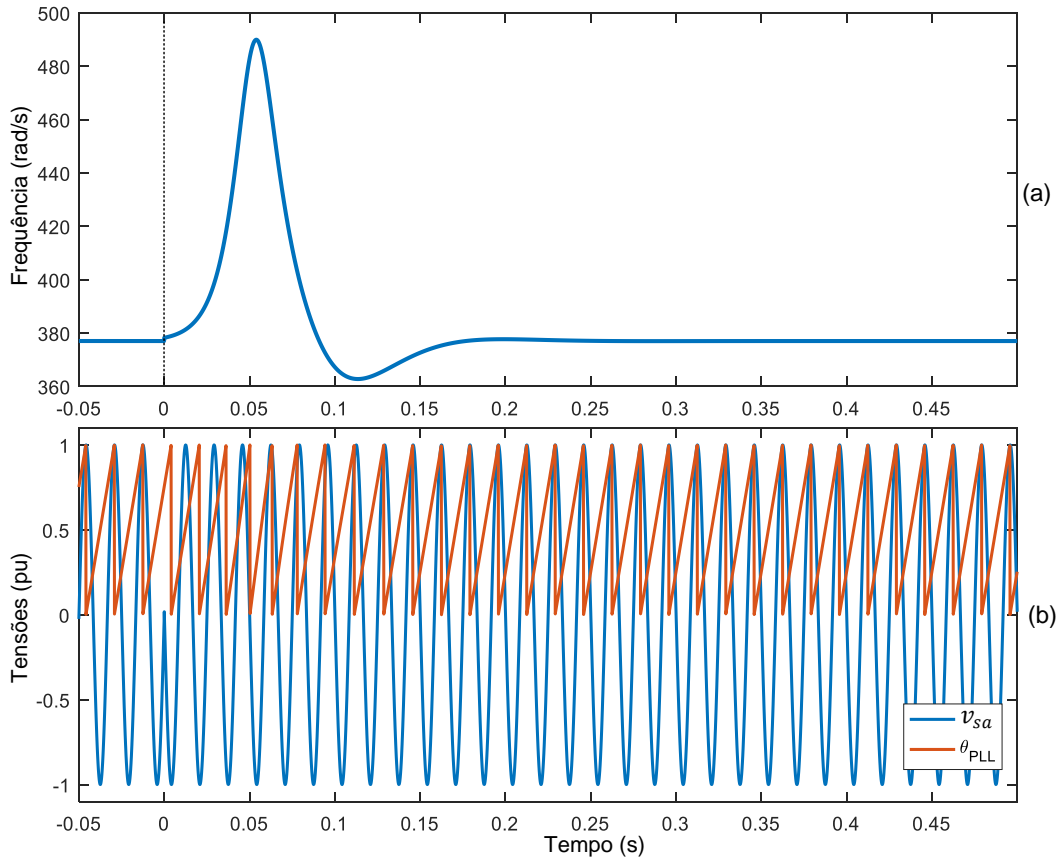


Figura A.9 – Comportamento do PLL projetado diante de um deslocamento de fase de 180° , resultados de simulação *off-line*: (a) frequência rastreada; (b) ângulo θ_{PLL} e tensão na fase a.

Mudança de referencial

Neste trabalho, o controle do conversor é realizado utilizando o referencial síncrono girante dq . A transformação das grandezas elétricas (tensões e correntes) do referencial abc para dq é feita com a transformada de Park que utiliza como entrada o ângulo θ_{PLL} . Como principal vantagem da escolha deste referencial para o desenvolvimento do sistema de controle está que as grandezas em estado estacionário se caracterizam por apresentar apenas parcelas contínuas, quando apenas componentes fundamentais de sequência positiva existirem nos sinais. Isto favorece o desempenho dos controladores PI existentes no laço interno de controle de corrente, os quais podem atingir o erro nulo em regime permanente [123]. Adicionalmente, considerando as expressões para as potências ativa e reativa de (A.9) [124]:

$$\begin{aligned} P &= v_{sd} \cdot i_d + v_{sq} \cdot i_q, \\ Q &= v_{sq} \cdot i_d - v_{sd} \cdot i_q, \end{aligned} \tag{A.9}$$

onde v_{sd} e v_{sq} são os componentes dq da tensão no PCC, e i_d e i_q são os componentes dq da corrente do conversor, é observado que ao alinhar o eixo d do referencial dq com o vetor espacial das tensões na rede (o que se traduz em $v_{sq} = 0$), a corrente i_d passa a estar diretamente relacionada com a potência ativa, enquanto que a corrente i_q com a potência reativa. Desta forma é possível desacoplar o controle dos dois componentes da corrente do inversor.

Determinação das correntes de referência

Para comandar as potências nos terminais CA do conversor VSC as referências de potência (P_{ref} e Q_{ref} , que são entradas para o controlador) são transformadas em referências de corrente (i_{dref} e i_{qref}) utilizando:

$$\begin{bmatrix} i_{dref} \\ i_{qref} \end{bmatrix} = \frac{1}{v_{sd}^2 + v_{sq}^2} \cdot \begin{bmatrix} v_{sd} & v_{sq} \\ v_{sq} & -v_{sd} \end{bmatrix} \times \begin{bmatrix} P_{ref} \\ Q_{ref} \end{bmatrix}. \quad (A.10)$$

As saídas de (A.10) servem como entrada para o laço interno de controle de corrente.

Controle interno de corrente

O diagrama esquemático do controlador interno de corrente utilizado neste trabalho é mostrado na Figura A.10. Este consiste em um par de controladores em laço fechado para os quais o erro de corrente em cada eixo de referência (d e q) serve como entrada para um controlador linear Proporcional Integral (PI). Conforme detalhado a seguir, um conjunto de sinais *feed-forward* é acrescentado à saída do controlador PI, que tem como saída as referências de tensão para o chaveamento PWM nas coordenadas dq .

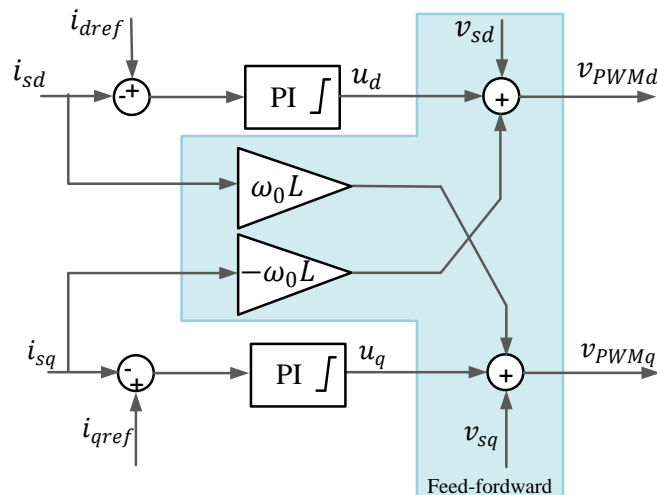


Figura A.10 – Diagrama esquemático do controlador de corrente.

Para explicar o funcionamento do controlador de corrente torna-se apropriado partir do diagrama unifilar da planta a ser controlada, ou seja, o sistema elétrico da Figura 3.11. As correntes do lado CA deste sistema podem ser representadas pelas equações que seguem:

$$\begin{aligned} L_f \frac{d}{dt} i_a &= -R_f i_a + v_{ia} - v_{sa}; \\ L_f \frac{d}{dt} i_b &= -R_f i_b + v_{ib} - v_{sb}; \\ L_f \frac{d}{dt} i_c &= -R_f i_c + v_{ic} - v_{sc}. \end{aligned} \quad (\text{A.11})$$

As quais podem ser agrupadas, utilizando a representação por vetor espacial [125], em:

$$L_f \frac{d}{dt} \vec{i} = -R_f \vec{i} + \vec{v}_i - \vec{v}_s. \quad (\text{A.12})$$

Conforme detalhado em [81], esta equação pode ser escrita nas coordenadas dq assumindo operação em regime permanente do PLL, ou seja, referencial girante em sincronismo com a frequência da rede ($\omega_{PLL} = \omega_{in}$), dando como resultado:

$$\begin{aligned} L_f \frac{d}{dt} i_d &= \omega_{in} L_f i_q - R_f i_d + v_{id} - v_{sd}; \\ L_f \frac{d}{dt} i_q &= -\omega_{in} L_f i_d - R_f i_q + v_{iq} - v_{sq}. \end{aligned} \quad (\text{A.13})$$

Em (A.13) observa-se que os termos $\omega_{in} L_f$ acoplam dinamicamente as correntes dos eixos d e q . Neste contexto, as variáveis somadas na saída do controlador PI (bloco *feed-forward*) no esquema da Figura A.10 têm justamente a função de desacoplar i_d e i_q . Considerando os termos de *feed-forward*, as saídas do controlador de corrente são:

$$\begin{aligned} v_{PWMd} &= u_d - \omega_{in} L_f i_q + v_{sd}; \\ v_{PWMq} &= u_q + \omega_{in} L_f i_d + v_{sq}, \end{aligned} \quad (\text{A.14})$$

onde u_d e u_q são as saídas dos controladores PI nos eixos de referência d e q .

Sob a consideração de que o chaveamento PWM opera de forma ideal, de tal forma que as tensões sintetizadas pelo conversor (v_{id} e v_{iq}) são iguais às referências dadas pelo controlador de corrente (v_{PWMd} e v_{PWMq}), a substituição de (A.14) em (A.13) dá como resultado:

$$L_f \frac{d}{dt} i_d = -R_f i_d + u_d; \quad (\text{A.15})$$

$$L_f \frac{d}{dt} i_q = -R_f i_q + u_q.$$

A equação (A.15) descreve dois sistemas lineares desacoplados de primeira ordem, nos quais i_d e i_q podem ser controlados de forma independente por u_d e u_q , respectivamente. Assim, o sistema em malha fechada resultante é o mostrado na Figura A.11. No qual pode ser observado que as representações equivalentes das plantas em ambos os eixos são idênticas, portanto, o mesmo controlador PI pode ser utilizado para os dois casos.

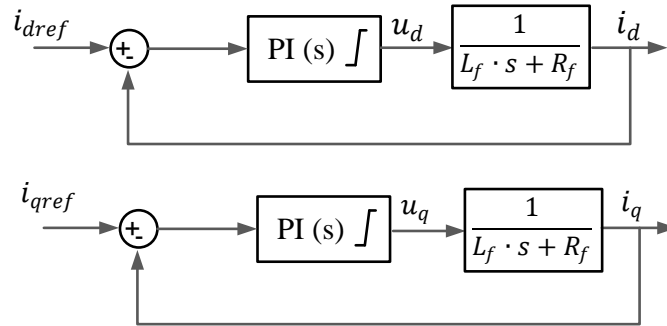


Figura A.11 – Diagrama esquemático do controlador de corrente.

O sistema simplificado da figura anterior é utilizado neste trabalho para especificar os valores numéricos dos ganhos do controlador PI. Para isto, considera-se a função de transferência deste tipo de controlador, conforme segue:

$$PI(s) = \frac{K_P \cdot s + K_I}{s}, \quad (\text{A.16})$$

onde K_P é o ganho proporcional e K_I o ganho integral. No sistema da Figura A.11, a função de transferência de laço aberto $\gamma(s)$ é:

$$\gamma(s) = \left(\frac{K_P}{L \cdot s} \right) \left(\frac{s + K_I/K_P}{s + R_f/L_f} \right). \quad (\text{A.17})$$

Uma prática comum no ajuste do controlador PI consiste em cancelar o pólo da planta (localizado em $s = -R_f/L_f$) mediante ajuste do zero do controlador (localizado em $s = -K_I/K_P$) [81], assim a função de transferência em laço aberto passa a ser:

$$\gamma(s) = \left(\frac{K_P}{L_f \cdot s} \right), \quad (\text{A.18})$$

e a função de transferência em laço fechado, ou seja, $\gamma(s)/(1 + \gamma(s))$ se torna:

$$G_I(s) = \left(\frac{K_P}{L_f \cdot s + K_P} \right) = \frac{1}{\tau_I \cdot s + 1}, \quad (\text{A.19})$$

onde $\tau_I = L_f/K_P$ é a constante de tempo resultante do laço fechado e a constante integral do controlador PI tem que ser definida como $K_I = R_f/\tau_I$.

A equação (A.19) mostra que a seleção de K_P e K_I , conforme os critérios descritos, origina uma resposta das correntes i_d e i_q baseada numa função de transferência de primeira ordem, onde a constante de tempo τ_I é um parâmetro de projeto. Segundo [81], τ_I deve ser dimensionada de tal forma que seja suficientemente pequena para garantir uma resposta rápida do controlador, mas suficientemente maior do que o período de chaveamento $1/f_{ch}$ (onde f_{ch} é a frequência de chaveamento do conversor).

Considerando que neste trabalho o controle é desenvolvido utilizando parâmetros em valor por unidade (pu), a expressão resultante para o ajuste dos ganhos do controlador é:

$$K_P = \frac{L_f}{\tau_I} \cdot \frac{I_{nCA}}{V_{nCC}/2} \quad \text{e} \quad K_I = \frac{R_f}{\tau_I} \cdot \frac{I_{nCA}}{V_{nCC}/2}, \quad (\text{A.20})$$

onde I_{nCA} é valor de pico da corrente nominal do lado CA e V_{nCC} é o valor nominal da tensão do elo CC. Para o sistema elétrico da Figura 3.11, e especificando a constante de tempo do controlador de corrente τ_I em 1 ms, os ganhos do controlador PI são: $K_P = 0,36$ [pu/pu] e $K_I = 5,4$ [pu/(pu·s)].

Modulação PWM

A modulação por largura de pulso (PWM) é o bloco de controle encarregado de determinar os comandos de fechamento/abertura das chaves do conversor VSC, de tal forma que a tensão de referência (v_{PWM}) seja sintetizada no lado CA do conversor. Neste trabalho é utilizado o chaveamento PWM com portadora triangular [27], [81].

D. Separação do sistema elétrico através do modelo

Bergeron de linha

No modelo de Bergeron, as linhas são representadas por parâmetros distribuídos de indutância e de capacitância e por uma resistência agrupada, dividida em três segmentos: $R/4$ em cada extremo da linha e $R/2$ na parte central, onde R é a resistência total da linha [29], [70]. Este modelo pode ser representado pelo par de redes de dois terminais da Figura A.12, onde a impedância de interface (Z_{Ber}) é:

$$Z_{Ber} = Z_0 + R/4, \quad (A.21)$$

onde Z_0 é a impedância característica da linha, definida como:

$$Z_0 = \sqrt{l/c}, \quad (A.22)$$

com l e c iguais à indutância e capacitância da linha por unidade de comprimento, nesta ordem. As injeções de corrente da representação da Figura A.12 são:

$$\begin{aligned} i_k(t) &= \frac{1 + H_{Ber}}{2} \cdot \left(\frac{-1}{Z_{Ber}} \cdot v_m(t - \tau) + i_{m,k}(t - \tau) \right) + \\ &\quad \frac{1 - H_{Ber}}{2} \cdot \left(\frac{-1}{Z_{Ber}} \cdot v_k(t - \tau) + i_{k,m}(t - \tau) \right); \\ i_m(t) &= \frac{1 + H_{Ber}}{2} \cdot \left(\frac{-1}{Z_{Ber}} \cdot v_k(t - \tau) - i_{k,m}(t - \tau) \right) + \\ &\quad \frac{1 - H_{Ber}}{2} \cdot \left(\frac{-1}{Z_{Ber}} \cdot v_m(t - \tau) - i_{m,k}(t - \tau) \right), \end{aligned} \quad (A.23)$$

com

$$H_{Ber} = \frac{Z_0 - R/4}{Z_0 + R/4} \quad \text{e} \quad \tau = d \cdot \sqrt{l \cdot c}, \quad (A.24)$$

onde d é o comprimento da linha.

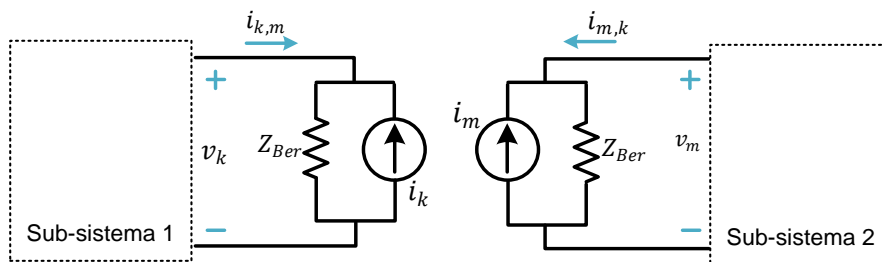


Figura A.12 – Representação do modelo de linha Bergeron.

A equação (A.23) mostra que as ondas de tensão e de corrente enviadas por um extremo da linha são recebidas pelo outro extremo com um atraso de τ segundos. Assim, se o valor de τ for tal que coincida com um múltiplo do passo de simulação (ou seja, $\tau = b \cdot \Delta t$, com b inteiro) é possível obter um desacoplamento natural entre os dois subsistemas conectados aos extremos do modelo da linha, no qual a solução de cada subsistema requer unicamente de informações históricas do outro subsistema (dados de tensão e corrente b passos de simulação atrás). Na prática, os SDTR costumam configurar os modelos Bergeron de linhas para forçar tal coincidência entre τ e Δt , o que eventualmente modifica os parâmetros originais da linha, introduzindo, por exemplo, capacitâncias parasitas [7], [126].