



MODELAGEM MATEMÁTICA DE ALGORITMOS *PHASE-LOCKED LOOP*  
BASEADOS EM CONTROLADORES PROPORCIONAIS INTEGRAIS E  
PROPOSTA DE UM NOVO ALGORITMO BASEADO EM ESTRATÉGIA *FUZZY*

Thiago Americano do Brasil

Dissertação de Mestrado 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 Mestre em Engenharia Elétrica.

Orientador: Mauricio Aredes

Rio de Janeiro  
Dezembro de 2013

MODELAGEM MATEMÁTICA DE ALGORITMOS *PHASE-LOCKED LOOP*  
BASEADOS EM CONTROLADORES PROPORCIONAIS INTEGRAIS E  
PROPOSTA DE UM NOVO ALGORITMO BASEADO EM ESTRATÉGIA *FUZZY*

Thiago Americano do Brasil

DISSERTAÇÃO 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 MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

---

Prof. Mauricio Aredes, Dr.-Ing.

---

Prof. Luis Guilherme Barbosa Rolim, Dr. Ing.

---

Prof. Guilherme Gonçalves Sotelo, D. Sc.

RIO DE JANEIRO, RJ – BRASIL

DEZEMBRO DE 2013

Brasil, Thiago Americano do

Modelagem matemática de algoritmos *Phase-Locked Loop* baseados em controladores proporcionais integrais e proposta de um novo algoritmo baseado em estratégia *fuzzy*/ Thiago Americano do Brasil – Rio de Janeiro: UFRJ/COPPE, 2013.

XIII, 111 p.: il.; 29,7 cm.

Orientador: Mauricio Aredes

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia Elétrica, 2013.

Referências Bibliográficas: p. 83-86.

1. PLL. 2. Lógica *Fuzzy*. 3. Sincronismo. I. Aredes, Mauricio. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

# Agradecimentos

Primeiramente, gostaria de agradecer a Nosso Senhor Jesus Cristo e Sua Mãe, a Santíssima Virgem Maria. Nos momentos de provação fui levantado e, nos de humildade, fui exaltado.

Agradeço também a minha amorosa família que sempre me apoiou incondicionalmente. Posso não ser o melhor filho todos os dias, mas tenho os melhores pais sempre!

Agradeço a minha namorada e minha melhor amiga Gabriela Chaves. Você me faz alguém melhor todos os dias, meu anjo. Tenha certeza disto!

Agradeço ao meu orientador e amigo Mauricio Aredes pelos diversos conselhos e suportes, tanto em âmbito profissional e acadêmico quanto pessoal.

A todos os membros da banca examinadora pela disponibilidade e presteza no dia da defesa.

A todos os amigos do Laboratório de Eletrônica de Potência e Média Tensão (LEMT). Vocês são praticamente minha segunda família!

A meus amigos de tantos anos que compreendem minha ausência em alguns eventos devido às responsabilidades com o curso de mestrado.

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

MODELAGEM MATEMÁTICA DE ALGORITMOS *PHASE-LOCKED LOOP*  
BASEADOS EM CONTROLADORES PROPORCIONAIS INTEGRAIS E  
PROPOSTA DE UM NOVO ALGORITMO BASEADO EM ESTRATÉGIA *FUZZY*

Thiago Americano do Brasil

Dezembro/2013

Orientador: Mauricio Aredes

Programa: Engenharia Elétrica

O correto ângulo de fase de uma corrente (ou tensão) de linha é uma informação primordial em diversos equipamentos de eletrônica de potência em geral, principalmente daqueles conectados a rede como: retificadores a tiristor, filtros ativos, *DVRs* e também em sistemas de geração distribuída como plantas fotovoltaicas e eólicas. Por esta razão, existem na literatura diversos algoritmos de *Phase-Locked Loop* (PLL). No entanto, muitas vezes tais algoritmos apresentam resposta dinâmica insatisfatória quando da presença de perturbações como harmônicos e degraus de fase. Neste trabalho, serão abordadas 3 (três) estratégias diferentes de PLL monofásico e a proposta de uma metodologia para sintonia de seus ganhos proporcional e integral. Além disto, apresenta-se um PLL baseado em teoria de controle *fuzzy* supervisor, tornando-se possível avaliar o comportamento destes sistemas frente às perturbações supracitadas, analisando-se, portanto, a melhoria proporcionada pelo PLL baseado em regras *fuzzy*.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

MATHEMATICAL MODELING OF PHASE-LOCKED LOOP ALGORITHMS  
BASED ON PROPORTIONAL INTEGRAL CONTROLLER AND PROPOSAL OF A  
NOVEL ALGORITHM BASED ON FUZZY STRATEGY

Thiago Americano do Brasil

December/2013

Advisor: Mauricio Aredes

Department: Electrical Engineering

The correct phase angle of a line voltage/current is primordial information in a vast array of power electronics equipments, mainly to those connected to grid as: thyristor based rectifiers, active power filters, DVRs and also distributed generation systems as photovoltaic and wind power plants. For this reason, there are several papers approaching Phase-Locked Loops. However, those algorithms generally present poor dynamic response when facing disturbances as phase-angle jumps and harmonic distortion. In this work, 3 (three) single-phase PLL strategies as well as a methodology for optimal tuning will be presented. Furthermore, a fuzzy based PLL will be introduced and its results can be compared to the other several PLL algorithms.

# Sumário

1. Introdução.....	1
1.1 Descrição de <i>Phase - Locked Loop</i> .....	1
1.2 Motivação .....	2
1.3 Objetivos.....	3
1.3.1 Objetivo Geral .....	3
1.3.2 Objetivos Específicos .....	3
1.4 Estrutura do Trabalho .....	3
2. Critérios de desempenho de <i>Phase-Locked Loops</i> .....	5
2.1 Critérios de desempenho.....	6
2.1.1 Análise de Resposta Transitória .....	6
2.2 Teoria de controladores .....	7
2.2.1 Interpretação no domínio do tempo.....	7
2.2.2 Interpretação no domínio da frequência .....	8
3. <i>Phase-Locked Loops</i> monofásicos na literatura.....	12
3.1 Estruturas de PLL monofásicos .....	13
3.1.1 <i>Power Based PLL (pPLL)</i> .....	14
3.1.2 <i>ParkPLL</i> .....	26
3.1.3 <i>Enhanced PLL (EPLL)</i> .....	33
4. Análise de desempenho do <i>pPLL</i> , <i>parkPLL</i> e <i>EPLL</i> .....	38
4.1 Digitalização .....	38
4.1.1 Transformada Z .....	38
4.1.2 Integração numérica .....	39
4.2 Mapeamento do semiplano esquerdo do plano s no plano z.....	39
4.3 Resposta a distúrbios dos modelos de PLL .....	40
4.3.1 Modelagem digital.....	40
4.3.2 Resposta ao degrau de fase para o modelo linearizado .....	42
4.3.3 Resposta à inserção de harmônicos para o modelo linearizado.....	43
4.3.4 Resposta à inserção de sub-harmônicos para o modelo linearizado.....	43
4.3.5 Resposta ao degrau de fase para o modelo digital.....	44

4.3.6	Resposta à inserção de harmônicos para o modelo digital .....	46
4.3.7	Resposta à inserção de sub-harmônicos para o modelo digital .....	47
4.4	Conclusões parciais.....	48
5.	Proposta de melhorias .....	50
5.1	Sub-harmônicos e a topologia do <i>pPLL</i> e <i>parkPLL</i> .....	50
5.2	Proposta de correção para o problema de sub-harmônicos.....	51
5.2.1	<i>pPLL</i> , <i>parkPLL</i> e <i>EPLL</i> .....	52
5.2.2	Simulação do modelo digital de <i>EPLL</i> .....	58
5.3	Conclusões parciais.....	61
6.	FAPLL ( <i>Fuzzy-Assisted PLL</i> ) .....	62
6.1	Introdução à lógica <i>Fuzzy</i> .....	62
6.1.1	Modelagem de plantas e processos.....	63
6.1.2	Recursos da lógica <i>fuzzy</i> .....	65
6.2	Pré-modelagem do controlador proporcional integral <i>Fuzzy</i> .....	66
6.3	Modelagem do controlador <i>fuzzy</i> .....	70
6.3.1	<i>Fuzzificação</i> .....	71
6.3.2	Composição da inferência a partir de uma tabela de regras .....	72
6.3.3	Defuzzificação do sistema a partir de uma tabela de regras.....	74
6.4	Simulação do modelo linearizado de FAPLL.....	76
6.4.1	Comparações do modelo linearizado do FAPLL com o <i>pPLL/parkPLL</i> e <i>EPLL</i> linear .....	76
6.5	Simulação do modelo digital de FAPLL .....	78
6.5.1	Resposta ao degrau de fase para o modelo digital do FAPLL .....	78
6.5.2	Resposta à inserção de harmônicos para o modelo digital do FAPLL.....	79
6.5.3	Resposta à inserção de sub-harmônicos para o modelo digital do FAPLL	79
7.	Conclusões e Trabalhos Futuros .....	81
7.1	Conclusões gerais .....	81
7.2	Trabalhos futuros .....	82
	Referências Bibliográficas.....	83
I.	Anexo 1 .....	87
a.	Sistemas de primeira ordem.....	87



b.	Sistemas de segunda ordem .....	89
c.	Especificações de resposta transitória.....	91
II.	Anexo 2 .....	95
a.	Controladores .....	95
b.	Controladores proporcional, integral e derivativo .....	96
c.	Análise da implementação do controlador proporcional integral .....	102
III.	Anexo 3 .....	104
a.	Forward Differences (Aproximação de Euler) .....	104
b.	Backward Differences (Aproximação de Euler).....	105
c.	Transformação bilinear (Aproximação de Tustin).....	106
d.	Mapeamento com Forward Differences .....	107
e.	Mapeamento com Backward Differences .....	107
f.	Mapeamento com aproximação de Tustin .....	108
IV.	Anexo 4 .....	110
a.	<i>Script</i> do MATLAB para cálculo dos parâmetros do <i>pPLL</i> .....	110
b.	<i>Script</i> do MATLAB para cálculo dos parâmetros do <i>parkPLL</i> .....	111

# Lista de figuras

<i>Figura 1.1 – Esquema básico de um PLL monofásico.</i>	2
<i>Figura 2.1 – Configuração de pólos e zeros de um controlador PI.</i>	7
<i>Figura 2.2 – Diagrama de sistema de controle simples para interpretação no domínio da frequência de um controlador PI.</i>	8
<i>Figura 2.3 – Resposta ao degrau com controlador PI sintonizado em <math>K_p = 15</math> e <math>T_i = 0.01s</math>.</i>	9
<i>Figura 2.4 – Diagrama de resposta em frequência do controlador PI sintonizado em <math>K_p = 15</math> e <math>T_i = 0.01s</math>.</i>	9
<i>Figura 2.5 - Resposta ao degrau com controlador PI sintonizado em <math>K_p = 5</math> e <math>T_i = 0.1s</math>.</i>	10
<i>Figura 2.6 - Diagrama de resposta em frequência do controlador PI sintonizado em <math>K_p = 5</math> e <math>T_i = 0.1s</math>.</i>	10
<i>Figura 3.1 – Modelo linear de PLL.</i>	14
<i>Figura 3.2 – pPLL monofásico.</i>	15
<i>Figura 3.3 – Modelo linearizado do pPLL.</i>	16
<i>Figura 3.4 – Resposta ao degrau e diagrama de bode em malha aberta do pPLL.</i>	25
<i>Figura 3.5 – Resposta inicial do modelo digital e erro do controlador PI do pPLL.</i>	25
<i>Figura 3.6 – Comportamento do modelo real para entrada unitária de fundamental e 2º harmônico.</i>	25
<i>Figura 3.7 – parkPLL monofásico. Adaptado de [19].</i>	26
<i>Figura 3.8 – Resposta ao degrau e diagrama de bode em malha aberta do parkPLL.</i>	32
<i>Figura 3.9 – Resposta inicial do modelo digital e erro do controlador PI do parkPLL.</i>	32
<i>Figura 3.10 - Comportamento do modelo real do parkPLL para entrada unitária de fundamental e 2º harmônico.</i>	32
<i>Figura 3.11 – EPLL monofásico.</i>	33
<i>Figura 3.12 - Resposta ao degrau e diagrama de bode em malha aberta do EPLL.</i>	37
<i>Figura 3.13 – Resposta inicial do modelo digital e erro do controlador PI do EPLL.</i>	37
<i>Figura 3.14 – Comportamento do modelo real do EPLL para entrada unitária de fundamental e 2º harmônico.</i>	37
<i>Figura 4.1 – Exemplo de amostragem de sinal analógico utilizando-se amostrador de ordem zero (ZOH).</i>	39
<i>Figura 4.2 – Resposta ao degrau de fase para o modelo linearizado.</i>	42
<i>Figura 4.3 – Resposta a inserção de 1.0 p.u de 2º harmônico para o modelo linearizado.</i>	43
<i>Figura 4.4 – Resposta a inserção de 1.0 p.u de perturbação sub-harmônica de 50 Hz para o modelo linearizado.</i>	44
<i>Figura 4.5 - Resposta do pPLL ao degrau de 30° em <math>t = 0.15s</math>.</i>	45
<i>Figura 4.6 - Resposta do parkPLL ao degrau de 30° em <math>t = 0.15s</math>.</i>	45
<i>Figura 4.7 - Resposta do EPLL ao degrau de 30° em <math>t = 0.15s</math>.</i>	45

<i>Figura 4.8 - Resposta do pPLL a inserção de 1.0 p.u de perturbação de 2° harmônico em t = 0.15s.</i>	46
<i>Figura 4.9 - Resposta do parkPLL a inserção de 1.0 p.u de perturbação de 2° harmônico em t = 0.15s.</i>	46
<i>Figura 4.10 - Resposta do EPLL a inserção de 1.0 p.u de perturbação de 2° harmônico em t = 0.15s.</i>	47
<i>Figura 4.11 - Resposta do pPLL a inserção de 1.0 p.u de perturbação sub-harmônica de 50 Hz em t = 0.15s.</i>	47
<i>Figura 4.12 - Resposta do parkPLL a inserção de 1.0 p.u de perturbação sub-harmônica de 50 Hz em t = 0.15s.</i>	48
<i>Figura 4.13 - Resposta do EPLL a inserção de 1.0 p.u de perturbação sub-harmônica de 50 Hz em t = 0.15s.</i>	48
<i>Figura 5.1 – Diagramas de resposta em frequência do pPLL/parkPLL e EPLL.</i>	51
<i>Figura 5.2 - Resposta do filtro passa-baixa de 2ª ordem com frequência de ressonância em <math>\omega_c = 377 \text{ rad/s}</math> para diversos valores de fator de qualidade Q.</i>	53
<i>Figura 5.3 – Diagramas de Bode e respostas ao degrau do EPLL e filtro aproximado para <math>\omega_{c2} = 30000</math> e <math>Q=1</math>.</i>	54
<i>Figura 5.4 - Diagrama de Bode do EPLL para <math>\omega_{c2} = 3000</math> e <math>Q=1</math>.</i>	56
<i>Figura 5.5 - Diagrama de Bode do EPLL para <math>\omega_{c2} = 300</math> e <math>Q=1</math>.</i>	57
<i>Figura 5.6 – Comportamento do EPLL com parâmetros <math>\omega_{c2} = 30000</math> e <math>Q = 1</math> e 20% de perturbação sub-harmônica de 50 Hz.</i>	59
<i>Figura 5.7 - Comportamento do EPLL com parâmetros <math>\omega_{c2} = 3000</math> e <math>Q = 1</math> e 20% de perturbação sub-harmônica de 50 Hz.</i>	60
<i>Figura 5.8 - Comportamento do EPLL com parâmetros <math>\omega_{c2} = 300</math> e <math>Q = 1</math> e 20% de perturbação sub-harmônica de 50 Hz.</i>	61
<i>Figura 6.1 – Identificação do comportamento do operador. Adaptado de [31].</i>	65
<i>Figura 6.2 – Modelo discreto de PLL. Adaptado de [7].</i>	66
<i>Figura 6.3 – Loop Bandwidth normalizado em relação ao coeficiente de amortecimento <math>\xi</math>.</i>	68
<i>Figura 6.4 – Excursão do ganho proporcional do controlador em relação ao loop bandwidth.</i>	69
<i>Figura 6.5 - Excursão do ganho integral do controlador em relação ao loop bandwidth.</i>	70
<i>Figura 6.6 – Diagrama de blocos do controlador fuzzy proposto para o Loop Filter do FAPLL.</i>	70
<i>Figura 6.7 – Funções de pertinência do erro e derivada do erro.</i>	72
<i>Figura 6.8 – Produto lógico fuzzy proveniente da regra 1.</i>	73
<i>Figura 6.9 – Produto lógico fuzzy proveniente da regra 2.</i>	73
<i>Figura 6.10 – Soma lógica (máx) fuzzy da inferência mín-máx.</i>	74
<i>Figura 6.11 – Excursão da variável <math>u_k</math> a partir do valor fuzzificado de <math>e_k</math> e <math>e_k'</math>.</i>	75
<i>Figura 6.12 – Funções de pertinência singleton de saída.</i>	76
<i>Figura 6.13 – Comparação entre FAPLL e demais PLLs otimamente sintonizados.</i>	77
<i>Figura 6.14 – Resposta do FAPLL ao degrau de 30° em t = 0.15s.</i>	78

<i>Figura 6.15 - Resposta do FAPLL a inserção de 1.0 p.u de 2° harmônico em <math>t = 0.15s</math>.</i>	79
<i>Figura 6.16 - Resposta do FAPLL a inserção de 1.0 p.u de sub-harmônico de 50 Hz em <math>t = 0.15s</math>.</i>	80
<i>Figura I.1 - Resposta ao degrau de um sistema de primeira ordem de ganho <math>K=1</math>.</i>	88
<i>Figura I.2 – Resposta a rampa unitária de uma função de primeira ordem com <math>K=1</math> e <math>T=50</math> (azul). Em verde, a rampa unitária de referência.</i>	89
<i>Figura I.3 - Comportamento da resposta ao degrau de uma função de segunda ordem em função do fator de amortecimento <math>\xi</math>.</i>	91
<i>Figura I.4 – Curva de resposta ao degrau unitário mostrando <math>t_d</math>, <math>t_r</math>, <math>t_p</math>, <math>M_p</math> e <math>t_s</math>.</i>	94
<i>Figura II.1 - Diagrama de bloco de um processo.</i>	95
<i>Figura II.2 – Diagrama de controladores série e por realimentação.</i>	96
<i>Figura II.3 - Diagrama de blocos de um controlador proporcional. Adaptada de [13].</i>	97
<i>Figura II.4 – Diagrama de blocos de um controlador integral. Adaptada de [13].</i>	98
<i>Figura II.5 – Diagrama de blocos de um controlador proporcional integral. Adaptado de [13].</i>	99
<i>Figura II.6 – Diagramas esboçando a resposta ao degrau de um controlador proporcional (verde) e proporcional integral(amarelo).</i>	99
<i>Figura II.7 – Resposta ao degrau unitário (verde) perante o erro atuante do controlador integral (amarelo).</i>	99
<i>Figura II.8 – Diagrama de blocos de um controlador proporcional derivativo. Adaptado de [13].</i>	100
<i>Figura II.9 - Diagramas esboçando a resposta a rampa unitária de um controlador proporcional (amarelo) e proporcional e derivativo (verde).</i>	101
<i>Figura II.10 - Diagrama de blocos de um controlador proporcional – integral e derivativo. Adaptado de [13].</i>	102
<i>Figura II.11 - Sistema de controle com controlador PI.</i>	102
<i>Figura III.1 - Mapeamento da região de estabilidade do plano <math>s</math> no plano <math>z</math> utilizando Forward Differences. Adaptado de [25].</i>	107
<i>Figura III.2 - Mapeamento da região de estabilidade do plano <math>s</math> no plano <math>z</math> utilizando Backward Differences. Adaptado de [25].</i>	108
<i>Figura III.3 - Mapeamento da região de estabilidade do plano <math>s</math> no plano <math>z</math> utilizando aproximação de Tustin. Adaptado de [25].</i>	109

# Lista de tabelas

<i>Tabela 5.1 – Tabela comparativa ilustrando o efeito amplificativo de perturbações sub-harmônicas no pPLL/parkPLL e EPLL a partir dos valores obtidos na Figura 5.1. ....</i>	<i>51</i>
<i>Tabela 5.2 - Tabela comparativa ilustrando o efeito amplificativo de perturbações sub-harmônicas no EPLL e filtro para <math>\omega c2 = 30000</math> e <math>Q=1</math>. .....</i>	<i>55</i>
<i>Tabela 5.3 - Tabela comparativa ilustrando o efeito amplificativo de perturbações sub-harmônicas no EPLL e filtro para <math>\omega c2 = 3000</math> e <math>Q=1</math>. .....</i>	<i>56</i>
<i>Tabela 5.4 - Tabela comparativa ilustrando o efeito amplificativo de perturbações sub-harmônicas no EPLL e filtro para <math>\omega c2 = 300</math> e <math>Q=1</math>. .....</i>	<i>57</i>
<i>Tabela 6.1 – Tabela de regras do FAPLL. ....</i>	<i>74</i>
<i>Tabela III.1 - Tabela de aproximações por integração numérica. ....</i>	<i>107</i>

# 1. Introdução

## 1.1 Descrição de *Phase - Locked Loop*

*Phase Locked Loops* (PLL) têm um fator preponderante de providenciar um sinal de referência de fase sincronizado com um sistema de corrente alternada em questão [1]. Este algoritmo é amplamente utilizado em eletrônica de potência tanto na forma de *hardware* quanto de *software* embarcado. Em geral, com o advento de microprocessadores mais poderosos e, inclusive, com unidades aritméticas de ponto flutuante especializadas, prefere-se atualmente seguir o caminho da discretização deste algoritmo em um processador, o que garante mais robustez e capacidade de modificação de seus parâmetros de forma mais rápida e amigável.

O ângulo de fase e a frequência da rede no qual o equipamento está inserido são informações vitais para equipamentos do tipo FACTS (*Flexible AC Transmission Systems*), APF (*active power filters*), DVR (*dynamic voltage restorers*), UPS (*uninterruptable power supply*) e equipamentos de armazenamento de energia e geração distribuída [2].

De modo a estimar os parâmetros supracitados dos sinais monofásicos de entrada de um PLL, vários métodos foram propostos. Em [3], listou-se os mais recorrentes na literatura.

Do ponto de vista das aplicações de conversores de potência conectados a rede elétrica, pode-se tratar o PLL como um sistema de controle em malha-fechada realimentado, o qual sincroniza seu sinal de saída em frequência, bem como em fase, com a componente fundamental da rede. Tomando-se em conta as diferenças presentes nas diversas topologias em [3], pode-se dizer que todas as técnicas são compostas a partir das 3 (três) estruturas básicas a seguir: 1) *Phase Detector* (PD), 2) *Loop Filter* (LF), e 3) *Voltage-Controlled Oscillator* (VCO), como pode ser ilustrado na Figura 1.1.

Basicamente, o *Phase Detector* é responsável pela detecção da diferença de fase entre o sinal de entrada  $V_i$  e o sinal de saída e realimentado  $V_o$ . O *Loop Filter* se apresenta como um filtro com características de passa-baixa. O mesmo será abordado

com bastante profundidade no Capítulo II. O *VCO* (*Voltage-Controller Oscillator*) é a parcela cuja frequência de saída é função de sua tensão de entrada.

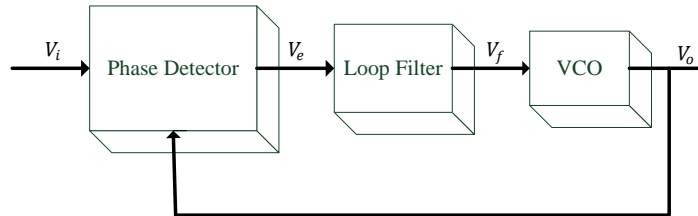


Figura 1.1 – Esquema básico de um PLL monofásico.

Em geral, a diferença primordial em termos topológicos entre diferentes PLLs está em como o bloco do *Phase Detector* é implementado. Em alguns casos, como o do presente trabalho, a diferenciação se encontra em torno do *Loop Filter*.

De modo usual, a teoria envolvendo o modelo matemático de *Phase-Locked Loops* pode ser dividida em 2 (duas) frentes: Teoria Linear e Teoria Não Linear. A teoria não linear envolve a implementação real do PLL, no entanto, é mais complicada e de difícil análise para o projetista [4]. Entende-se pelo trabalho do projetista a tarefa de sintonização das malhas presentes na Figura 1.1, mais frequentemente do *Loop Filter*.

Desta forma, naturalmente, PLLs são geralmente modelados a partir de teoria de controle linear. Pode-se, portanto, partir de um modelo real pré-definido no domínio do tempo e chegar ao modelo linearizado, o qual descreve com boa precisão e acurácia o comportamento do algoritmo na maior parte das aplicações [4].

## 1.2 Motivação

Uma extensa pesquisa vem sendo desenvolvida no campo de algoritmos de sincronismo como visto em [5, 6, 7]. No entanto, a partir do uso exclusivo de teoria de controle clássico, não é possível realizar a implementação de algoritmos cuja eficiência se aplique as mais diversas condições existentes dos sinais de tensão e corrente de um conversor de eletrônica de potência sem a necessidade da resintonia dos parâmetros do *Loop Filter* manualmente. Portanto, ao realizar o sincronismo em equipamentos cujos sinais de tensão e corrente monitorados podem sofrer variações bruscas em sua forma é imprescindível a utilização de um algoritmo com capacidade adaptativa, de modo que

uma má sintonia do *Phase-Locked Loop* não tenha como consequência a má operação de um equipamento como um todo.

### **1.3 Objetivos**

De modo a considerar as necessidades mencionadas na motivação deste trabalho, os objetivos serão apresentados a seguir:

#### **1.3.1 Objetivo Geral**

Analisar as principais propostas de algoritmos de *Phase-Locked Loop* monofásicos já existentes na literatura e propor uma nova estratégia robusta, baseada em teoria de controle *Fuzzy Supervisória*.

#### **1.3.2 Objetivos Específicos**

- Expor as principais propostas de algoritmos de PLL presentes na literatura, estudá-las e apresentar suas respectivas linearizações.
- Analisar, a partir da teoria de controle clássica, os critérios de desempenho das respectivas topologias de PLL.
- Propor uma nova estratégia não linear e *fuzzy* supervisória que acrescente mais robustez e melhorias na estabilidade geral do algoritmo de PLL.
- Apresentação e comparação dos resultados envolvendo a nova estratégia frente às estratégias supracitadas.

### **1.4 Estrutura do Trabalho**

O presente trabalho apresenta uma estrutura de 7 (sete) partes, incluindo este Capítulo. No Capítulo 2 é apresentada a teoria de sistemas de primeira e segunda ordem, bem como uma breve revisão nos conceitos de critério de desempenho.

No Capítulo 3, serão apresentadas e analisadas do ponto de vista de critérios de desempenho as principais propostas de algoritmos monofásicos presentes na literatura.

No Capítulo 4, introduz-se o processo de digitalização das malhas de controle dos PLLs, além de análises práticas a partir dos modelos linearizados e reais, podendo-se confirmar a teoria proposta no Capítulo 3.



No Capítulo 5, serão apresentadas propostas para melhorias na capacidade de sincronismo dos PLLs em regime permanente.

No Capítulo 6 será apresentada a estratégia *fuzzy* supervisória utilizada no algoritmo FAPLL proposto, bem como os resultados obtidos.

No Capítulo 7 serão abordadas as últimas considerações e as principais conclusões do trabalho desenvolvido.

## 2. Critérios de desempenho de *Phase-Locked Loops*

*Phase-Locked Loops* já se encontram em formas digitais ou analógicas em equipamentos elétricos e/ou eletrônicos há algum tempo [8, 9]. A literatura nos mostra que seu uso foi disseminado nos primórdios das televisões com tecnologia de tubos de raios catódicos, realizando suas sincronias vertical e horizontal. Em alguns aspectos, trata-se já de um campo amplamente discutido e maturado [10]. No entanto, PLLs são incluídos em um gigantesco número de aplicações eletrônicas envolvendo tecnologia de ponta, o que torna as discussões sobre diferentes topologias de PLLs e suas melhorias ainda assunto bastante recorrente.

Como já mostrado na Figura 1.1, o diagrama de blocos mais básico de um PLL deve conter as seguintes componentes:

- *Phase Detector*: Trata-se de um dispositivo não linear cuja saída contém a diferença de fase entre a referência e o sinal de saída.
- Oscilador controlado por tensão (*VCO*): Trata-se de outro dispositivo não linear que produz uma resposta oscilante cuja frequência é controlada por uma tensão de entrada determinada pela resposta do estágio de filtro (*Loop Filter*).
- *Loop Filter*: É comumente presente na forma de um dispositivo cuja função de transferência seja a de um filtro passa-baixa. É geralmente atribuída a esta função um controlador PI (Proporcional Integral).

PLLs possuem diversas características únicas quando analisados pelo ponto de vista de controle clássico. Primeiramente, sua correta operação depende da premissa de que o mesmo é não linear. Este *loop* não poderia existir sem a presença de dois dispositivos não lineares, mais precisamente o *phase detector* e o *VCO* [10].

De acordo com [10], inicialmente esperava-se que os estudos de PLLs concentrassem mais a parcela de pessoas *experts* em controle moderno, devido a característica não linear de sua malha. No entanto, de fato a teoria de controle aplicada na maior parte da literatura envolvendo PLLs é baseada em controle linear, e em alguns casos, somadas a parcelas de não linearidades, como visto em [11, 12].

A análise de estabilidade e implementação da malha de controle do PLL tende a ser feita através da combinação de análise linear, simulação e adequação aos requisitos impostos pelo projeto.

Desta forma, faz-se necessário um estudo preliminar sobre análise de desempenho transitório e em regime permanente de uma função de transferência. Além disto, será realizada uma breve introdução ao universo dos controladores largamente utilizados em controle moderno e clássico. A partir de todas estas informações será possível, então, estudar as diversas topologias de PLLs mais utilizadas em eletrônica de potência com todo o aparato matemático necessário para tal.

## **2.1 Critérios de desempenho**

A resposta temporal de um sistema consiste de duas partes: a resposta transitória e a resposta em regime permanente (estacionária). Basicamente, a resposta transitória é a parte da resposta que vai do estado inicial até o estado final. Em relação à resposta estacionária, trata-se da maneira como a saída se comporta após a extinção de todas as formas de transitório.

Nas seções a seguir, serão explicados os aspectos referentes às análises transitórias e em regime permanente no que se refere a sistemas de primeira e segunda ordem.

### **2.1.1 Análise de Resposta Transitória**

A análise de resposta transitória é uma das formas mais utilizadas para avaliação do critério de desempenho de um sistema. Uma das maneiras de realizar esta análise é a partir da resposta ao degrau de um sistema a ser estudado. No caso do presente trabalho, é de extrema importância que sejam fornecidas as ferramentas necessárias para tal.

Portanto o Anexo 1 traz um resumo da teoria envolvendo sistemas de primeira e segunda ordem tal como abordado em [13, 14].

## 2.2 Teoria de controladores

Toda a base teórica neste capítulo tem como objetivo final a implementação de sistemas de controle. O Anexo 2 aborda uma breve introdução à teoria de controladores.

### 2.2.1 Interpretação no domínio do tempo

A configuração de pólos e zeros de um controlador PI da equação (II.7) é mostrada abaixo na Figura 2.1. Em primeira análise, pode parecer que o controle PI diminui o erro de estado estacionário em detrimento da estabilidade geral do sistema. No entanto, deve ser lembrado que a localização do zero de  $G_c(s)$  deve ser escolhida apropriadamente, de modo que tanto o amortecimento quanto o erro estacionário possam ser melhorados.

Como bem observado em [15], o controlador PI é basicamente um filtro *passa-baixa*. Deste modo, é completamente normal que o sistema compensado pelo controlador tenha um tempo de subida mais lento e um tempo de acomodação maior. Uma forma de sintonia do controlador muito utilizada é escolhendo valores não muito elevados para  $K_i$  e  $K_p$  de modo que o zero do sistema se localize relativamente próximo à origem e distante dos pólos dominantes do sistema.

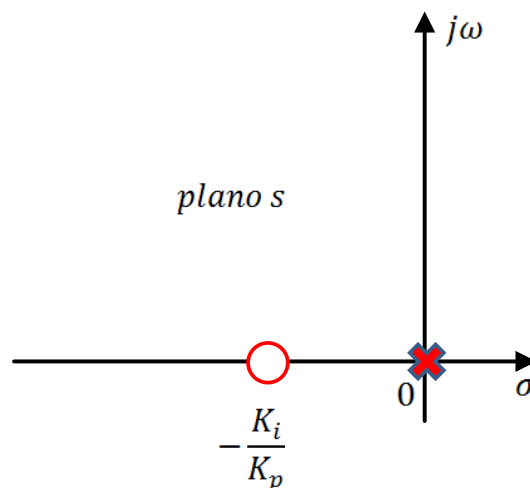


Figura 2.1 – Configuração de pólos e zeros de um controlador PI.

### 2.2.2 Interpretação no domínio da frequência

Nesta seção será analisada a resposta em frequência do controlador PI presente na equação (II.7). A principal motivação desta seção é mostrar o *trade-off* necessário na sintonia dos parâmetros de um controlador PI. Como foi relatado na seção anterior, a escolha de valores de  $K_p$  e  $T_i$  não é uma tarefa simples. Devem ser levados em consideração todos os critérios de desempenho levantados Anexo 1. De forma a evidenciar o *trade-off* mencionado acima, serão mostrados dois casos com parâmetros  $K_p$  e  $T_i$  distintos.

Tomando um sistema de controle simples, como o da Figura 2.2, é possível verificar como tempo de acomodação e largura de banda são variáveis mutuamente excludentes. Basicamente, a Figura 2.2 apresenta uma planta de primeira ordem sendo controlada a partir de um controlador proporcional integral. Serão realizados dois ensaios diferentes para esta planta. O primeiro apresentará como parâmetros  $K_p$  e  $T_i$  os valores de 15 e 0.01s, respectivamente. O segundo terá como parâmetros  $K_p$  e  $T_i$  os valores de 5 e 0.1s, respectivamente. A resposta ao degrau, bem como a resposta em frequência do sistema (em malha aberta) são analisadas abaixo:

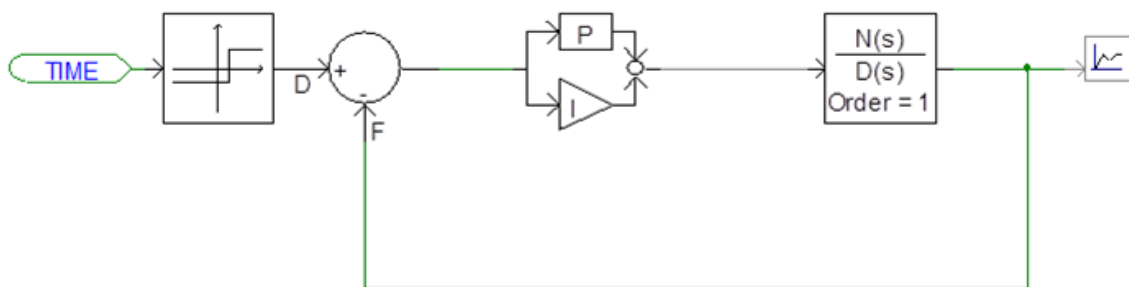


Figura 2.2 – Diagrama de sistema de controle simples para interpretação no domínio da frequência de um controlador PI.

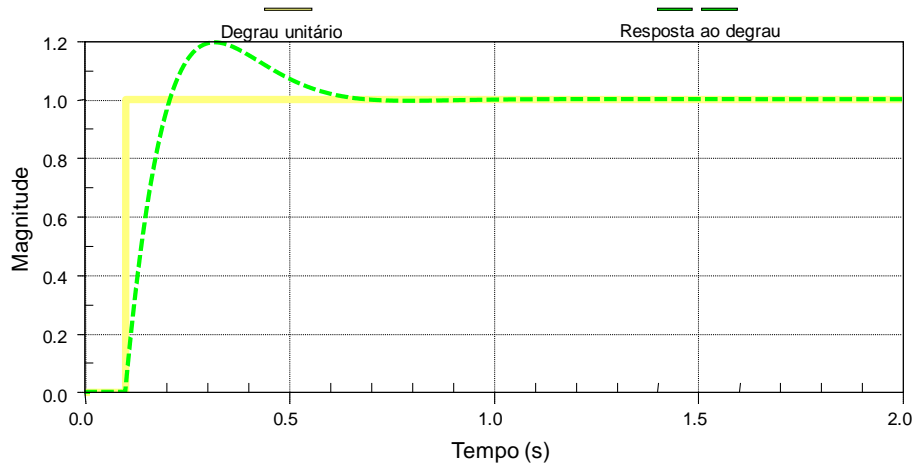


Figura 2.3 – Resposta ao degrau com controlador PI sintonizado em  $K_p = 15$  e  $T_i = 0.01s$ .

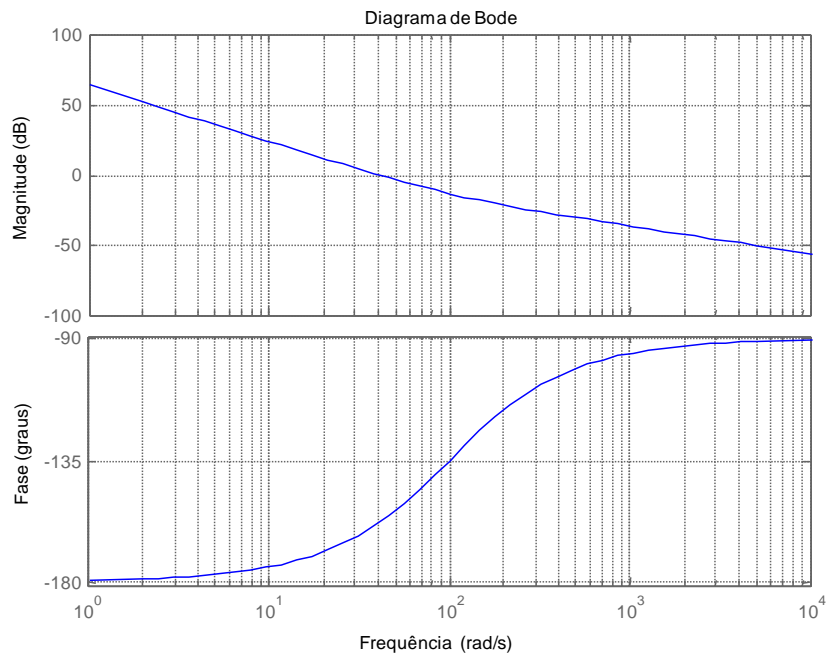


Figura 2.4 – Diagrama de resposta em frequência do controlador PI sintonizado em  $K_p = 15$  e  $T_i = 0.01s$ .

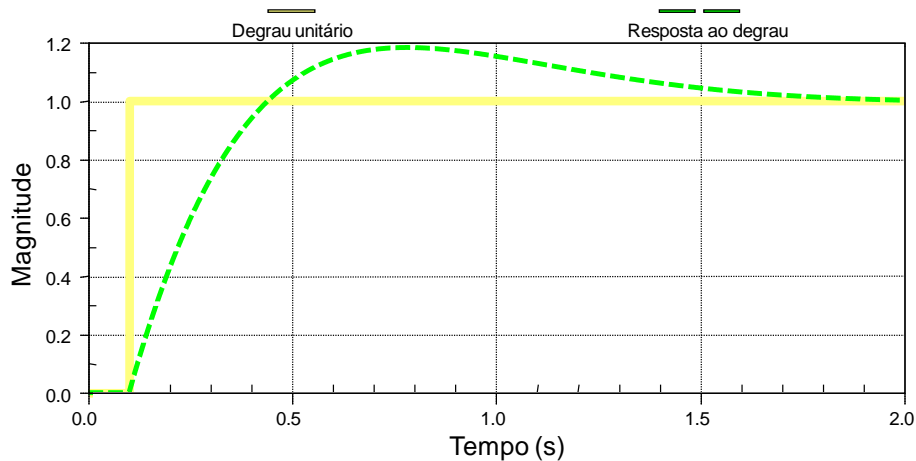


Figura 2.5 - Resposta ao degrau com controlador PI sintonizado em  $K_p = 5$  e  $T_i = 0.1s$ .

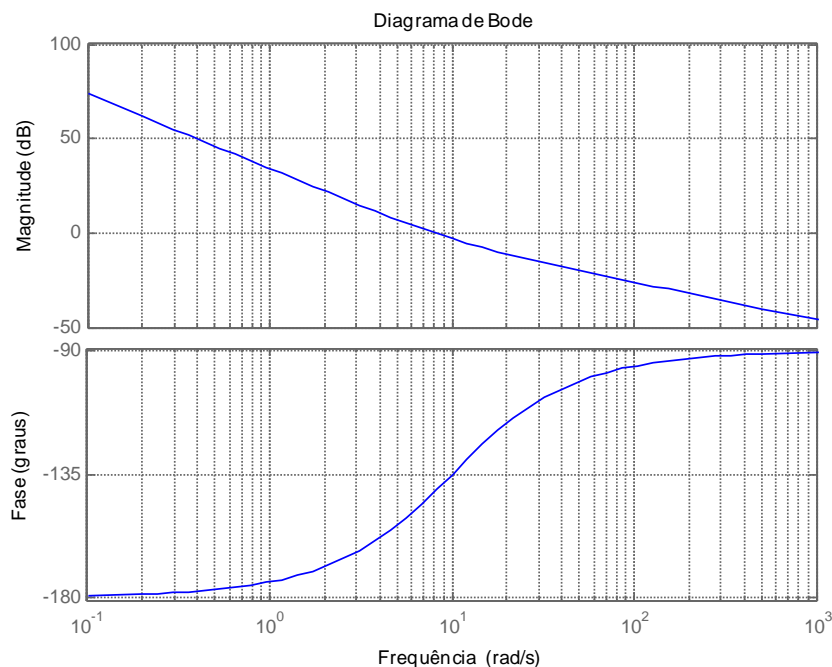


Figura 2.6 - Diagrama de resposta em frequência do controlador PI sintonizado em  $K_p = 5$  e  $T_i = 0.1s$ .

É possível observar a partir das Figura 2.3 e Figura 2.4 que o controlador sintonizado em  $K_p = 15$  e  $T_i = 0.01s$  apresenta tempo de acomodação  $t_s \cong 0.75s$  e atenuação a partir da frequência  $\omega_c = 40$  rad/s. No caso do controlador sintonizado em  $K_p = 5$  e  $T_i = 0.1s$ , pode-se verificar a partir das Figura 2.5 e Figura 2.6 que o tempo de acomodação  $t_s \cong 2s$  e a atenuação começa a ser verificada a partir de  $\omega_c = 8$  rad/s. Ou seja, de posse destes dados é possível verificar a incongruência inerente a controladores baseados em teoria de controle clássico, mais precisamente do

controlador PI. Uma diminuição na banda passante aumenta de forma geral a estabilidade e rejeição a perturbações do sistema. No entanto, o faz em detrimento de uma degradação na dinâmica do sistema, introduzindo-se atrasos que nem sempre são admissíveis em certos tipos de controle. Este ponto será abordado e contornado no Capítulo 6.

Como será visto no Capítulo 3 a seguir, um controlador PI bem sintonizado de uma malha de controle de *Phase-Locked Loop* deve possuir os seguintes atributos:

- Melhorar o amortecimento e diminuir o percentual de ultrapassagem  $M_p$  do sinal.
- Ajustar de forma ideal a banda do sistema de controle, de forma que não degrade muito a dinâmica do mesmo, enquanto é capaz de rejeitar perturbações mais proeminentes.
- Aprimorar estabilidade do sistema a partir do conceito de margem de fase.



### 3. *Phase-Locked Loops* monofásicos na literatura

O correto ângulo de fase de uma corrente (ou tensão) de linha é uma informação primordial em diversos equipamentos de eletrônica de potência em geral, principalmente daqueles conectados a rede como: retificadores a tiristor, filtros ativos, *S-DVRs* e também em sistemas de geração distribuída como plantas fotovoltaicas e eólicas.

Em sistemas *UPS* (*uninterruptible power supply*), é necessária a garantia de boa sincronização entre a tensão de saída proveniente do inversor e a fonte de tensão primária. Para configuração de múltiplos *UPSs* em paralelo, a sincronização deve ser, portanto ainda mais precisa. De forma a estimar o ângulo de fase, existem métodos em malha aberta e malha fechada [16, 17]. Neste capítulo serão estudadas 3 (três) estratégias diferentes de métodos de PLL.

Geralmente, a frequência do sinal de entrada do PLL varia dentro de uma faixa limitada, mesmo em sistemas isolados. Em sistemas conectados a rede, a taxa de variação é limitada pela inércia mecânica dos geradores. Além disto, na ocorrência de uma falta na rede, os equipamentos a ela conectados se tornam expostos a variações bruscas no ângulo de fase e subtensões [18]. Somando-se a isto, harmônicos, *notches*, *spikes* e outros tipos de perturbações indesejáveis são comuns na rede de distribuição [19].

As razões citadas acima demonstram a necessidade de desenvolvimento de algoritmos de sincronismo robustos de forma a contemplar os requisitos de desempenho cada vez mais rigorosos da indústria. Os aspectos de mérito que qualificam um algoritmo de *Phase-Locked Loop* são: erro de fase em regime permanente, velocidade de resposta a perturbações na frequência, amplitude e/ou fase e capacidade de rejeição a harmônicos.

Nos últimos anos, diversos algoritmos de PLL com as mais diferentes características foram desenvolvidos e apresentados na literatura [16, 20, 21]. Um dos grandes interesses deste trabalho é facilitar a visualização do comportamento de cada PLL (e comparar seus desempenhos) a partir da forma como serão apresentados os resultados iniciais. Em geral, os resultados são apresentados a partir de formas de onda

dente de serra ou senóides, representando os ângulos de fase real e estimado. Desta forma, uma primeira apresentação do modelo linearizado de cada PLL, demonstrando-se seus aspectos de mérito a partir deste torna-se extremamente atraente. Após isto, pode ser verificada a pertinência deste modelo em relação ao real.

O principal objetivo deste capítulo é avaliar, descrever e desenvolver (a partir da teoria de controle clássico) uma metodologia para ajuste de parâmetros de 3 (três) algoritmos de PLL monofásicos muito difundidos na literatura.

As estruturas selecionadas possuem relativamente simples implementação digital e baixa carga computacional.

O primeiro algoritmo a ser analisado é baseado na teoria de potência instantânea, como proposto em [20]. É conhecido como *power-based PLL* ou *pPLL*. No entanto, neste caso, trata-se da sua versão monofásica.

O segundo algoritmo a ser analisado é baseado na transformação inversa de *Park*, como visto em [3]. Este algoritmo será abreviado por *parkPLL*.

O terceiro e último algoritmo a ser visto é baseado em teoria adaptativa da detecção de fase, presente em [16]. O algoritmo baseado nesta teoria é conhecido como *enhanced PLL*, ou *EPLL*.

### **3.1 Estruturas de PLL monofásicos**

Apesar de suas diferenças básicas, todos os algoritmos de PLL podem ser derivados de uma estrutura padrão, divididas entre *Phase Detector*, *Loop Filter* e *Voltage Controlled Oscillator*. A diferença construtiva nos PLLs apresentados a seguir basicamente se refere ao conteúdo do *Phase Detector* como já relatado no Capítulo 1.

A implementação do *Loop Filter* e do *VCO* dos PLLs apresentados nas próximas seções segue um mesmo padrão, buscando sempre a máxima otimização possível entre baixo tempo de acomodação, estabilidade transitória e boa imunidade a ruídos e a perturbação de frequências acima da nominal. Basicamente, a primeira análise é realizada a partir do modelo linearizado, como visto na Figura 3.1.

A entrada e saída desta estrutura são o ângulo de fase  $\theta$  do sinal de entrada e o ângulo estimado  $\hat{\theta}$ , respectivamente. O termo  $\omega_{ff}$  define a frequência central em torno da qual o PLL irá sincronizar seu ângulo de fase.

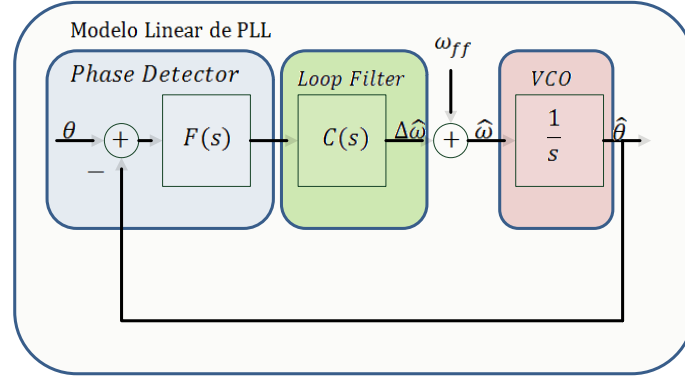


Figura 3.1 – Modelo linear de PLL.

### 3.1.1 Power Based PLL (pPLL)

O diagrama de blocos da estrutura clássica do *pPLL* pode ser visto na Figura 3.2. Devido ao fato de sua parcela *Phase Detector* ser baseada em apenas um único multiplicador, a analogia com o conceito de potência elétrica pode ser utilizada de forma a tornar o entendimento mais intuitivo [19]. Caso a potência média fictícia seja nula, então, espera-se que a corrente fictícia  $i_s$  esteja em quadratura com a fundamental da tensão de entrada  $e_i$ .

Assumindo-se uma tensão de entrada  $e_i$  puramente senoidal, é possível chegar à equação:

$$p = V \cos \theta \sin \hat{\theta} \quad (3.1)$$

A partir do uso de identidades matemáticas, é possível a verificação de que:

$$p = \frac{V}{2} \sin(\hat{\theta} - \theta) + \frac{V}{2} \sin(\hat{\theta} + \theta) \quad (3.2)$$

O filtro passa-baixa incluído no *Phase Detector* possui a propriedade de, em regime permanente, filtrar os harmônicos de  $p$  levando a  $\bar{p}$ . Desta forma, levando-se em consideração que  $\theta = \omega t + \phi$ ,  $\hat{\theta} = \hat{\omega} t + \hat{\phi}$  e trabalhando em torno do ponto de operação  $\hat{\omega} \cong \omega$ , é possível estabelecer o valor de  $p$  para pequenas variações de  $\phi - \hat{\phi}$  como:

$$\bar{p} \cong \frac{V}{2}(\hat{\phi} - \phi) \quad (3.3)$$

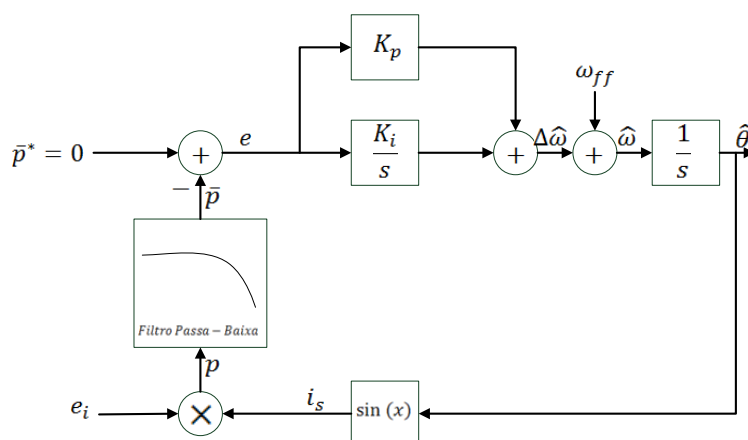


Figura 3.2 – *pPLL monofásico.*

Neste caso, a dinâmica do *phase Detector* depende inteiramente da estrutura do filtro passa-baixa. Como foi analisado em (3.2), o produto da tensão de entrada com a corrente fictícia  $i_s$  produz uma componente de segundo harmônico que deve ser filtrada. No entanto, o filtro passa-baixa a ser utilizado deve ser implementado com a devida atenção, já que para frequências de corte muito baixas, o atraso de fase se torna muito grande, degradando a velocidade de resposta do sistema por completo. Isto levaria a um algoritmo de *Phase-Locked Loop* com estreitíssima banda, ou seja, ótima imunidade a perturbações fora da frequência fundamental, mas altíssimo tempo de acomodação  $t_s$ .

Desta forma, deve-se realizar o projeto do filtro passa-baixa tendo em mente que a frequência de corte deve ser tal que mantenha uma atenuação adequada para a componente de segundo harmônico e o mínimo atraso de fase possível para a frequência de corte  $\omega_c$  escolhida para o sistema em malha aberta.

Tanto [19] quanto [2] explicitam que tanto um sinal *dc* como uma componente de segundo harmônico na entrada  $e_i$  são capazes de produzir uma parcela de componente fundamental na saída do *Phase Detector*, que também necessita ser filtrada. Inclusive, [21] enuncia que cada componente harmônica de ordem  $h$  e amplitude  $V_h$  produz 2 (duas) componentes de ordem  $h \pm 1$  e amplitude  $\frac{V_h}{2}$  na saída do *phase detector*.

Portanto, faz-se necessário e desejável uma pequena parcela de atenuação na frequência fundamental, de forma que oscilações mais severas na frequência e fase estimadas possam ser prevenidas.

Em [2], mostrou-se que o modelo linearizado já contemplando inclusive as perturbações pode ser expresso por:

$$\hat{\theta}(s) = G(s)(\theta(s) + D(s)) \quad (3.4)$$

onde  $D(s)$  é expresso pela perturbação presente na entrada (harmônicos de ordem ímpar) e  $G(s)$  é dado pela função de transferência em malha fechada da Figura 3.3.

Considerando-se a equação (3.3) e (3.4), é possível chegar a função de transferência de malha fechada  $G(s)$ :

$$G(s) = \frac{k_p k_v s + k_i k_v}{\tau_p s^3 + s^2 + k_p k_v s + k_i k_v} \quad (3.5)$$

onde  $\tau_p = \frac{1}{\omega_p}$  é a constante de tempo do filtro passa baixa de primeira ordem do sistema e  $k_v$  é dado como o ganho estático  $\frac{V}{2}$  do *Phase Detector*.

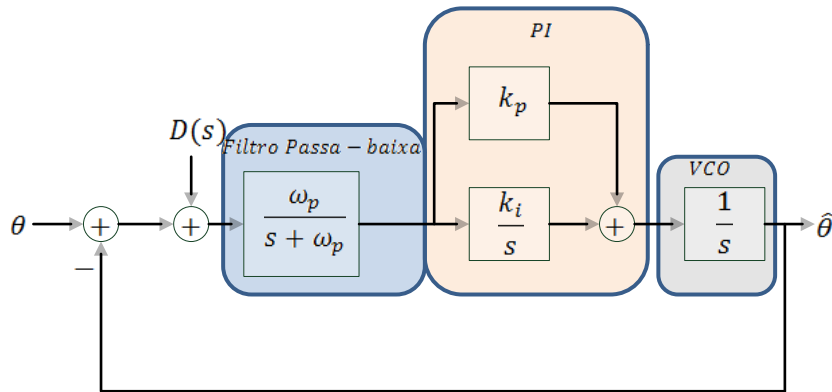


Figura 3.3 – Modelo linearizado do pPLL.

### 3.1.1.1 Estabilidade do sistema

O conceito de estabilidade de um sistema é crucial na síntese de sistemas de controle realimentados. Um sistema é considerado estável caso sua resposta temporal seja limitada para qualquer sinal de entrada limitado.

Para a verificação da estabilidade do sistema, é usual a utilização de um dos métodos abaixo:

- Método de *Routh-Hurwitz*
- Diagrama de *Nyquist*
- Análise temporal

Vale lembrar que nenhum dos métodos acima calcula os pólos da equação característica. O método de *Routh-Hurwitz* (que será usado neste trabalho), por exemplo, realiza a análise de estabilidade a partir dos sinais presentes na equação característica, de modo a verificar se todos os pólos da mesma se encontram no semi-plano esquerdo.

De acordo com [22], o critério de estabilidade de *Routh-Hurwitz* é um método que fornece uma resposta direta sobre a questão de estabilidade de sistemas lineares.

Imaginando a equação característica abaixo, tem-se que:

$$\begin{aligned}\Delta(s) &= a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0 \\ &= a_n (s - r_1)(s - r_2) \dots (s - r_n) = 0\end{aligned}\tag{3.6}$$

onde  $r_i$  é a  $i$ -ésima raiz da equação característica. Para a estabilidade, é necessário verificar se nenhuma destas raízes se encontra no semiplano direito. Expandindo-se a equação (3.6), é possível chegar em:

$$\begin{aligned}\Delta(s) &= a_n s^n - a_n (r_1 + r_2 + \dots + r_n) s^{n-1} \\ &\quad + a_n (r_1 r_2 + r_2 r_3 + \dots) s^{n-2} \\ &\quad - a_n ((r_1 r_2 r_3 + r_1 r_2 r_4 + \dots) s^{n-3} + \dots \\ &\quad + (-1)^n a_n (r_1 r_2 r_3 \dots r_n) = 0\end{aligned}\tag{3.7}$$

Examinando a equação (3.7), pode-se notar que se todas as raízes estiverem no semiplano esquerdo, todos os termos do polinômio tem o mesmo sinal. Além disto, nenhum coeficiente pode ser nulo. É importante ressaltar que essas condições são apenas necessárias [22], e caso não se verifiquem o sistema será instável. No entanto, nada pode ser dito em relação à estabilidade do sistema.

Para isto, o método de *Routh-Hurwitz* utiliza condições necessárias e suficientes para verificar a estabilidade.

Rapidamente, será explicado o método *Routh-Hurwitz* abaixo. O mesmo será utilizado na equação característica dos 3 (três) modelos de *Phase-Locked Loop* enunciados neste capítulo.

Método de Routh-Hurwitz: Para aplicar o método, organizam-se os coeficientes da equação característica na forma de um arranjo do tipo:

$$\begin{array}{c|cccc}
 s^n & a_n & a_{n-2} & a_{n-4} & \dots \\
 s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & \dots
 \end{array}$$

As linhas subsequentes deste arranjo devem ser completadas da forma abaixo:

$$\begin{array}{c|cccc}
 s^n & a_n & a_{n-2} & a_{n-4} & \dots \\
 s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} & \dots \\
 s^{n-2} & b_{n-1} & b_{n-3} & b_{n-5} & \\
 s^{n-3} & c_{n-1} & c_{n-3} & c_{n-5} & \\
 \vdots & \vdots & \vdots & \vdots & \\
 s^0 & d_{n-1} & & & 
 \end{array}$$

sendo que

$$b_{n-1} = \frac{(a_{n-1})(a_{n-2}) - a_n(a_{n-3})}{a_{n-1}} = \frac{-1}{a_{n-1}} \begin{vmatrix} a_n & a_{n-2} \\ a_{n-1} & a_{n-3} \end{vmatrix} \quad (3.8)$$

$$b_{n-3} = \frac{-1}{a_{n-1}} \begin{vmatrix} a_n & a_{n-4} \\ a_{n-1} & a_{n-5} \end{vmatrix} \quad (3.9)$$

$$(3.10)$$

$$c_{n-1} = \frac{-1}{b_{n-1}} \begin{vmatrix} a_{n-1} & a_{n-3} \\ b_{n-1} & b_{n-3} \end{vmatrix}$$

O critério de *Routh-Hurwitz* assegura que o número de raízes com parte real positiva é igual ao número de mudanças de sinais dos elementos da primeira coluna do arranjo. Desta forma, a condição necessária e suficiente para o sistema ser estável é que todos os elementos da primeira coluna tenham o mesmo sinal.

Pode-se considerar 4 (quatro) casos distintos de utilização do critério:

- Nenhum elemento da primeira coluna é nulo.
- Há um zero na primeira coluna, no entanto, alguns elementos na linha onde ocorre o zero não são nulos.
- Há um zero na primeira coluna e todos os elementos na linha onde ocorre o zero também são nulos.
- O mesmo caso anterior, no entanto, com raízes múltiplas no eixo imaginário.

Em seguida, será considerado apenas o primeiro dos casos de utilização do critério de *Routh-Hurwitz*. Este caso é o mesmo encontrado na equação (3.5), como será visto mais adiante.

Caso 1: Nenhum elemento da primeira coluna é nulo:

**Exemplo de sistema de terceira ordem:** Para um sistema de terceira ordem genérico cuja equação característica seja dada por:

$$\Delta(s) = a_3s^3 + a_2s^2 + a_1s + a_0 \quad (3.11)$$

o arranjo de *Routh* será dado como:

$s^3$		$a_3$	$a_1$
$s^2$		$a_2$	$a_0$
$s^1$		$b_1$	0
$s^0$		$c_1$	0



onde os coeficientes  $b_1$  e  $c_1$  são dados por:

$$b_1 = \frac{a_2 a_1 - a_0 a_3}{a_2} \quad (3.12)$$

$$c_1 = \frac{b_1 a_0}{b_1} = a_0 \quad (3.13)$$

Portanto, para que o sistema seja estável, todos os coeficientes devem ser positivos e  $a_2 a_1 > a_0 a_3$ . Caso  $a_2 a_1 = a_0 a_3$ , o sistema terá um par de raízes no eixo imaginário, resultando em estabilidade marginal [22].

Aplicando-se o método anterior à equação (3.5), pode-se chegar à conclusão de que o sistema será estável para:

$$0 < k_i < k_p \omega_p \quad (3.14)$$

Esta é a primeira verificação a ser feita em todos os modelos de PLL a serem analisados neste capítulo. A partir de (3.14), é possível saber até onde é plausível excursionar os ganhos do controlador PI com segurança e dentro da margem de estabilidade.

Neste trabalho, serão utilizadas 2 (dois) requisitos para a implementação de cada malha de PLL. Os requisitos são:

- Margem de fase adequada [14].
  - $30^\circ < \phi_m < 80^\circ$
- Atenuação de  $-20 \text{ db}$  para uma perturbação de segundo harmônico na entrada.

A partir destes requisitos, será possível a obtenção dos ganhos proporcional ( $k_p$ ) e integral ( $k_i$ ) do controlador PI, bem como a constante de tempo  $\tau_p$  do filtro de primeira ordem utilizado na saída do *Phase Detector*.

A seguir, será realizada uma breve introdução à metodologia proposta acima.

Margem de estabilidade:

Para a obtenção de conclusões sobre a estabilidade absoluta e relativa de um SLIT (Sistema Linear Invariante no Tempo) são necessárias análises provenientes diretamente da resposta em frequência do sistema, através dos métodos de:

- Diagrama de *Bode*
- Diagrama de *Nyquist*

A interpretação da resposta em frequência da função de transferência de malha aberta do sistema traz informações fundamentais para a análise e projeto de um sistema realimentado [23].

O critério de estabilidade de *Nyquist* relaciona a estabilidade de um sistema de controle realimentado em malha fechada através da resposta em frequência do mesmo sistema operando em malha aberta [13]. Este método pode ser utilizado para a avaliação da estabilidade relativa do sistema.

A partir do Diagrama de *Nyquist*, observa-se a proximidade da curva de resposta em frequência do sistema operando em malha aberta com o ponto  $-1.0 + 0.0j$  no plano  $G(j\omega)H(j\omega)$ .

A proximidade da curva de resposta em frequência do sistema operando em malha aberta ao ponto em questão será avaliada em módulo e fase através de 2 (duas) medidas definidas como *margem de fase* e *margem de ganho*.

A margem de ganho de  $G_m$  é a faixa que se pode incrementar ou decrementar a curva de resposta em frequência de módulo da função de transferência de malha aberta de um sistema até que se alcance o ponto de estabilidade crítica [23].

A margem de fase  $\phi_m$  é o valor angular a ser acrescido ou decrescido a curva de fase da resposta em frequência de um sistema operando em malha aberta na frequência em que a curva de módulo da resposta em frequência deste mesmo sistema apresenta valor unitário (0.0 db).

$$\phi_m = 180^\circ + \arg\{G(j\omega_{0db})H(j\omega_{0db})\} \quad (3.15)$$

onde  $\omega_{0db}$  é a frequência em que o módulo de  $G(j\omega)H(j\omega)$  é igual a 1 (0.0 db).

### 3.1.1.2 Sintonia dos controladores

Para a sintonia ótima do controlador PI e do filtro passa-baixa, utilizou-se a teoria prevista em [2, 24] aliada aos princípios analisados na seção 3.1.1.1.

Para a análise de resposta em frequência do  $pPLL$ , faz-se necessário considerar sua função de transferência em malha aberta, dada por:

$$G_{ol}(s) = \frac{G(s)}{1 - G(s)} = \frac{k_p \omega_p k_v (s + \omega_z)}{s^2 (s + \omega_p)} \quad (3.16)$$

onde  $\omega_z = k_i/k_p$ .

A margem de fase deste sistema pode ser calculada a partir de (3.15), sendo dada por:

$$\phi_m = \arctg\left(\frac{\omega_{0db}}{\omega_p}\right) - \arctg\left(\frac{\omega_{0db}}{\omega_z}\right) \quad (3.17)$$

De modo a cumprir os requisitos estabelecidos na seção 3.1.1.1, deve-se verificar para quais valores de  $\omega_{0db}$ ,  $\omega_p$  e  $\omega_z$  a margem de fase do sistema é máxima. Isto é realizado derivando-se a equação (3.17) em relação à  $\omega_{0db}$  e igualando-a a zero, como previsto por [24]. Desta forma, é possível chegar às relações a seguir:

$$\omega_{0db} = \sqrt{\omega_p \omega_z} \quad (3.18)$$

$$\omega_{0db} = k_p \quad (3.19)$$

Ou seja, de modo a estabelecer as 2 (duas) relações propostas na seção 3.1.1.1, utiliza-se as equações (3.18) e (3.19) na função de transferência em malha aberta (3.16). Desta forma, chega-se a relação abaixo:

$$G_{ol}(s) = \frac{k_p \omega_p k_v \left(s + \frac{k_p}{\omega_p}\right)}{s^2 (s + \omega_p)} \quad (3.20)$$

A função de transferência em malha aberta do sistema agora se encontra em função do polo do filtro passa-baixa. Deste modo, é possível obter o valor de  $\omega_p$  de

acordo com a atenuação proposta de  $-20 \text{ db}$ . Lembrando que  $-20 \text{ db}$  equivale a 90% de atenuação, é possível chegar a seguinte equação:

$$|G_{ol}(s)| = \left| \frac{k_p \omega_p k_v \left( s + \frac{k_p}{\omega_p} \right)}{s^2 (s + \omega_p)} \right|_{s=j\omega_{2h}} \quad (3.21)$$

Reorganizando (3.21), chega-se a:

$$|G_{ol}(s)|^2 - \left[ \frac{(k_p \omega_p k_v \omega_{2h})^2 + (k_p^3 k_v)^2}{(\omega_{2h} \omega_p)^2 + (\omega_{2h}^2)^2} \right] = 0 \quad (3.22)$$

De modo a obter-se o valor de  $\omega_p$ , faz-se necessário o uso de uma ferramenta matemática para resolução de polinômios de ordem mais alta. Foi utilizado, portanto, o *software* Matlab para o cálculo de  $\omega_p$ . Tomando  $|G_{ol}(s)| = 0.1$ ,  $k_v = 1.0$ ,  $k_p = 150$  e  $\omega_{2h} = 753.96 \text{ rad/s}$ , onde  $\omega_{2h}$  é a frequência angular para o segundo harmônico, utilizou-se o algoritmo apresentado no Anexo 4 para a obtenção dos valores de  $\omega_p$ ,  $k_i$  e  $\omega_z$ . Os valores encontrados foram os seguintes:

$$\omega_p = 437.01 \frac{\text{rad}}{\text{s}}, k_i = 7722.92 \text{ s}^{-1}, \omega_z = 51.48 \frac{\text{rad}}{\text{s}}$$

Através da função de transferência (3.5), é possível estimar o valor de  $t_s$ . Para isto, utiliza-se a aproximação de segunda ordem (I.16). Isto pode ser realizado levando-se em consideração que o coeficiente do termo de ordem 3 (três) do denominador de (3.5) é muito menor do que os outros coeficientes do denominador.

$$t_s = \frac{8}{k_p k_v} = 0.053 \text{ s}$$

É possível validar a atenuação proposta na seção 3.1.1.1 na Figura 3.4. Para a resposta ao degrau, o tempo  $t_s$  estimado acima se mostrou concreto como pode ser analisado a partir da Figura 3.4.

Para a implementação do algoritmo de *pPLL* (a ser tratado no presente trabalho por “digital”), utilizou-se a ferramenta Simulink do pacote MATLAB para a simulação da planta apresentada na Figura 3.2. Os mesmos parâmetros encontrados a partir do modelo linearizado foram utilizados no modelo digital. A entrada foi um seno com

amplitude unitária. De modo a verificar a validade do modelo linearizado do *pPLL*, comparou-se na Figura 3.5 o erro do controlador PI no modelo linearizado (para uma entrada de degrau unitário) com o erro do algoritmo digital implementado. O comportamento obtido pelo modelo linearizado foi bastante próximo ao do modelo digital, excetuando-se as oscilações em torno do ponto de operação.

De modo a verificar se a atenuação do sinal de saída respeita o nível apresentado na seção 3.1.1.1, introduziu-se a seguir um sinal de 2° harmônico com amplitude também unitária a entrada original. Lembrando que uma atenuação de  $-20\text{ db}$  representa uma rejeição de 90% do sinal de entrada.

O resposta do *pPLL* a esta entrada, bem como o espectro harmônico do sinal sincronizado de saída podem ser analisados nas Figura 3.6. É possível verificar que a atenuação do 2° harmônico foi muito próxima ao requisito de 90% proposto anteriormente.

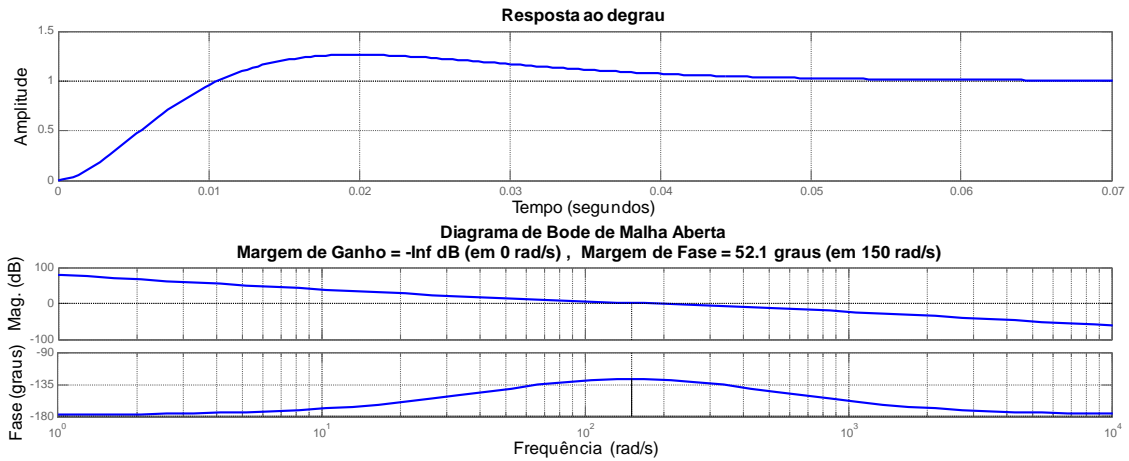


Figura 3.4 – Resposta ao degrau e diagrama de bode em malha aberta do pPLL.

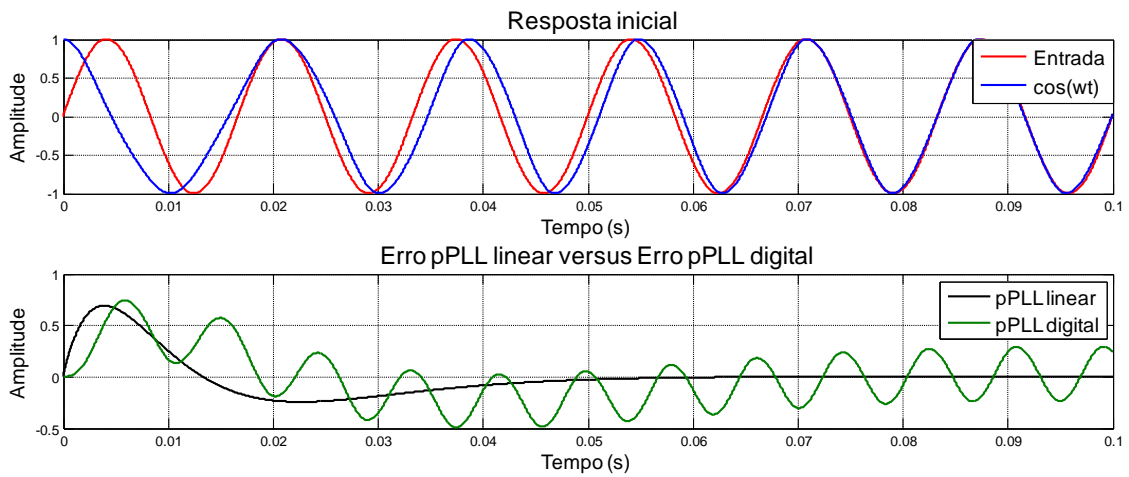


Figura 3.5 – Resposta inicial do modelo digital e erro do controlador PI do pPLL.

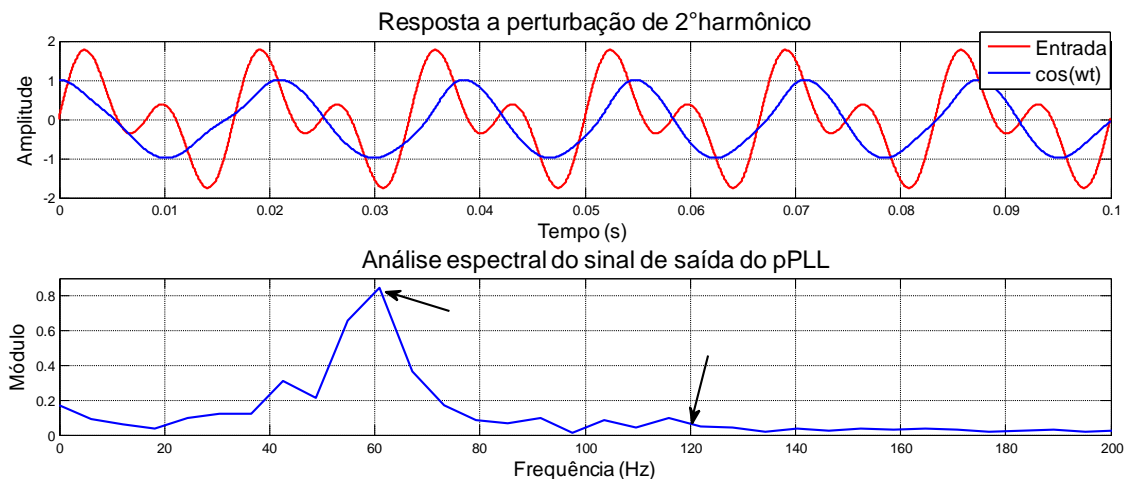


Figura 3.6 – Comportamento do modelo real para entrada unitária de fundamental e 2º harmônico.

### 3.1.2 ParkPLL

O diagrama de blocos da estrutura do *parkPLL* [3] pode ser analisada na Figura 3.7. Este PLL é uma versão monofásica do algoritmo proposto em [20]. A componente  $v_\beta$  do eixo estacionário é obtida a partir da transformada inversa de *Park* dos sinais síncronos já filtrados  $v_d'$  e  $v_q'$ . Desta forma, um sistema trifásico balanceado pode ser emulado [19]. O comportamento dinâmico do *phase detector* do *parkPLL* é determinado pelas constantes de tempo  $\tau_q$  e  $\tau_d$  dos filtros de primeira ordem. O ganho estático do *phase detector* será analisado a seguir.

As transformações efetuadas no *parkPLL* são:

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} \text{sen}\hat{\theta} & \text{cos}\hat{\theta} \\ \text{cos}\hat{\theta} & -\text{sen}\hat{\theta} \end{bmatrix} \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} \quad (3.23)$$

$$\begin{bmatrix} v_\alpha' \\ v_\beta' \end{bmatrix} = \begin{bmatrix} \text{sen}\hat{\theta} & \text{cos}\hat{\theta} \\ \text{cos}\hat{\theta} & -\text{sen}\hat{\theta} \end{bmatrix} \begin{bmatrix} v_d' \\ v_q' \end{bmatrix} \quad (3.24)$$

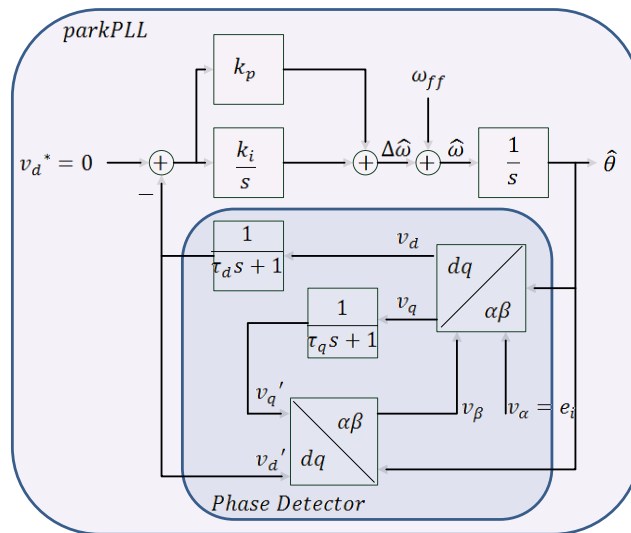


Figura 3.7 – *parkPLL* monofásico. Adaptado de [19].

As componentes filtradas  $v_d'$  e  $v_q'$  podem ser analisadas como:

$$v_d'(s) = \frac{v_d(s)}{\tau_d s + 1} \quad (3.25)$$

$$v_q'(s) = \frac{v_q(s)}{\tau_q s + 1} \quad (3.26)$$

Através da manipulação matemática de (3.23)-(3.26), obtêm-se a equação abaixo, a qual descreve o comportamento do *phase detector* no eixo de referência girante. Esta expressão representa um sistema linear variante no tempo, já que a matriz de estados e vetor de ganhos são ambos funções de  $\theta(t)$ .

$$\frac{d}{dt} \begin{bmatrix} v_d' \\ v_q' \end{bmatrix} = \begin{bmatrix} -\frac{\text{sen}^2 \hat{\theta}}{\tau_d} & -\frac{\text{sen} \hat{\theta} \cos \hat{\theta}}{\tau_d} \\ -\frac{\text{sen} \hat{\theta} \cos \hat{\theta}}{\tau_q} & -\frac{\text{cos}^2 \hat{\theta}}{\tau_q} \end{bmatrix} \begin{bmatrix} v_d' \\ v_q' \end{bmatrix} + \begin{bmatrix} \frac{\text{sen} \hat{\theta}}{\tau_d} \\ \frac{\text{cos} \hat{\theta}}{\tau_q} \end{bmatrix} v_\alpha \quad (3.27)$$

Levando-se em consideração que  $\hat{\theta} = \hat{\omega}t + \hat{\phi}$ ,  $v_\alpha = V \cos(\omega t + \phi)$  e aproximando-se  $\hat{\omega} \cong \omega$ , o ponto de equilíbrio do sistema pode ser encontrado zerando os termos derivativos [19] em (3.27), resolvendo-se então  $v_d'$  e  $v_q'$ , o que garante que:

$$\bar{V}_d' = V \text{sen} \phi_e \quad (3.28)$$

$$\bar{V}_q' = V \text{cos} \phi_e \quad (3.29)$$

onde  $\phi_e = \hat{\phi} - \phi$ ,  $\bar{V}_d'$  e  $\bar{V}_q'$  são os valores de saída de estado estacionário do *phase detector*.

A equação (3.28), revela que, caso a componente  $v_d'$  seja regulada em zero,  $\phi_e$  também será zero. Nesta situação, a equação (3.29) mostra que  $\bar{V}_q'$  é igual à amplitude do sinal de entrada. A equação diferencial que rege as variáveis do eixo estacionário  $v_\alpha'$  e  $v_\beta'$  é dada por:



$$\frac{d}{dt} \begin{bmatrix} v_{\alpha}' \\ v_{\beta}' \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau} & \frac{d\hat{\theta}}{dt} \\ -\frac{d\hat{\theta}}{dt} & 0 \end{bmatrix} \begin{bmatrix} v_{\alpha}' \\ v_{\beta}' \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} v_{\alpha} \quad (3.30)$$

onde  $\tau = \tau_d = \tau_q$  é dada como a constante de tempo dos filtros passa-baixa. Como na seção anterior, reconhecendo  $\frac{d\hat{\theta}}{dt} = \hat{\omega}$  e fazendo  $\hat{\omega} \cong \omega$ , a equação (3.30) torna-se um sistema linear invariante no tempo com excitação senoidal [19], cuja equação característica é dada por  $\det(\lambda I - A) = 0$  ou:

$$\lambda^2 + \frac{\lambda}{\tau} + \hat{\omega}^2 = 0 \quad (3.31)$$

Os autovalores acima dependem diretamente de  $\tau$  e  $\hat{\omega}$  de acordo com:

$$\lambda_{1,2} = -\frac{1}{2\tau} \pm \frac{1}{2} \sqrt{\frac{1}{\tau^2} - 4\hat{\omega}^2} \quad (3.32)$$

A equação (3.32) mostra que o *phase detector* encontra-se sempre assintoticamente estável em torno do ponto de equilíbrio  $\hat{\omega} \cong \omega$ . Em caso de  $\tau^{-1} \gg 2\hat{\omega}$ , existirá um par de raízes reais. Um destes polos será  $\lambda_1 \approx \tau^{-1}$ , que gera uma dinâmica rápida. No entanto,  $\lambda_2$  será aproximadamente zero, dominando o sistema com sua dinâmica extremamente lenta. Por outro lado, caso  $\tau^{-1} \ll 2\hat{\omega}$ , um par de polos complexos com pequena parcela real aparecerá. Isto não é algo desejável, já que tornará a dinâmica lenta e oscilatória.

Após a aplicação da transformada de *Park*, tem-se que:

$$v_d' = v_{\alpha}' \sin \hat{\theta} + v_{\beta}' \cos \hat{\theta} \quad (3.33)$$

Considerando-se que os termos oscilantes devido à excitação senoidal decaem a zero de acordo com o amortecimento relativo à parcela real presente em (3.32), a função de transferência da saída do *phase detector* em relação a uma variação de fase é dada por:

$$F(s) = \frac{v_d'(s)}{\phi_e(s)} \cong \frac{\bar{V}_d'}{2\tau s + 1} \quad (3.34)$$

Desta forma, para variações de fase  $\hat{\phi} - \phi$  a função de transferência de malha fechada do sistema da Figura 3.7 já linearizado pode ser dado por:

$$\frac{\hat{\phi}(s)}{\phi(s)} \cong k_v \frac{sk_p + k_i}{2\tau s^3 + s^2 + sk_vk_p + k_vk_i} \quad (3.35)$$

### 3.1.2.1 Estabilidade do sistema

Assim como realizado na seção 3.1.1.1, faz-se necessário o estudo da estabilidade do *parkPLL*, de modo que os parâmetros do mesmo possam ser escolhidos corretamente, contemplando os requisitos propostos na seção supracitada.

A primeira etapa desta seção trata-se justamente da verificação da margem de estabilidade da função de transferência de malha fechada da equação (3.35). Para tal, pode-se aplicar o critério de *Routh-Hurwitz*. A equação característica de (3.35) respeita o modelo apresentado em (3.11). Desta forma, utilizam-se as equações (3.12) e (3.13), como mostrado a seguir.

$s^3$	$2\tau$	$k_vk_p$
$s^2$	$1$	$k_vk_i$
$s^1$	$b_1$	$0$
$s^0$	$c_1$	$0$

onde  $b_1 = \frac{k_vk_p - 2\tau k_vk_i}{k_vk_p}$  e  $c_1 = 2\tau$ .

Portanto, para que o sistema seja estável, todos os coeficientes devem ser positivos e  $a_2a_1 > a_0a_3$ . Caso  $a_2a_1 = a_0a_3$ , o sistema terá um par de raízes no eixo imaginário, resultando em estabilidade marginal [22].

Aplicando-se o método anterior à equação (3.35), pode-se chegar à conclusão de que o sistema será estável para:

$$0 < k_i < \frac{k_p}{2\tau} \quad (3.36)$$

Assim como no caso do *pPLL*, esta deve ser a primeira verificação a ser feita em no *parkPLL*. A partir de (3.36), será possível saber até onde é plausível excursionar os ganhos do controlador PI com segurança e dentro da margem de estabilidade.

Novamente, serão utilizadas os 2 (dois) requisitos para a implementação da malha do *parkPLL*. Os requisitos são:

- Margem de fase adequada [14].
  - $30^\circ < \phi_m < 80^\circ$
- Atenuação de -20 *db* para uma perturbação de segundo harmônico na entrada.

A partir destes requisitos, será possível a obtenção dos ganhos proporcional ( $k_p$ ) e integral ( $k_i$ ) do controlador PI, bem como a constante de tempo  $\tau$  dos filtros de primeira ordem utilizados no *phase detector*.

### 3.1.2.2 Sintonia dos controladores

Para a sintonia ótima do controlador PI e dos filtros passa-baixas, utilizou-se novamente a teoria prevista em [2, 24] aliada aos princípios já analisados na seção 3.1.1.1.

Para a análise de resposta em frequência do *parkPLL*, faz-se necessário considerar sua função de transferência em malha aberta, dada novamente pela equação (3.16). No entanto, desta vez  $\omega_z = k_i/k_p$  e  $\omega_p = \frac{1}{2\tau}$ .

De modo a cumprir os requisitos estabelecidos na seção 3.1.1.1, deve-se verificar para que valores de  $\omega_{0db}$ ,  $\omega_p$  e  $\omega_z$  a margem de fase do sistema é máxima. Isto é realizado utilizando as equações (3.18) e (3.19) na equação (3.22).

Novamente, para o cálculo do valor de  $\omega_p$ , faz-se necessário o uso do *software* Matlab. Tomando  $|G_{o1}(s)| = 0.1$ ,  $k_v = 1.0$ ,  $k_p = 150$  e  $\omega_{2h} = 753.96$  rad/s, onde  $\omega_{2h}$  é a frequência angular para o segundo harmônico, utilizou-se o algoritmo apresentado no Anexo 4 obtendo-se os valores de  $\omega_p$ ,  $k_i$  e  $\omega_z$  e  $\tau$ . Os valores encontrados foram os seguintes:

$$\omega_p = 437.01 \frac{\text{rad}}{\text{s}}, k_i = 7722.92 \text{ s}^{-1}, \omega_z = 51.48 \frac{\text{rad}}{\text{s}}, \tau = 0.001144\text{s}$$

Através da função de transferência (3.35), é possível estimar o valor de  $t_s$ . Para isto, utiliza-se a aproximação de segunda ordem (I.16). Isto pode ser realizado levando-se em consideração que o coeficiente do termo de ordem 3 (três) do denominador de (3.35) é muito menor do que os outros coeficientes do denominador.

$$t_s = \frac{8}{k_p k_v} = 0.053\text{s}$$

Para a resposta ao degrau, o tempo  $t_s$  estimado acima mostrou-se concreto na Figura 3.8. Por observação, é possível verificar que, como os modelos linearizados de ambos os PLLs já apresentados (*pPLL* e *parkPLL*) apresentam funções de transferência muito parecidas, basicamente a resposta de ambos (tanto na frequência quanto ao degrau) foi exatamente a mesma para o requisitos impostos. No entanto, o valor da constante de tempo  $\tau$  difere-se de  $\tau_p$ , devido à diferença entre as topologias digitais de ambos os PLLs.

Para a implementação do modelo digital, utilizou-se novamente a ferramenta Simulink com a planta apresentada na Figura 3.7. Os mesmos parâmetros encontrados a partir do modelo linearizado foram utilizados no modelo digital. Comparou-se novamente o erro do controlador PI no modelo linearizado com o obtido para modelo digital, como analisado na Figura 3.9.

É possível observar a partir da Figura 3.9 que em  $t \cong 0.060\text{s}$  ocorreu o sincronismo do sinal de saída do *parkPLL* com o sinal de entrada  $v_\alpha$ .

De modo a verificar se a atenuação do sinal de saída respeita o nível apresentado na seção 3.1.1.1, introduziu-se a seguir um sinal de 2º harmônico com amplitude também unitária a entrada original. Vale lembrar que uma atenuação de  $-20 \text{ db}$  representa uma rejeição de 90% do sinal de entrada.

A resposta do *parkPLL* a esta entrada, bem como o espectro harmônico do sinal sincronizado de saída podem ser analisados nas Figura 3.10. A partir da análise destes resultados, é possível ver que o modelo linearizado previu com razoável precisão tanto o tempo de assentamento do modelo digital do *parkPLL* quanto sua capacidade de atenuação harmônica.

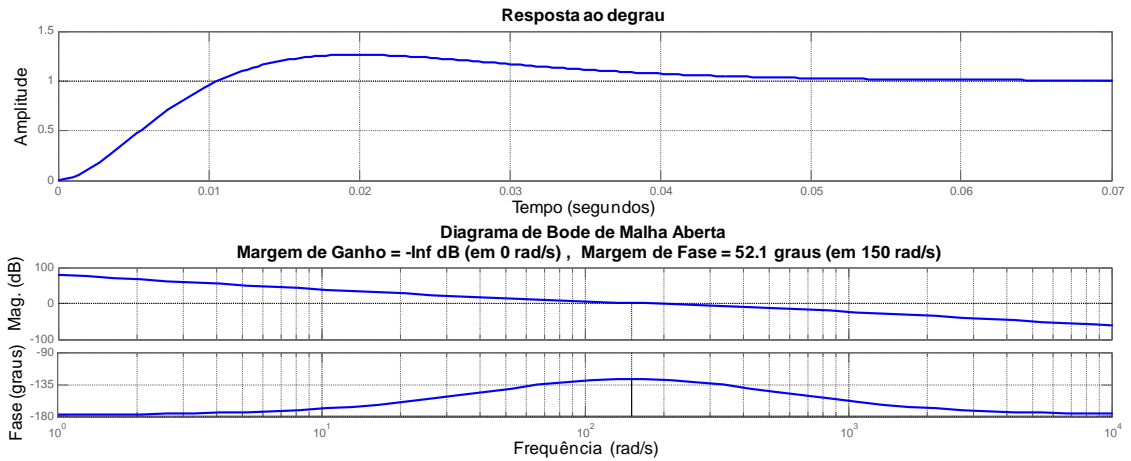


Figura 3.8 – Resposta ao degrau e diagrama de bode em malha aberta do parkPLL.

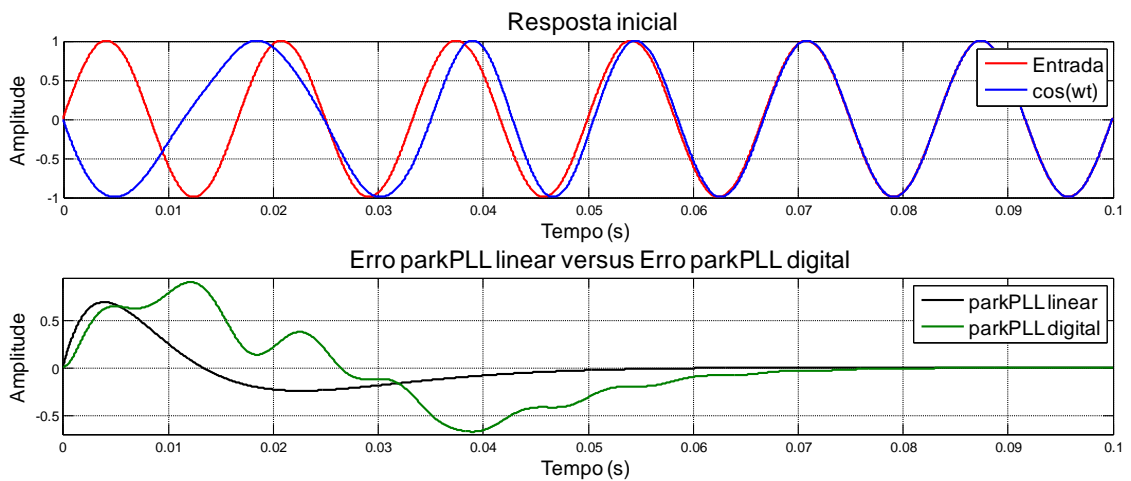


Figura 3.9 – Resposta inicial do modelo digital e erro do controlador PI do parkPLL.

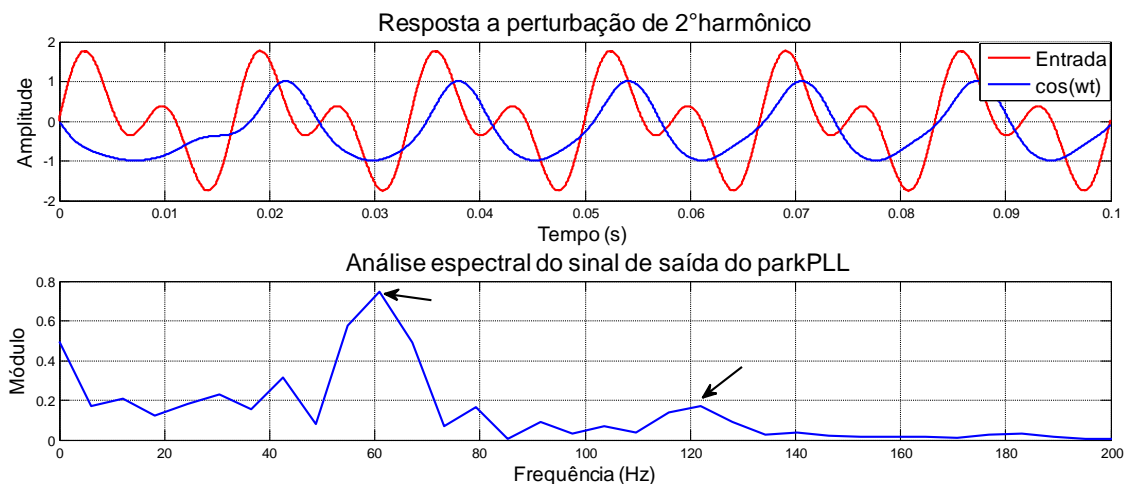


Figura 3.10 - Comportamento do modelo real do parkPLL para entrada unitária de fundamental e 2º harmônico.

### 3.1.3 Enhanced PLL (EPLL)

O *Enhanced PLL* (EPLL) é baseado na teoria de filtro adaptativo proposta em [16, 5]. Basicamente, este algoritmo reconstrói em tempo real a componente fundamental do sinal de entrada a partir da estimação de sua amplitude, fase e frequência. O ganho  $K$  controla a velocidade de convergência de  $\hat{A}$ , que é a amplitude estimada do sinal de entrada (Figura 3.11).

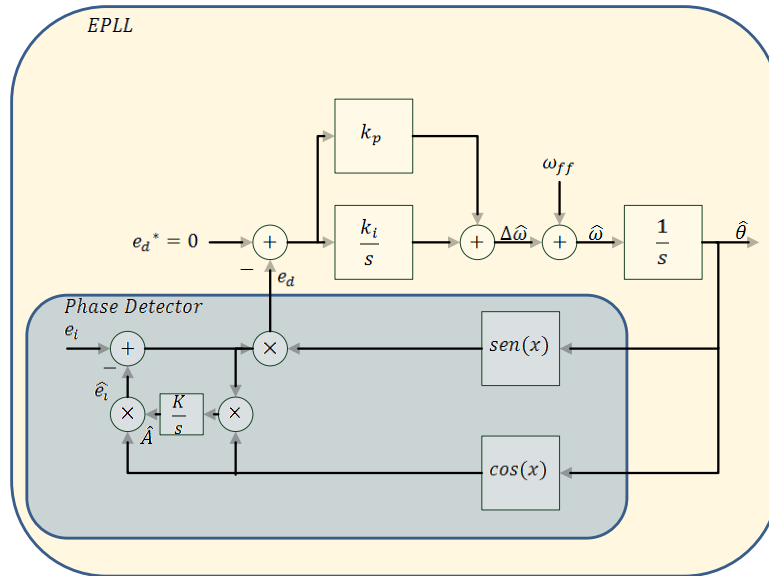


Figura 3.11 – EPLL monofásico.

Supondo novamente uma entrada  $e_i$  puramente senoidal, tal que  $e_i = V\cos\theta$ , o ganho estático do *phase detector* pode ser encontrado a partir da expressão da saída  $e_d$  como função de  $\theta$ ,  $\hat{\theta}$  e  $V$ :

$$e_d = \frac{V}{2}\text{sen}(\hat{\theta} - \theta) + \frac{V}{2}\text{sen}(\hat{\theta} + \theta) - \frac{\hat{A}}{2}\text{sen}(2\hat{\theta}) \quad (3.37)$$

Como anteriormente, assume-se que  $\hat{\theta} = \hat{\omega}t + \hat{\phi}$ ,  $\theta = \omega t + \phi$ ,  $\hat{\omega} \cong \omega$ . Além disto, quando  $\hat{A}$  converge para  $\hat{V}$ , o termo oscilatório  $\frac{V}{2}\text{sen}(\hat{\theta} + \theta)$  se torna nulo. Portanto, pode-se aproximar a equação (3.37) pela equação abaixo:

$$e_d \cong \frac{V}{2}(\hat{\phi} - \phi) \quad (3.38)$$

Utilizando-se da equação (3.38) no diagrama de blocos da Figura 3.11, é possível chegar à função de transferência em malha fechada do EPLL linearizado.

$$\frac{\hat{\phi}(s)}{\phi(s)} = k_v \frac{sk_p + k_i}{s^2 + sk_vk_p + k_vk_i} \quad (3.39)$$

Baseado na equação (3.39), é possível obter os ganhos  $k_p$  e  $k_i$ . É importantíssimo seguir as especificações da seção 3.1.1.1, de modo que o PLL apresente boa dinâmica e capacidade de rejeição a perturbações.

### 3.1.3.1 Estabilidade do sistema

Faz-se necessário nesta seção o estudo da estabilidade do *EPLL*, de modo que os parâmetros do mesmo possam ser escolhidos corretamente, contemplando os requisitos propostos na seção supracitada.

A primeira etapa desta seção trata-se justamente da verificação da margem de estabilidade da função de transferência de malha fechada da equação (3.39). Para tal, aplica-se novamente o critério de *Routh-Hurwitz*. No entanto, para este caso, utiliza-se o critério para uma função de segunda ordem do tipo:

$$\Delta(s) = a_2s^2 + a_1s + a_0 \quad (3.40)$$

O critério de *Routh-Hurwitz* é dado a seguir:

$s^2$		$a_2$	$a_0$
$s^1$		$a_1$	$0$
$s^0$		$b_1$	$0$

onde  $b_1 = \frac{a_1a_0 - a_2(0)}{a_1}$ .

Portanto, para que o sistema seja estável, basta que todos os coeficientes da equação característica do sistema (3.39) sejam todos positivos ou todos negativos [22].

Levando em consideração que  $k_v, k_i$  e  $k_p$  sempre terão valor positivo, pode-se dizer teoricamente que o sistema (3.39) será sempre estável.

### 3.1.3.2 Sintonia dos controladores

Para a sintonia ótica do controlador PI, utilizou-se uma teoria ligeiramente diferente da teoria utilizada anteriormente [2, 24] aliada aos princípios já analisados na seção 3.1.1.1.

De modo a determinar a expressão matemática da margem de fase deste sistema, deve-se primeiro estabelecer a função de transferência de malha aberta do sistema de segunda ordem da equação (I.16):

$$G_{ol}(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s} \quad (3.41)$$

Para encontrar o módulo de (3.41) em função de  $\omega_{0db}$ , realiza-se o equacionamento abaixo:

$$|G(s)| = \left| \frac{\omega_n^2}{s(s + 2\xi\omega_n)} \right| = 1 \quad (3.42)$$

$$|G(s)| = \frac{\omega_n^2}{\sqrt{\omega^4 + 4\xi^2\omega_n^2\omega^2}} = 1 \quad (3.43)$$

Portanto, fazendo-se  $|G(s)|_{s=j\omega_{0db}}$ , é possível obter a frequência  $\omega_{0db}$  que satisfaz a equação anterior:

$$\omega_{0db} = \omega_n \sqrt{-2\xi^2 + \sqrt{4\xi^4 + 1}} \quad (3.44)$$

Deste modo, após determinada  $\omega_{0db}$  e de posse de (3.15), é possível estabelecer a relação entre a margem de fase  $\phi_m$  e os parâmetros de (3.42).

$$\phi_m = 180^\circ - \left( 90^\circ + \arctg \left( \frac{\omega_{0db}}{2\xi\omega_n} \right) \right) \quad (3.45)$$

$$\phi_m = 90 - \arctg \frac{\sqrt{-2\xi^2 + \sqrt{4\xi^4 + 1}}}{2\xi} \quad (3.46)$$



Assim, pode-se utilizar a relação acima para o cálculo dos parâmetros  $k_p$  e  $k_i$  do *EPLL* baseado em uma margem de fase já previamente escolhida. De maneira a contemplar o requisito referente à margem de fase da seção 3.1.1.1.

Escolhendo-se o tempo de assentamento  $t_s = 0.053s$  e uma margem de fase de  $\phi_m = 52.1^\circ$ , tal como os obtidos para o *pPLL* e *parkPLL*, é possível através das equações (I.25) e (3.46) chegar aos valores de amortecimento  $\xi$  e frequência natural de oscilação  $\omega_n$ :

$$\xi = 0.5033, \omega_n = 149.95 \text{ rad/s}$$

De posse destes valores e das equações (I.16) e (3.39), obtêm-se os seguintes valores de  $k_p$  e  $k_i$  para  $k_v = 1.0$ :

$$k_p = \frac{2\xi\omega_n}{k_v} \quad (3.47)$$

$$k_i = \frac{\omega_n^2}{k_v} \quad (3.48)$$

$$k_p \cong 150.93 \text{ e } k_i \cong 22485 \text{ s}^{-1}$$

Para a análise de resposta em frequência do *EPLL*, faz-se necessário considerar sua função de transferência em malha aberta. Levando em consideração que  $G(s)$  é a função de transferência em malha fechada presente em (3.39), é possível estabelecer então que:

$$G_{ol}(s) = \frac{G(s)}{1 - G(s)} = \frac{k_v k_p s + k_v k_i}{s^2} \quad (3.49)$$

Todos os testes verificados a seguir seguem os padrões estabelecidos nas seções 3.1.1.2 e 3.1.2.2. É possível novamente verificar pelas Figura 3.12, Figura 3.13 e Figura 3.14 que a capacidade de atenuação e tempo de assentamento obtidos para o *EPLL* digital foram estimados com precisão pelo seu modelo linearizado.

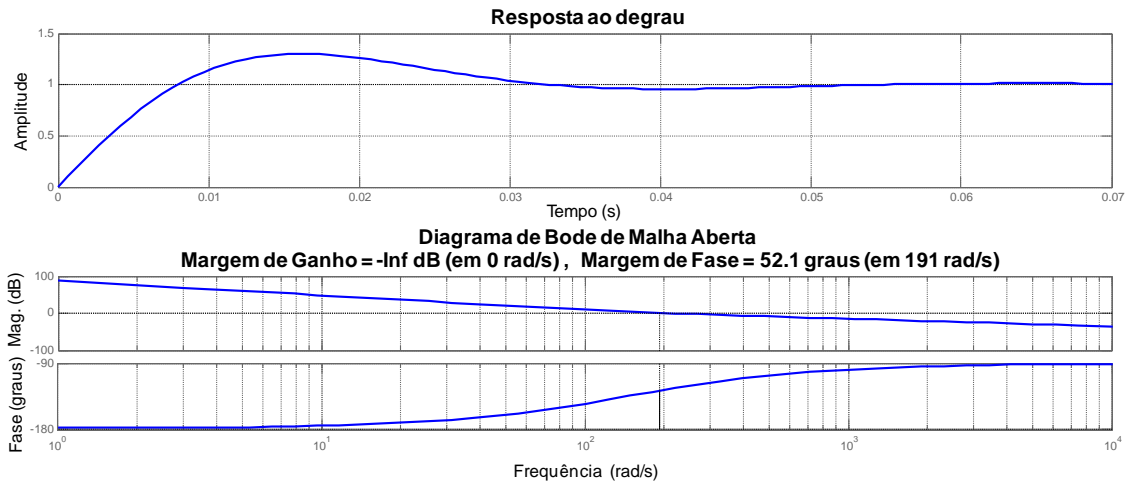


Figura 3.12 - Resposta ao degrau e diagrama de bode em malha aberta do EPLL.

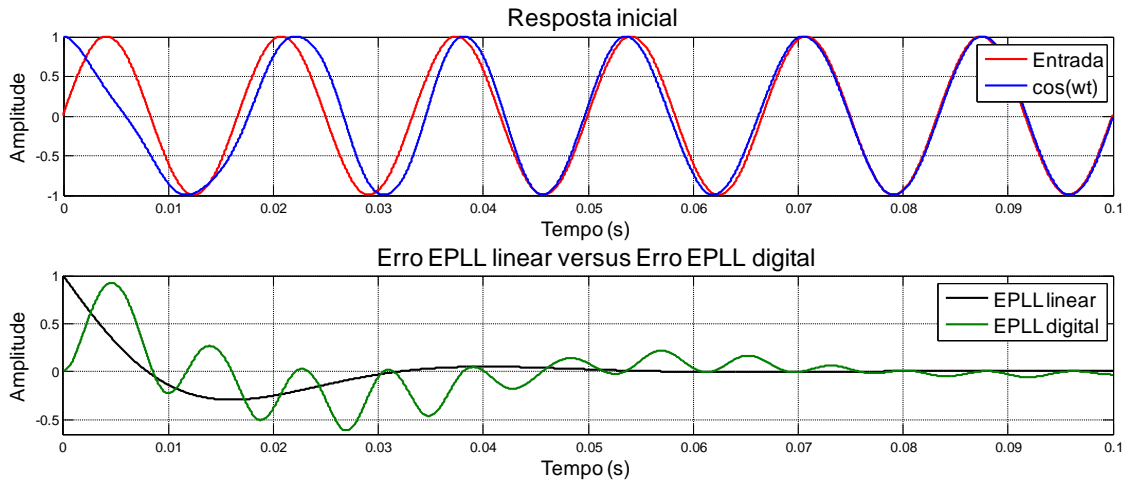


Figura 3.13 – Resposta inicial do modelo digital e erro do controlador PI do EPLL.

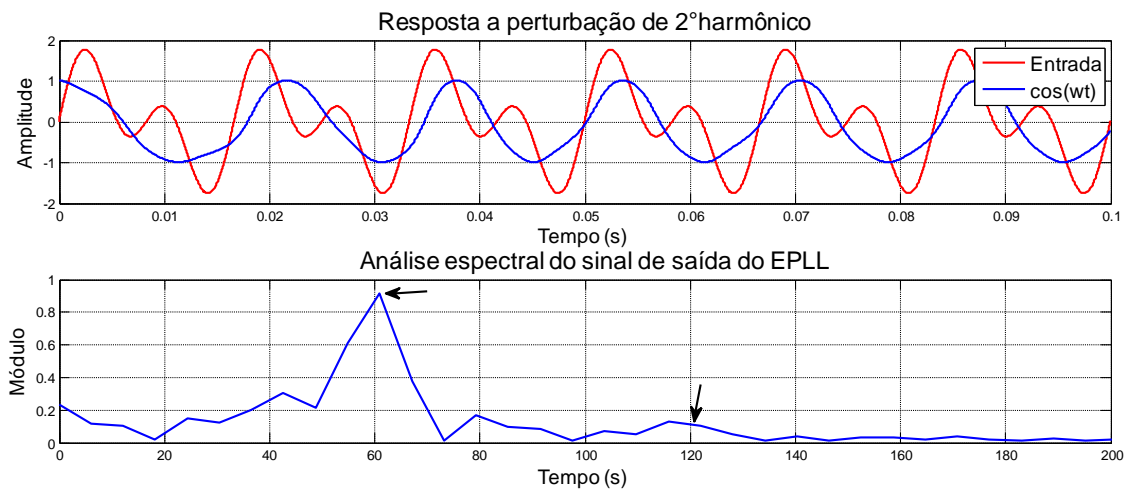


Figura 3.14 – Comportamento do modelo real do EPLL para entrada unitária de fundamental e 2º harmônico.

## 4. Análise de desempenho do $pPLL$ , $parkPLL$ e $EPLL$

No Capítulo 3, foi apresentada a modelagem matemática dos algoritmos  $pPLL$ ,  $parkPLL$  e  $EPLL$ . Apresentou-se a linearização dos algoritmos supracitados, comparando-as com os seus modelos digitais. A proximidade dos resultados obtidos garante, portanto, que a utilização de modelos linearizados para a estratégia de sintonia dos controladores e análise de desempenho do modelo digital de *Phase-Locked Loop* é bastante eficaz.

No presente capítulo, serão apresentados os seguintes tópicos:

- Estratégia de digitalização das malhas de  $PLL$  (linear e digital).
- Resposta temporal dos modelos a variações no sinal de entrada, tais como: degraus de fase, injeção de harmônicos e presença de sub-harmônicos.
- Comparação dos resultados obtidos para cada  $PLL$  analisado.
- Conclusão sobre os resultados.

### 4.1 Digitalização

Toda a análise de sintonia dos controladores feita na seção 3.1.1.2 foi realizada no domínio contínuo da frequência. Desta forma, de modo a possibilitar o embarque dos algoritmos em microprocessadores, faz-se necessário discretizar os modelos contínuos. A transformada  $Z$  é bastante utilizada para a análise em tempo discreto. É comumente utilizada no processamento digital de sinais, por exemplo, para a obtenção do comportamento de sinais digitalizados e para a criação de filtros digitais [25].

#### 4.1.1 Transformada $Z$

A transformada  $Z$  é responsável pela mudança do domínio do tempo discreto para o domínio da variável  $z$ . É uma alternativa mais adequada que a transformada de *Laplace* quando a variável tempo não é contínua.

O evento da digitalização pode ser analisado da seguinte maneira:

Imagina-se um sinal no domínio discreto  $n$  do tempo  $g[n]$ , onde  $g[n]$  é uma sequência de números reais ordenados segundo valores crescentes de  $n$ . Em casos

práticos,  $g[n]$  é obtida através da amostragem de  $g(t)$  a partir de intervalos regulares  $T_s$  conhecidos como *intervalo de amostragem* [25]. Ou seja,  $g[n] = g(nT_s)$ , como pode ser visto na Figura 4.1.

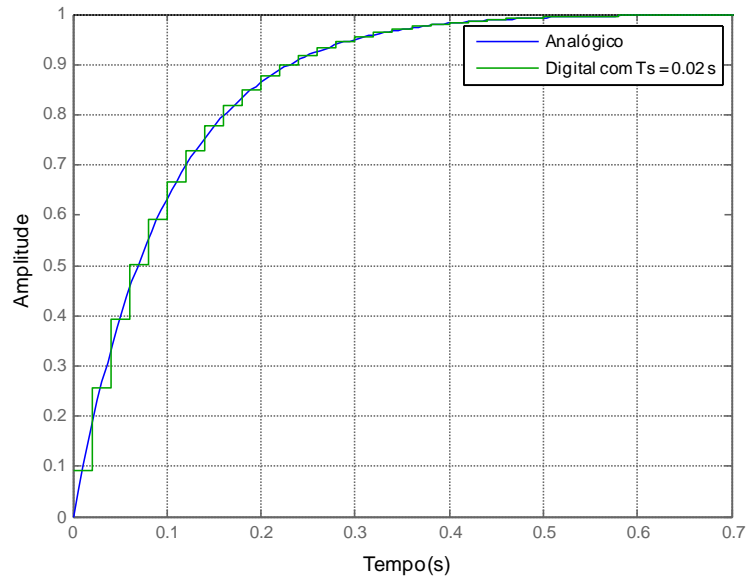


Figura 4.1 – Exemplo de amostragem de sinal analógico utilizando-se amostrador de ordem zero (ZOH).

#### 4.1.2 Integração numérica

Existem técnicas de integração numéricas bastante sofisticadas, utilizadas, por exemplo, para simular sistemas com modelos no espaço de estados, como por exemplo, as técnicas de *Range-Kutta* [26]. No Anexo 3, serão apresentadas técnicas simples, de passo fixo e lineares, de modo que a aproximação mantenha a característica de linearidade do sistema.

#### 4.2 Mapeamento do semiplano esquerdo do plano s no plano z.

O Anexo 3 aborda de modo breve o mapeamento do semiplano esquerdo do plano s no plano z para cada uma das aproximações apresentadas na seção anterior.

### 4.3 Resposta a distúrbios dos modelos de PLL

Nesta seção, será analisada a resposta dos modelos linearizados dos PLLs apresentados no Capítulo 3. A partir destes resultados, será possível prever com boa acurácia o desempenho dos modelos reais de PLL para diversos tipos de distúrbios. Estes distúrbios são todos amplamente conhecidos e utilizados para a análise de desempenho de algoritmos de sincronismo.

Em equipamentos como filtros ativos e condicionadores de qualidade de energia, de modo geral, os sinais de tensão e corrente frequentemente apresentam níveis de distorção harmônica alta. Como medida paliativa, os algoritmos de sincronismo utilizados para tal tendem a apresentar dinâmica mais lenta, de modo a aumentar a capacidade de filtragem do *Phase-Locked Loop*.

Nas próximas subseções, serão analisadas as respostas dos PLLs já apresentados anteriormente, frente a diversos distúrbios. Para isto, serão primeiramente discretizadas as suas plantas linearizadas e digitais.

#### 4.3.1 Modelagem digital

Para a modelagem discreta das malhas linearizada e digital dos PLLs, serão utilizados os seguintes parâmetros:

- Mapeamento com aproximação de *Tustin*
- Frequência de amostragem  $f_s = 10020 \text{ Hz}$

Desta forma, a discretização será implementada em 2 (duas) partes:

- Modelagem linearizada
- Modelo digital

A modelagem linear do pPLL, por exemplo, pode ser realizada a partir da utilização da equação (III.19) em (3.5). No entanto, em termos práticos, isto é extremamente custoso, haja vista a extensa aritmética para tal. Uma forma mais rápida (e mais transparente) de realizar o mesmo feito é a partir da discretização individual de cada um dos componentes da Figura 3.3, para logo após, implementar a sua malha.

A função de transferência do filtro de primeira ordem da Figura 3.3 pode ser dada por:

$$G_F(z) = \frac{Y_F(z)}{X_F(z)} = \frac{T_s(\omega_p + z^{-1})}{(\omega_p T_s - 2)z^{-1} + (\omega_p T_s + 2)} \quad (4.1)$$

Aplicando-se a transformada Z inversa na equação (4.1), é possível chegar à equação a diferenças do filtro passa-baixa:

$$y_F(k) = \frac{T_s(\omega_p x_F(k) + x_F(k-1)) - (\omega_p T_s - 2)y_F(k-1)}{\omega_p T_s + 2} \quad (4.2)$$

Analogamente, pode-se realizar o mesmo desenvolvimento para a planta do controlador proporcional integral e do integrador:

$$G_{PI}(z) = \frac{Y_{PI}(z)}{X_{PI}(z)} = \frac{(2k_p + k_i T_s) + (k_i T_s - 2k_p)z^{-1}}{2 - 2z^{-1}} \quad (4.3)$$

$$\begin{aligned} y_{PI}(k) \\ = \frac{(2k_p + k_i T_s)x_{PI}(k) + (k_i T_s - 2k_p)x_{PI}(k-1) + 2y_{PI}(k-1)}{2} \end{aligned} \quad (4.4)$$

$$G_{INT}(z) = \frac{Y_{INT}(z)}{X_{INT}(z)} = \frac{T_s(1 + z^{-1})}{2(1 - z^{-1})} \quad (4.5)$$

$$y_{INT}(k) = \frac{T_s(x_{INT}(k) + x_{INT}(k-1)) - 2y_{INT}(k-1)}{2} \quad (4.6)$$

A partir das equações (4.2), (4.4) e (4.6), é possível estabelecer tanto a malha do modelo linearizado da Figura 3.3 quanto o modelo digital para qualquer um dos PLLs apresentados na seção 3.1.

Foi realizada nas seções 3.1.1, 3.1.2 e 3.1.3 a análise matemática dos algoritmos *pPLL*, *parkPLL* e *EPLL* no tocante a sua linearização e regiões de estabilidade. Desta forma, pôde-se chegar aos ganhos ótimos de seus controladores proporcionais integrais, bem como a constante de tempo ideal do filtro passa-baixa do *phase detector* (excetuando-se o *EPLL*). Analisou-se, portanto, seus diagramas de Bode em malha aberta, bem como suas respostas ao degrau. Basicamente, como será visto mais a frente,

tais respostas são dois grandes indicadores para previsão do comportamento dos PLLs frente às perturbações com as quais o mesmo será confrontado.

### 4.3.2 Resposta ao degrau de fase para o modelo linearizado

O primeiro teste a ser realizado no modelo linear é no tocante ao degrau de fase. A metodologia adotada foi:

- Degrau de  $30^\circ$  (0.523 rad) em  $t = 0.0$  s
- Degrau de  $15^\circ$  (0.261 rad) em  $t = 0.5$  s
- Degrau de  $-30^\circ$  (-0.523 rad) em  $t = 1.0$  s

A partir deste teste, é possível analisar que todos os algoritmos apresentam o mesmo tempo de assentamento, como pode ser visualizado na Figura 4.2. O *EPLL* apresenta um nível de *overshoot* similar aos demais algoritmos. No entanto, seu comportamento transitório é mais oscilatório devido ao valor de seu coeficiente de amortecimento ser mais baixo.

Os resultados a seguir demonstram, portanto, que a digitalização a partir da aproximação de *Tustin* apresenta um satisfatório mapeamento do plano  $z$  no plano  $s$ .

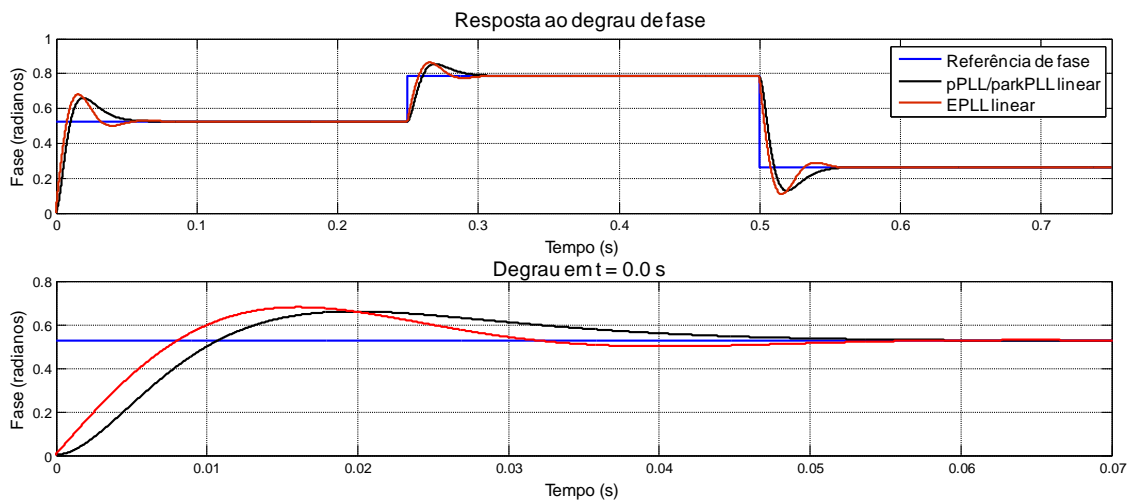


Figura 4.2 – Resposta ao degrau de fase para o modelo linearizado.

### 4.3.3 Resposta à inserção de harmônicos para o modelo linearizado

De modo a averiguar o comportamento dos PLLs discretizados frente a harmônicos, inseriu-se em  $D(s)$ , tal como verificado na Figura 3.3, uma perturbação de 1.0 p.u (pico) de 2º harmônico em torno do ponto de operação do modelo linearizado.

A Figura 4.3 demonstra esta análise. A saída do  $pPLL/parkPLL$  linear manteve-se em torno da referência de fase com um *ripple* de 0.1 rad. Ou seja, a atenuação respeita os -20 db propostos. No entanto, para o  $EPLL$ , ao contrário do apresentado na seção 3.1.3.2, obteve-se uma atenuação de 80% (aproximadamente -14 db), como pode ser visto na Figura 4.3.

Novamente, o valor da rejeição dos PLLs lineares discretizados foi muito próximo ao projetado e analisado na seção 3.1.1.2. Isto mostra que a digitalização do modelo a partir do método de aproximação de *Tustin* mapeia de forma coerente o plano  $s$  no plano  $z$ .

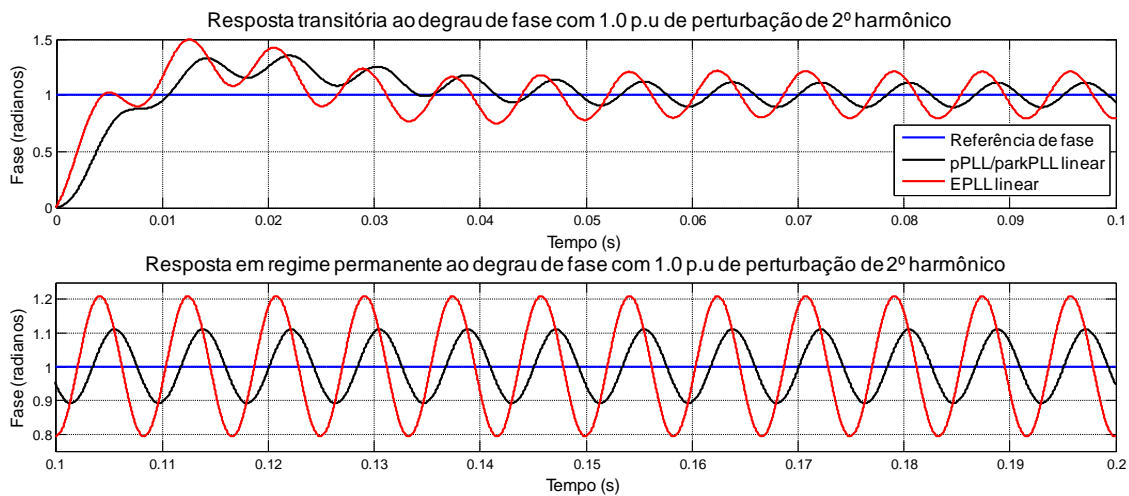


Figura 4.3 – Resposta a inserção de 1.0 p.u de 2º harmônico para o modelo linearizado.

### 4.3.4 Resposta à inserção de sub-harmônicos para o modelo linearizado

Em geral, devido à própria topologia dos  $PLLs$ , sub-harmônicos próximos à frequência fundamental tendem a ser amplificados, causando a perda de sincronismo quando da inserção mesmo de até pequenos níveis de perturbação sub-harmônica. Este efeito será explicado em detalhes no Capítulo 5. A Figura 4.4 demonstra este inconveniente. Como efeito comparativo, aplicou-se uma perturbação de 1.0 p.u de 50



Hz em torno do ponto de operação do modelo linearizado. É possível verificar que para os 3 (três) PLLs estudados, o nível de atenuação para o sub-harmônico de 50 Hz é de apenas 50% (aproximadamente -6 db). Deste modo, a referência de fase será rastreada com um erro de estado estacionário cíclico considerável, levando a um sincronismo errôneo.

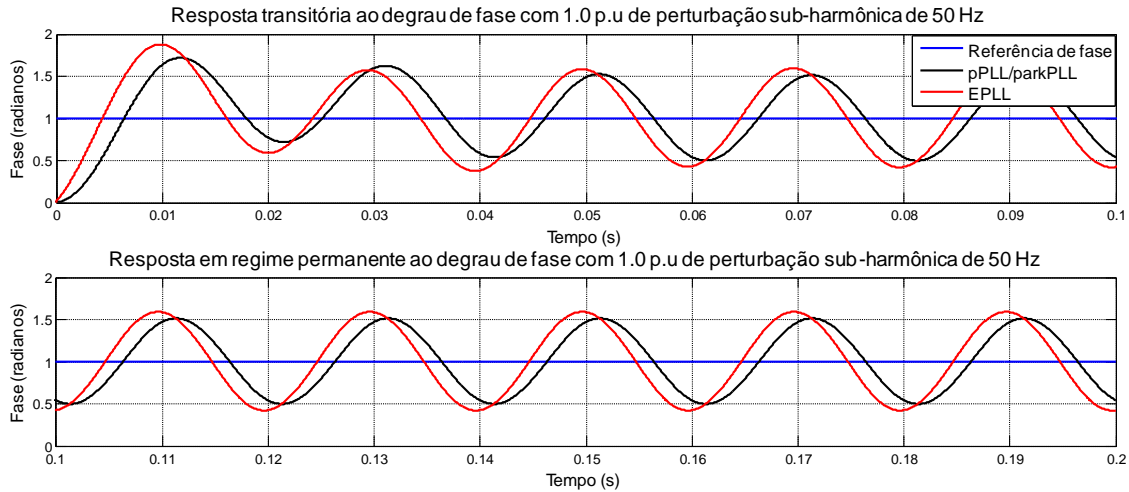


Figura 4.4 – Resposta a inserção de 1.0 p.u de perturbação sub-harmônica de 50 Hz para o modelo linearizado.

#### 4.3.5 Resposta ao degrau de fase para o modelo digital

Para esta análise foram realizados 3 (três) ensaios. O primeiro, presente na Figura 4.5, verifica a resposta do *pPLL* discretizado a um degrau de  $30^\circ$  em sua entrada para  $t = 0.15$  s. É possível verificar que, em aproximadamente 50 ms, tal como esperado, tanto o algoritmo linearizado quanto o digital efetuam o sincronismo, confirmando a teoria apresentada no Capítulo 3. Resultados semelhantes podem ser verificados para o *parkPLL* e *EPLL* nas Figura 4.6 e Figura 4.7, respectivamente.

Pode-se verificar a partir das Figura 4.5, Figura 4.6 e Figura 4.7 que o algoritmo rastreia a referência de fase com uma dinâmica muito próxima aos obtidos a partir dos modelos linearizados. Desta forma, conclui-se que, como esperado, o modelo linearizado prevê com segurança o tempo de assentamento e dinâmica do erro do controlador proporcional integral.

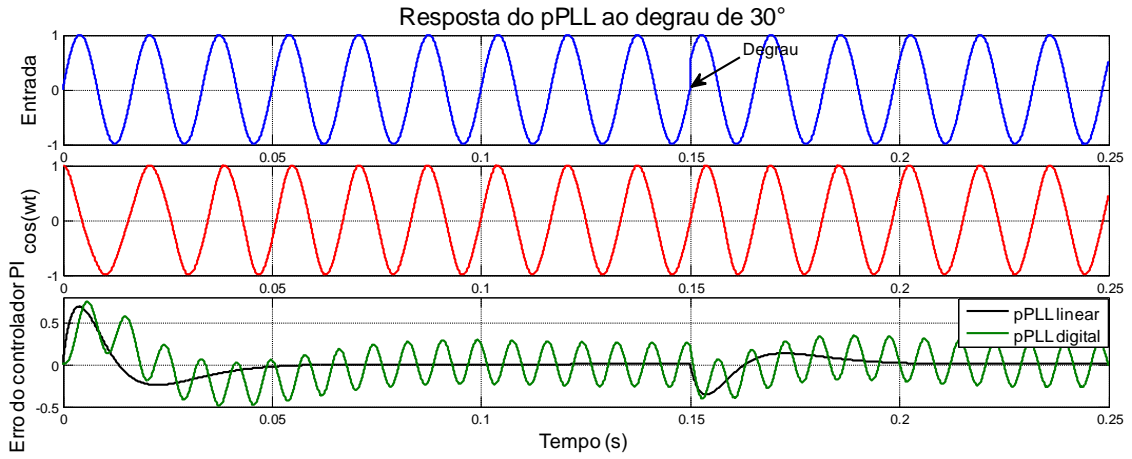


Figura 4.5 - Resposta do pPLL ao degrau de  $30^\circ$  em  $t = 0.15s$ .

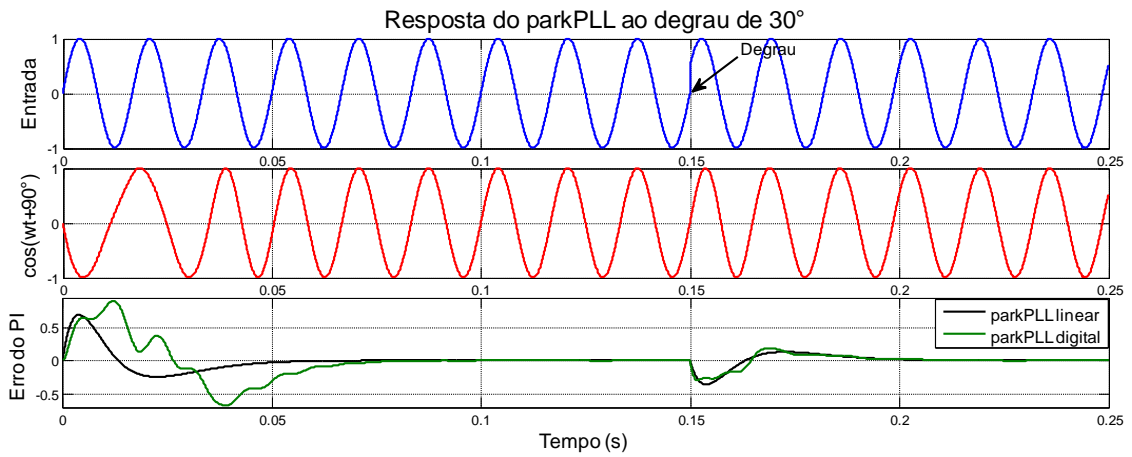


Figura 4.6 - Resposta do parkPLL ao degrau de  $30^\circ$  em  $t = 0.15s$ .

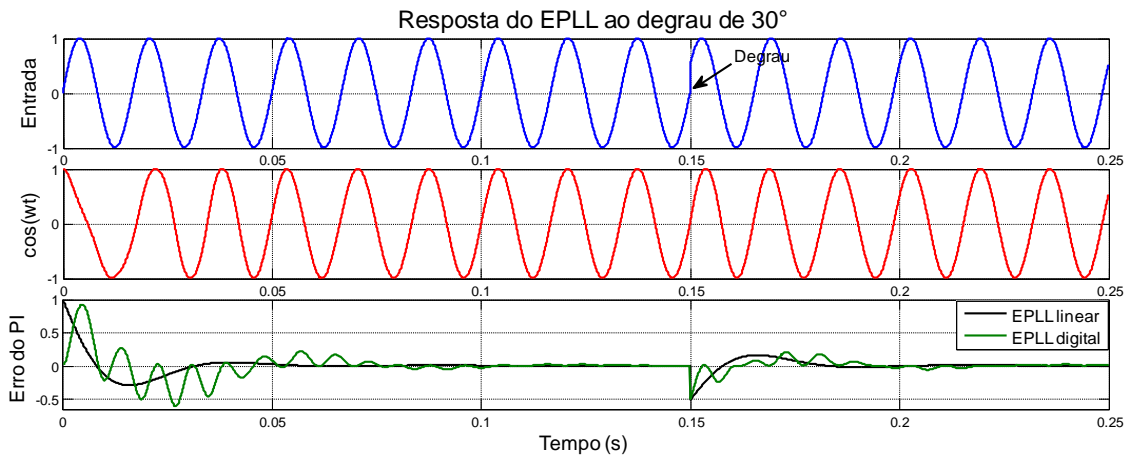


Figura 4.7 - Resposta do EPLL ao degrau de  $30^\circ$  em  $t = 0.15s$ .

### 4.3.6 Resposta à inserção de harmônicos para o modelo digital

Para esta análise, foi realizada a inserção de 1.0 p.u de perturbação harmônica de 2º harmônico no sinal de entrada no instante  $t = 0.15s$ . Como é possível verificar pela Figura 4.8, a atenuação no 2º harmônico obtida no sinal de saída do *pPLL* foi de 90%, como esperado e também obtido na seção 4.3.3. A mesma análise foi realizada para o *parkPLL* e *EPLL*, obtendo-se resultados semelhantes, confirmando-se assim, a teoria apresentada no Capítulo 3.

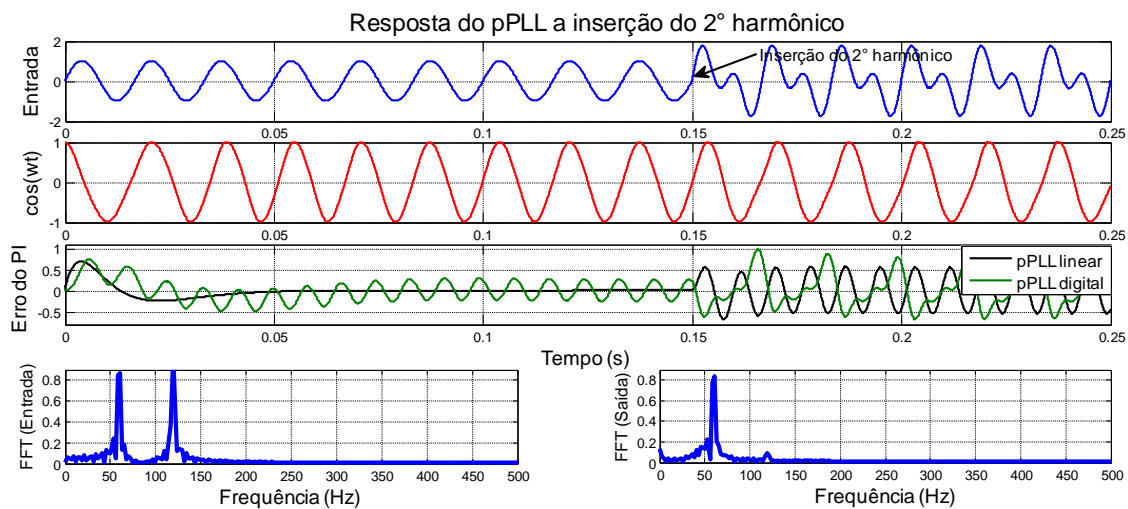


Figura 4.8 - Resposta do *pPLL* a inserção de 1.0 p.u de perturbação de 2º harmônico em  $t = 0.15s$ .

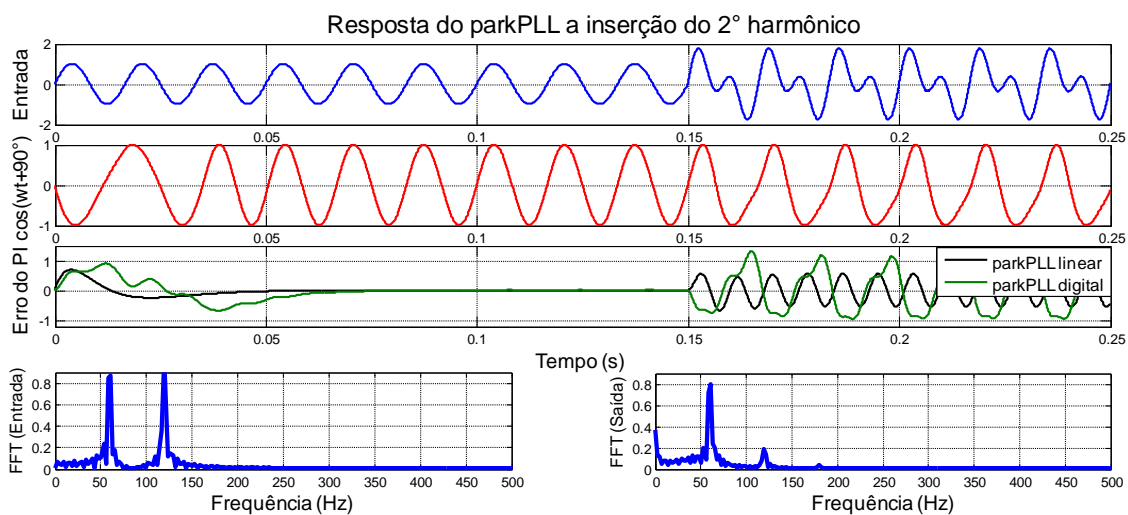


Figura 4.9 - Resposta do *parkPLL* a inserção de 1.0 p.u de perturbação de 2º harmônico em  $t = 0.15s$ .

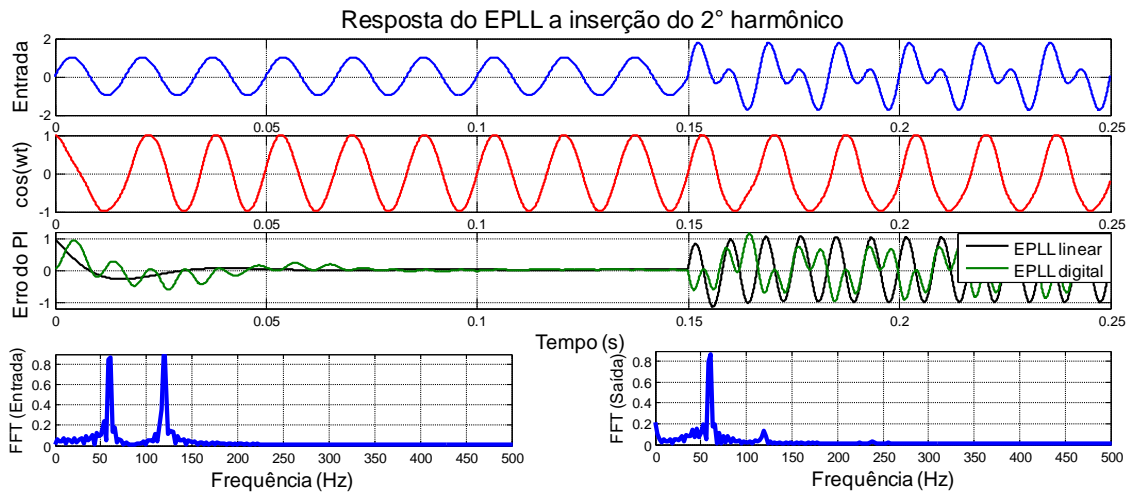


Figura 4.10 - Resposta do EPLL a inserção de 1.0 p.u de perturbação de 2º harmônico em  $t = 0.15s$ .

#### 4.3.7 Resposta à inserção de sub-harmônicos para o modelo digital

A análise realizada a seguir foi a seguinte: Inseriu-se uma perturbação de 1.0 p.u de sub-harmônico de 50 Hz à entrada no instante  $t = 0.15s$ . Como é possível verificar a partir das Figura 4.11, Figura 4.12 e Figura 4.13, a atenuação do sub-harmônico foi da ordem de 50%, tanto para o *pPLL* e *parkPLL* quanto para o *EPLL*. Este valor é similar ao obtido na seção 4.3.4, confirmando novamente a validade do modelo linearizado.

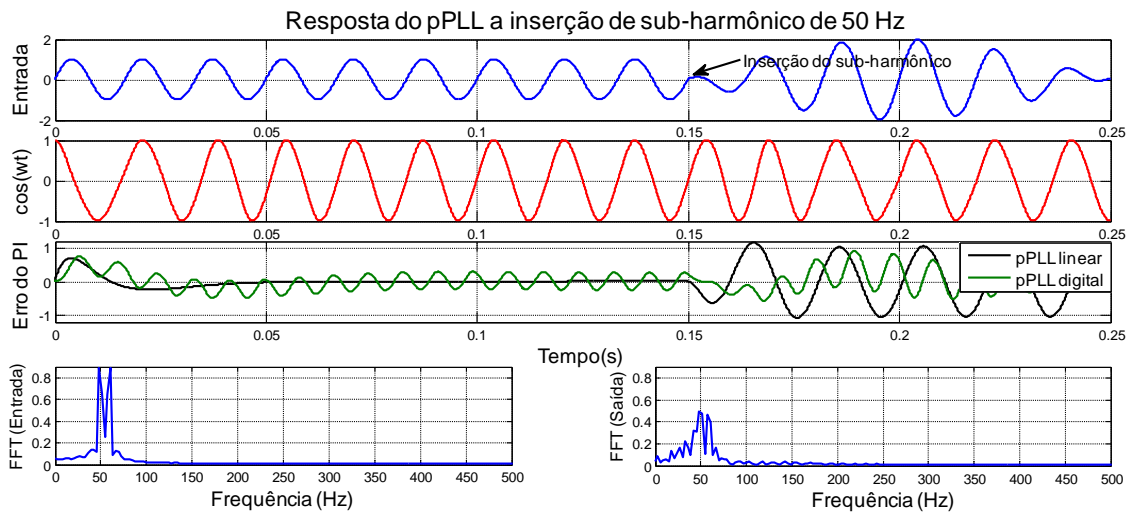


Figura 4.11 - Resposta do pPLL a inserção de 1.0 p.u de perturbação sub-harmônica de 50 Hz em  $t = 0.15s$ .

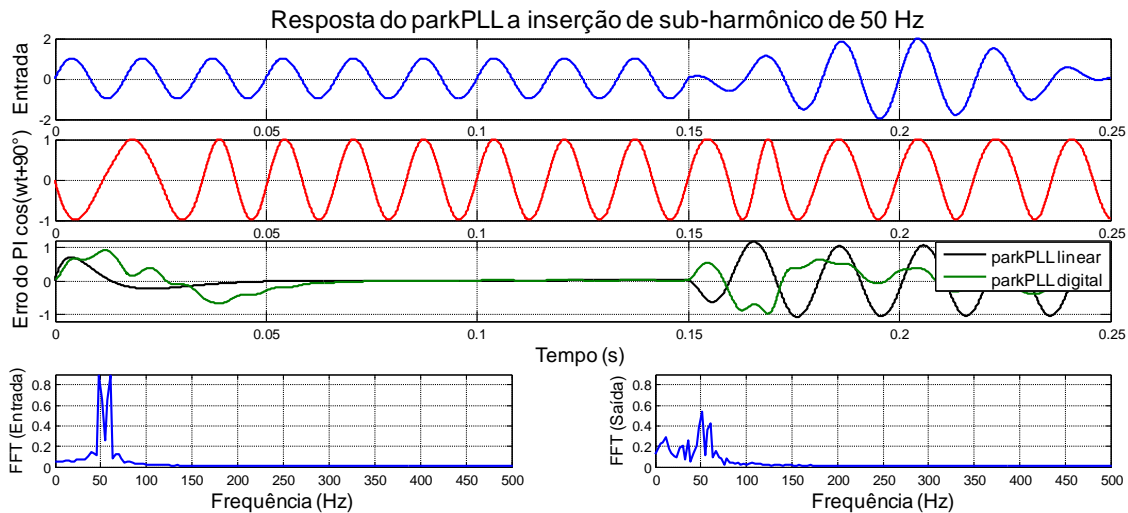


Figura 4.12 - Resposta do parkPLL a inserção de 1.0 p.u de perturbação sub-harmônica de 50 Hz em  $t = 0.15s$ .

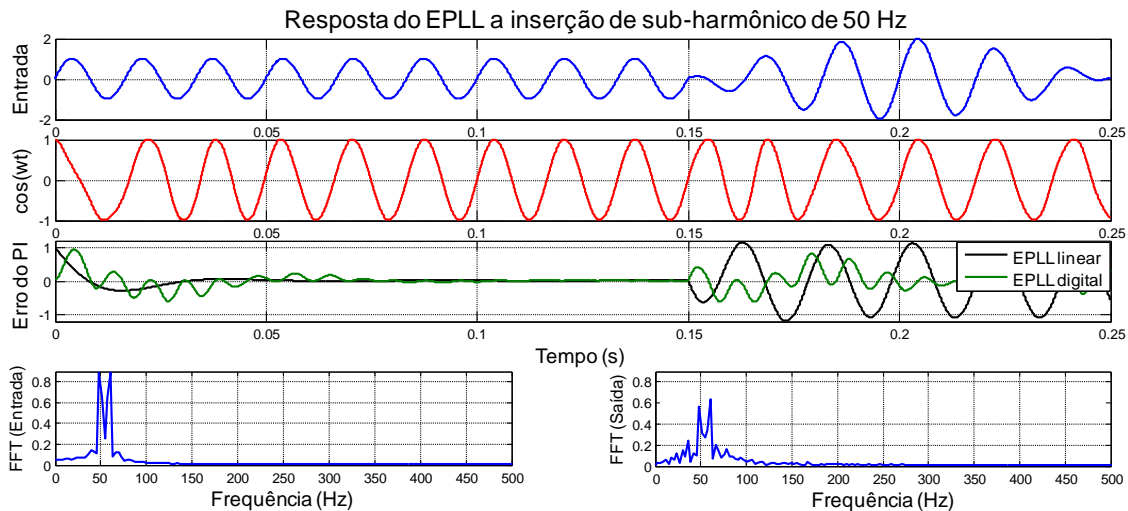


Figura 4.13 - Resposta do EPLL a inserção de 1.0 p.u de perturbação sub-harmônica de 50 Hz em  $t = 0.15s$ .

#### 4.4 Conclusões parciais

Neste capítulo, realizou-se uma análise de desempenho dos 3 (três) PLLs estudados neste trabalho: *pPLL*, *parkPLL* e *EPLL*. A partir das simulações realizadas, verificou-se que, para todos os casos, os resultados foram condizentes com a teoria apresentada no Capítulo 3.

É possível confirmar que a modelagem linear de PLLs monofásicos apresenta-se como uma análise inicial bastante pertinente dos algoritmos supracitados, sendo possível através daquela, estimar de antemão parâmetros importantes do algoritmo tais como tempo de assentamento e capacidade de atenuação harmônica.

Vale ressaltar que, a partir da análise realizada, comprovou-se a eficácia dos 3 (três) algoritmos. No entanto, o *EPLL* apresentou-se como a melhor escolha dos 3 (três), devido primeiramente a sua fácil implementação (não possui filtros passa-baixa ou transformações lineares como o *pPLL* e *parkPLL*, respectivamente) e maior margem de estabilidade.

## 5. Proposta de melhorias

Como foi analisado nos Capítulos 3 e 4, a resposta dos algoritmos de PLL, mesmo quando bem sintonizados, apresenta deficiências crônicas no tocante a diversos fenômenos ocorridos no sinal de entrada, tais como: presença de harmônicos, degraus de fase e, principalmente, presença de sub-harmônicos.

De todos os fenômenos citados acima, o sub-harmônico certamente é o maior responsável pela perda de sincronismo do algoritmo de PLL, como analisado nas seções 4.3.4 e 4.3.7.

De modo geral, pouco se aborda este tema na literatura. Devido à própria topologia da maior parte dos PLLs (principalmente, os baseados em controladores proporcionais integrais), a resposta precária perante perturbações sub-harmônicas é um fator inerente aos *Phase-Locked Loops*. Na seção a seguir, será abordada a causa do ocorrido, bem como uma proposta de melhoria na topologia dos PLLs, de forma a mitigar este problema.

### 5.1 Sub-harmônicos e a topologia do *pPLL* e *parkPLL*

De acordo com as seções 3.1.1, 3.1.3 e as equações (3.5) e (3.39), pode-se obter a relação entre um sinal de entrada de dada frequência e a sua atenuação característica através do diagrama de resposta em frequência. Utilizando-se dos parâmetros ótimos obtidos nas seções 3.1.1.2 e 3.1.3.2, obtêm-se os seguintes diagramas de resposta em frequência em malha fechada, apresentados na Figura 5.1.

É possível, portanto, verificar que para o *pPLL* e *parkPLL* as atenuações na frequência de 50 Hz (314 rad/s) são da ordem de -5.89 dB, ao passo que na frequência de 60 Hz (377 rad/s), atenua-se em -8.39 dB. No caso do *EPLL*, a atenuação na frequência de 50 Hz (314 rad/s) é de -4.63 dB, enquanto que na frequência de 60 Hz (377 rad/s), atenua-se em -6.70 dB. Tais resultados se mostraram condizentes com os obtidos nas seções 4.3.4 e 4.3.7.

Exemplificando de forma prática, toma-se o exemplo proposto nas seções 4.3.4 e 4.3.7. No entanto, neste caso, estipulou-se uma entrada unitária senoidal a 60 Hz aliada a uma perturbação de 20% de 50 Hz, obtendo-se o exposto na Tabela 5.1:

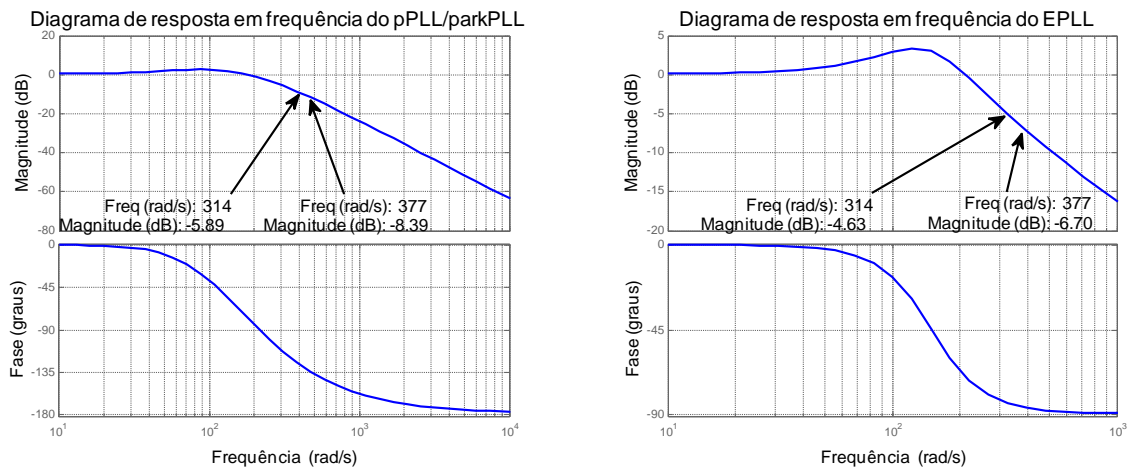


Figura 5.1 – Diagramas de resposta em frequência do pPLL/parkPLL e EPLL.

Tabela 5.1 – Tabela comparativa ilustrando o efeito amplificativo de perturbações sub-harmônicas no pPLL/parkPLL e EPLL a partir dos valores obtidos na Figura 5.1.

Entrada pPLL/parkPLL			Saída pPLL/parkPLL		
Magnitude 60 Hz (p.u)	Magnitude 50 Hz (p.u)	Proporção (%)	Magnitude 60 Hz (p.u)	Magnitude 50 Hz (p.u)	Proporção (%)
1.0	0.2	20	0.3806	0.1014	26.64
Entrada EPLL			Saída EPLL		
Magnitude 60 Hz (p.u)	Magnitude 50 Hz (p.u)	Proporção (%)	Magnitude 60 Hz (p.u)	Magnitude 50 Hz (p.u)	Proporção (%)
1.0	0.2	20	0.4623	0.1173	25.37

Ou seja, tanto a topologia do pPLL/parkPLL quanto do EPLL são inerentemente amplificativas para sub-harmônicos próximos a frequência fundamental.

## 5.2 Proposta de correção para o problema de sub-harmônicos

De modo a obter uma resposta sincronizada para todos os PLLs citados anteriormente em regime permanente, necessita-se estabelecer uma metodologia para mitigar os efeitos da oscilação devido às perturbações sub-harmônicas. Na próxima seção, serão apresentados as propostas para pPLL/parkPLL e EPLL.



### 5.2.1 *pPLL, parkPLL e EPLL*

Como explicitado na seção 3.1, a estrutura dos PLLs apresentados se assemelha em muito ao comportamento de um filtro passa-baixa de 2ª ordem. Este fato se deve principalmente ao comportamento do *Loop Filter*. Deste modo, as equações (3.5) e (3.35), podem ter seus comportamentos previstos a partir de:

$$G_f(s) = \frac{\omega_c^2}{s^2 + \frac{\omega_c}{Q}s + \omega_c^2} \quad (5.1)$$

onde  $\omega_c$  é a frequência de ressonância do filtro e  $Q$  é o seu fator de qualidade.

É importante ressaltar que a dinâmica da equação (5.1) é válida para valores de  $k_i \gg k_p$  e  $\tau \ll k_i$ . De modo geral, os PLLs se adequam a esta realidade, quando bem sintonizados, como realizado no Capítulo 3.

Portanto, tomando a equação (5.1), além dos parâmetros obtidos na seção 3.1.1.2, é possível chegar as seguintes equações:

$$k_i = \omega_c^2 \quad (5.2)$$

$$k_p = \frac{\omega_c}{Q} = \frac{\sqrt{k_i}}{Q} \quad (5.3)$$

A Figura 5.2 apresenta diversas respostas do sistema para diferentes valores de  $Q$ . Ou seja, é possível verificar que, tanto pela Figura 5.2 quanto pela equação (5.3) a relação entre o ganho proporcional  $k_p$  e o fator de qualidade  $Q$ .

Será realizada nas subseções a seguir uma análise prática de modo a estudar o comportamento do *EPLL* perante a inclusão de 20% de perturbação sub-harmônica (50 Hz) para uma variedade de frequências  $\omega_c$ . De modo a possibilitar uma análise mais breve, será utilizado um fator de qualidade  $Q$  unitário para todos os casos a serem estudados.

Além disto, serão comparadas as respostas do modelo linear presente em (3.39) com a aproximação de filtros passa-baixa de 2ª ordem exposta em (5.1).

Na seção 3.1.1.1, analisou-se a partir do critério de *Routh-Hurwitz* a margem de estabilidade do *pPLL*. A partir de (3.15), obtém-se a relação entre o ganho integral, o ganho proporcional e a constante de tempo  $\tau_p$  do filtro passa-baixa. De modo a respeitar o critério de estabilidade imposto, não será utilizado o filtro passa-baixas na implementação do PLL desta atual seção.

Ou seja, portanto, a implementação da solução visando mitigar o problema referente ao sincronismo sob sub-harmônico será atacado utilizando-se uma modificação do algoritmo *EPLL*.

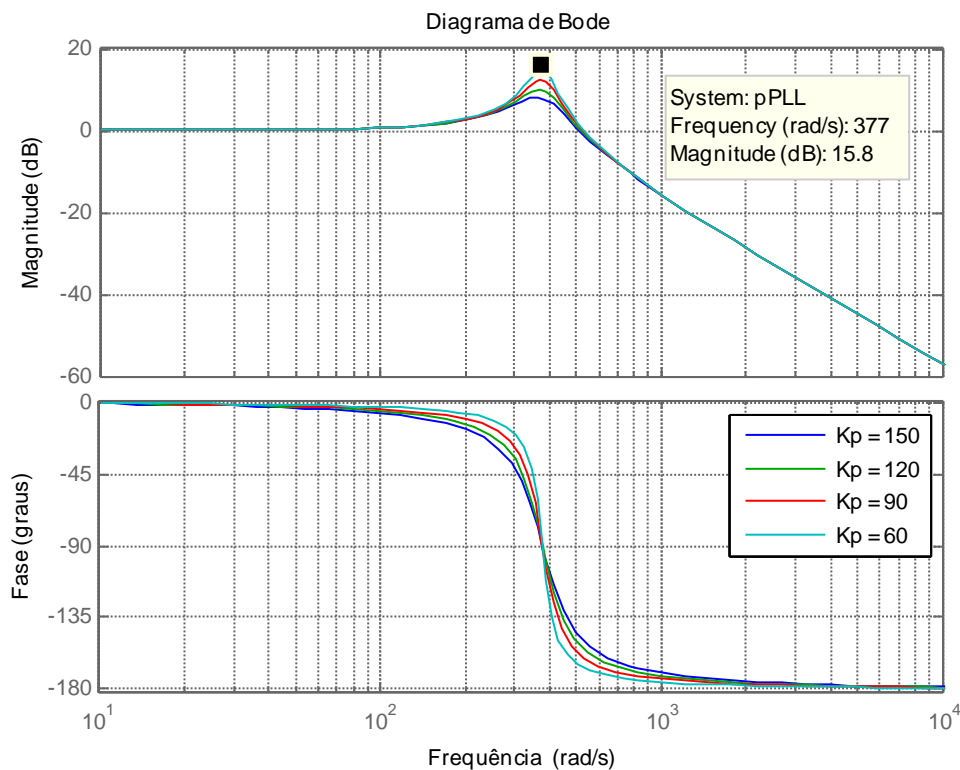


Figura 5.2 - Resposta do filtro passa-baixa de 2ª ordem com frequência de ressonância em  $\omega_c = 377$  rad/s para diversos valores de fator de qualidade  $Q$ .

### 5.2.1.1 Resposta do EPLL com parâmetros $\omega_c^2 = 30000$ e $Q = 1$

Como relatado na seção anterior, o comportamento do EPLL no tocante a presença de sub-harmônicos será averiguado de forma sistêmica, ou seja, a partir da interpretação dos resultados de seu diagrama de resposta em frequência. Deste modo, será possível identificar o que causa o efeito indesejado de oscilação do sinal de saída

do PLL em relação a sua referência fundamental quando da inserção de perturbações sub-harmônicas.

A Figura 5.3 apresenta os diagramas de Bode do *EPLL*, do filtro de 2ª ordem aproximado e a comparação entre os mesmos, além da resposta ao degrau de ambos.

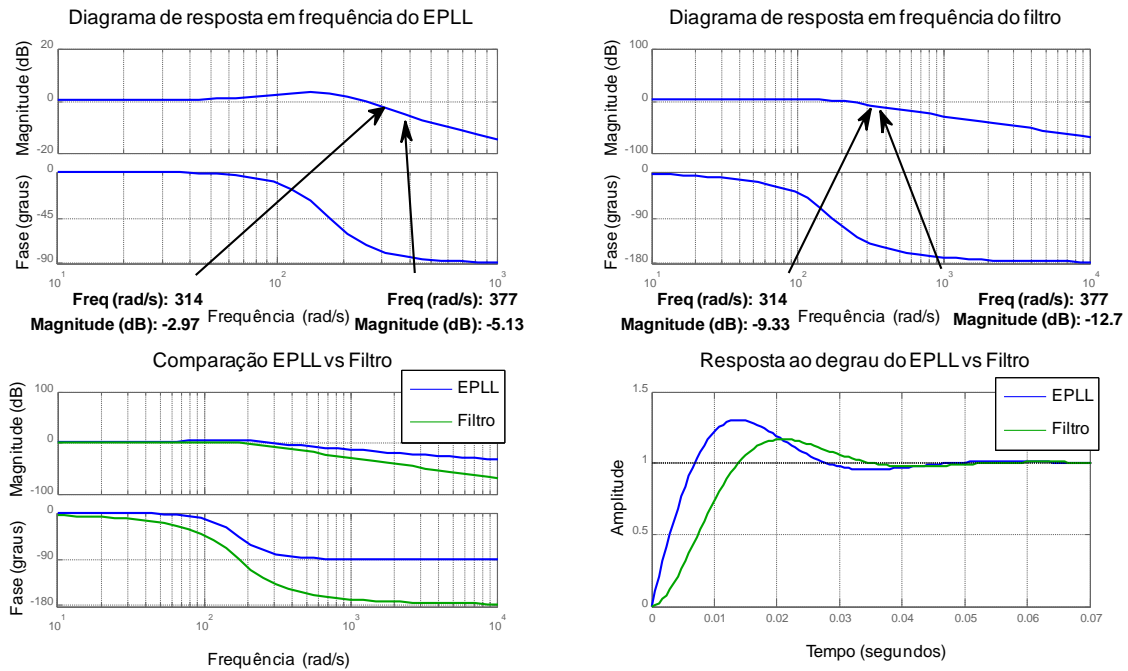


Figura 5.3 – Diagramas de Bode e respostas ao degrau do EPLL e filtro aproximado para  $\omega_c^2 = 30000$  e  $Q=1$ .

Como pode ser visto acima, o filtro não representa ainda uma boa aproximação para o *EPLL*, mesmo com o valor de  $k_p$  sendo mais de 100 (cem) vezes menor do que o valor do ganho integral  $k_i$ . A discrepância pode ser vista principalmente na resposta de fase do sistema. A seguir, encontram-se as análises comparativas do efeito amplificativo de sub-harmônicos.

É possível verificar que os tempos de acomodação de ambos os sistemas (*EPLL* e filtro) são parecidos. Além disto, apesar dos valores de atenuação serem bastante discrepantes entre o *EPLL* e o filtro, o grau de proporcionalidade entre componente de 50 Hz e componente de 60 Hz manteve-se próximo.

Tabela 5.2 - Tabela comparativa ilustrando o efeito amplificativo de perturbações sub-harmônicas no EPLL e filtro para  $\omega_c^2 = 30000$  e  $Q=1$ .

<b>Entrada EPLL</b>			<b>Saída EPLL</b>		
Magnitude 60 Hz (p.u)	Magnitude 50 Hz (p.u)	Proporção (%)	Magnitude 60 Hz (p.u)	Magnitude 50 Hz (p.u)	Proporção (%)
1.0	0.2	20	0.554	0.142	25.6
<b>Entrada Filtro</b>			<b>Saída Filtro</b>		
Magnitude 60 Hz (p.u)	Magnitude 50 Hz (p.u)	Proporção (%)	Magnitude 60 Hz (p.u)	Magnitude 50 Hz (p.u)	Proporção (%)
1.0	0.2	20	0.232	0.068	29.3

### 5.2.1.2 Resposta do EPLL com parâmetros $\omega_c^2 = 3000$ e $Q = 1$

A Figura 5.4 demonstra, respectivamente, os diagramas de Bode do EPLL e filtro aproximado, a comparação entre os mesmos e a resposta ao degrau de ambos.

Como pode ser visto abaixo, novamente o filtro não representa uma boa aproximação para o EPLL, mesmo com o valor de  $k_p$  sendo mais de 50 (cinquenta) vezes menor do que o valor do ganho integral  $k_i$ . A seguir, encontram-se as análises comparativas do efeito amplificativo de sub-harmônicos. Além disto, é especificado o tempo de resposta dos sistemas em questão, a partir de suas respostas ao degrau.

É possível verificar que os tempos de acomodação de ambos os sistemas (EPLL e filtro) são extremamente parecidos. Além disto, apesar dos valores de atenuação serem bastante discrepantes entre o EPLL e o filtro, o grau de proporcionalidade entre componente de 50 Hz e componente de 60 Hz manteve-se novamente bem próximo. Pode-se verificar a partir da Figura 5.4 e Tabela 5.3 que, devido à diminuição na diferença entre  $k_p$  e  $k_i$ , a aproximação do filtro foi ainda mais discrepante em relação ao caso anterior.

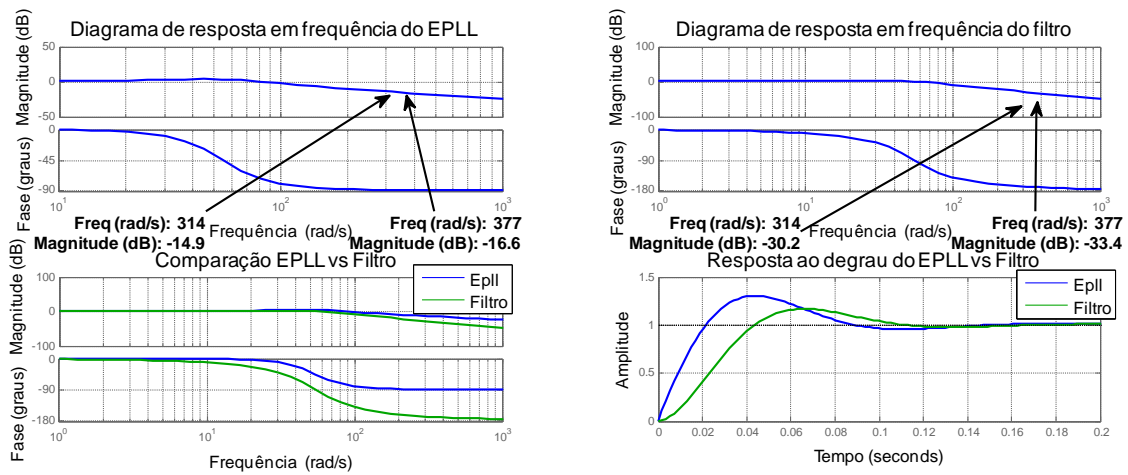


Figura 5.4 - Diagrama de Bode do EPLL para  $\omega_c^2 = 3000$  e  $Q=1$ .

Tabela 5.3 - Tabela comparativa ilustrando o efeito amplificativo de perturbações sub-harmônicas no EPLL e filtro para  $\omega_c^2 = 3000$  e  $Q=1$ .

Entrada EPLL			Saída EPLL		
Magnitude	Magnitude	Proporção	Magnitude	Magnitude	Proporção
60 Hz (p.u)	50 Hz (p.u)	(%)	60 Hz (p.u)	50 Hz (p.u)	(%)
1.0	0.2	20	0.148	0.036	24.3
Entrada Filtro			Saída Filtro		
Magnitude	Magnitude	Proporção	Magnitude	Magnitude	Proporção
60 Hz (p.u)	50 Hz (p.u)	(%)	60 Hz (p.u)	50 Hz (p.u)	(%)
1.0	0.2	20	0.021	0.006	28.5

### 5.2.1.3 Resposta do EPLL com parâmetros $\omega_c^2 = 300$ e $Q = 1$

A Figura 5.5 demonstra, respectivamente, os diagramas de Bode do EPLL, do filtro de 2ª ordem aproximado e a comparação entre os mesmos.

Como pode ser visto abaixo, novamente o filtro não representa uma boa aproximação para o EPLL, mesmo com o valor de  $k_p$  sendo mais de 15 (quinze) vezes menor do que o valor do ganho integral  $k_i$ . A seguir, na Tabela 5.4 encontram-se as análises comparativas do efeito amplificativo de sub-harmônicos. Além disto, são especificados na Figura 5.5 o tempo de resposta dos sistemas em questão, a partir de suas respostas ao degrau.

É possível verificar que os tempos de acomodação de ambos os sistemas (*EPLL* e filtro) são próximos. Além disto, apesar dos valores de atenuação serem bastante discrepantes entre o *EPLL* e o filtro, o grau de proporcionalidade entre componente de 50 Hz e componente de 60 Hz novamente manteve-se próximo. Pode-se verificar a partir da Tabela 5.4 que, devido à diminuição na diferença entre  $k_p$  e  $k_i$ , a aproximação do filtro foi ainda mais discrepante em relação aos casos anteriores.

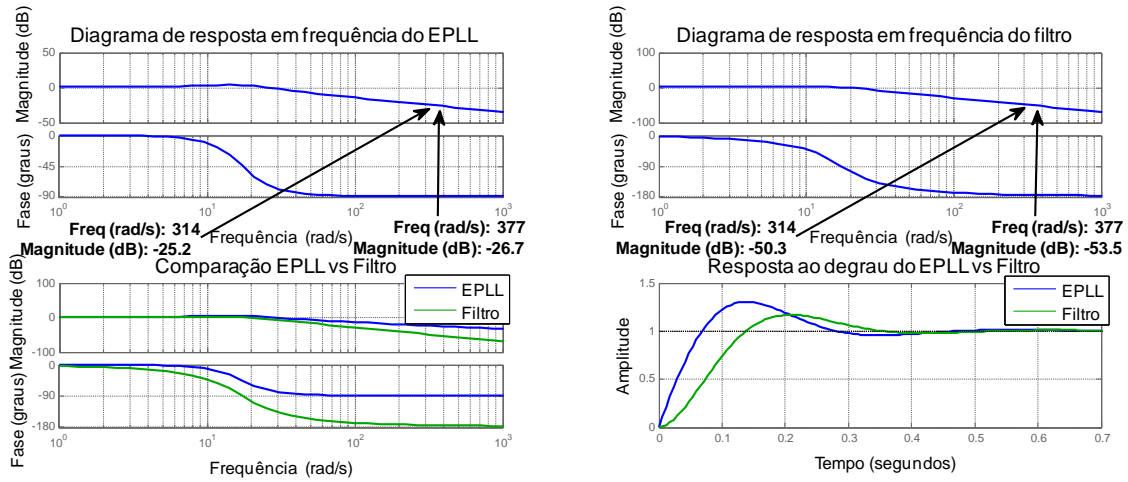


Figura 5.5 - Diagrama de Bode do EPLL para  $\omega_c^2 = 300$  e  $Q=1$ .

Tabela 5.4 - Tabela comparativa ilustrando o efeito amplificativo de perturbações sub-harmônicas no EPLL e filtro para  $\omega_c^2 = 300$  e  $Q=1$ .

Entrada EPLL			Saída EPLL		
Magnitude	Magnitude	Proporção	Magnitude	Magnitude	Proporção
60 Hz (p.u)	50 Hz (p.u)	(%)	60 Hz (p.u)	50 Hz (p.u)	(%)
1.0	0.2	20	0.046	0.011	23.9
Entrada Filtro			Saída Filtro		
Magnitude	Magnitude	Proporção	Magnitude	Magnitude	Proporção
60 Hz (p.u)	50 Hz (p.u)	(%)	60 Hz (p.u)	50 Hz (p.u)	(%)
1.0	0.2	20	0.002	0.0006	30.0

#### 5.2.1.4 Considerações parciais

Foi desenvolvida nas seções anteriores, de forma analítica, uma metodologia para verificação do comportamento do *EPLL* quando da inserção de perturbações sub-harmônicas. A partir das Tabela 5.2, Tabela 5.3 e Tabela 5.4 é possível verificar que a atenuação é maior na medida em que a banda do PLL se torna mais estreita (menores valores de  $\omega_c$ ). Além disto, é possível também verificar que a resposta do filtro de 2ª ordem aproximado torna-se cada vez mais discrepante em relação a resposta real do *EPLL* com a diminuição dos valores de  $\omega_c$ .

É interessante salientar que os fatores de proporção presentes nas tabelas apresentadas nas seções anteriores foram obtidos dos diagramas de resposta em frequência dos sistemas em questão. No entanto, vale ressaltar que, na malha real de PLL, a frequência com a qual o algoritmo irá obter sincronismo será a que possuir maior magnitude final. Ou seja, caso seja aplicada uma enorme perturbação (da mesma ordem da fundamental) sub-harmônica, o PLL será sincronizado com esta frequência.

No entanto, estas perturbações são praticamente inexistentes. De modo a investigar se a teoria proposta acima é cabível de ser implementada, serão apresentadas simulações para os 3 (três) casos descritos na seção 5.2.1 e, logo após, seus resultados serão comparados com a teoria proposta neste capítulo.

#### 5.2.2 Simulação do modelo digital de *EPLL*

Nesta seção serão apresentados os resultados de implementação do modelo digital de *EPLL* a partir da teoria deste capítulo. De modo a validar os resultados teóricos obtidos na seção 5.2.1, será apresentada a análise comparativa entre sinal de referência e sinal de saída do *EPLL*. A partir disto, parâmetros como nível de atenuação da perturbação sub-harmônica (50 Hz) e tempo de acomodação do sistema serão obtidos e comparados com o valor teórico obtido anteriormente.

De modo a averiguar o comportamento do *EPLL* perante a inserção de perturbações sub-harmônicas, serão analisados 3 (três) modelos hipotéticos, os quais terão o seu parâmetro  $\omega_c^2$  (referente à banda passante do algoritmo em questão) modificado de  $\omega_c^2 = 30000$  até  $\omega_c^2 = 300$ , conforme utilizado nas seções anteriores.

### 5.2.2.1 Comportamento do EPLL com parâmetros $\omega_c^2 = 30000$ e $Q = 1$

De modo a validar a teoria discutida anteriormente, faz-se necessário verificar se a implementação do EPLL digital discretizado apresentará comportamento próximo ao esperado. Portanto, a partir das equações (5.2) e (5.3), é possível obter os valores de  $k_p = 173.2$  e  $t_i = 0.0057$  s. De posse destes valores, implementa-se o EPLL digital, obtendo-se a resposta do algoritmo de PLL a um seno na frequência fundamental (60 Hz) aliado a 20% de perturbação sub-harmônica (50 Hz).

Como é possível verificar pela Figura 5.6, o efeito de batimento oscilatório ocasionado devido à perturbação é notório e permanente. A partir do espectro harmônico do sinal de saída do EPLL digital implementado é possível verificar que o índice espectral de 50 Hz é bem próximo ao previsto na seção 5.2.1.1, o que já era esperado, devido à similaridade dos ganhos proporcional e integral desta seção com os obtidos em 3.1.3.2.

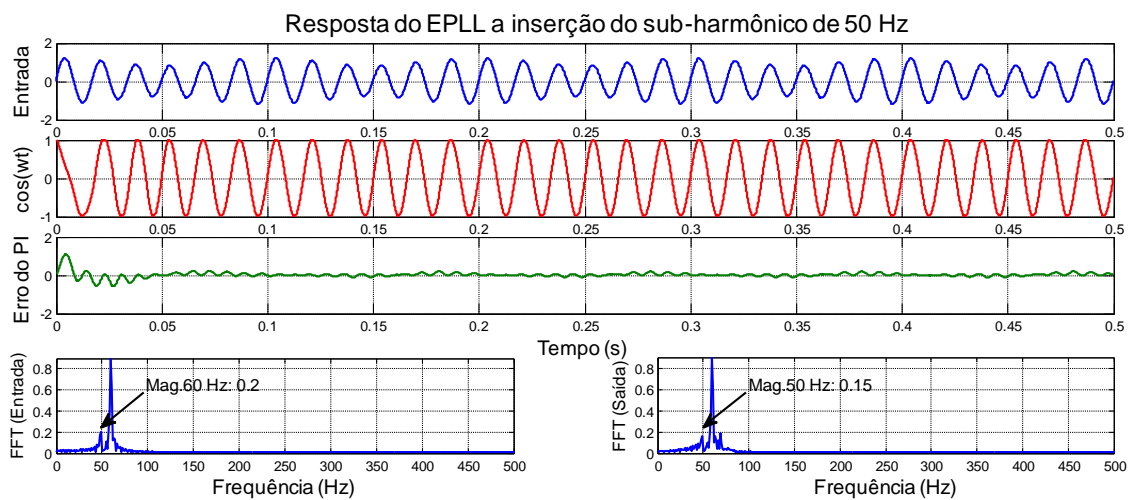


Figura 5.6 – Comportamento do EPLL com parâmetros  $\omega_c^2 = 30000$  e  $Q = 1$  e 20% de perturbação sub-harmônica de 50 Hz.

### 5.2.2.2 Comportamento do EPLL com parâmetros $\omega_c^2 = 3000$ e $Q = 1$

A partir das equações (5.2) e (5.3), é possível obter os valores de  $k_p = 54.7$  e  $t_i = 0.018$  s. De posse destes valores, implementa-se o EPLL digital, obtendo-se a



resposta do algoritmo de PLL a um seno na frequência fundamental (60 Hz) aliado a 20% de perturbação sub-harmônica (50 Hz).

Como é possível verificar pela Figura 5.7, o efeito de batimento oscilatório ocasionado devido à perturbação menos acentuado que para o caso anterior, mas ainda é permanente. A partir do espectro harmônico do sinal de saída do *EPLL* digital implementado é possível verificar que o índice espectral de 50 Hz é de fato menor que para o caso anterior. No entanto, existe um erro entre o percentual espectral de sub-harmônico obtido (aproximadamente 7%) e o obtido teoricamente na seção 5.2.1.2 (aproximadamente 3.6%).

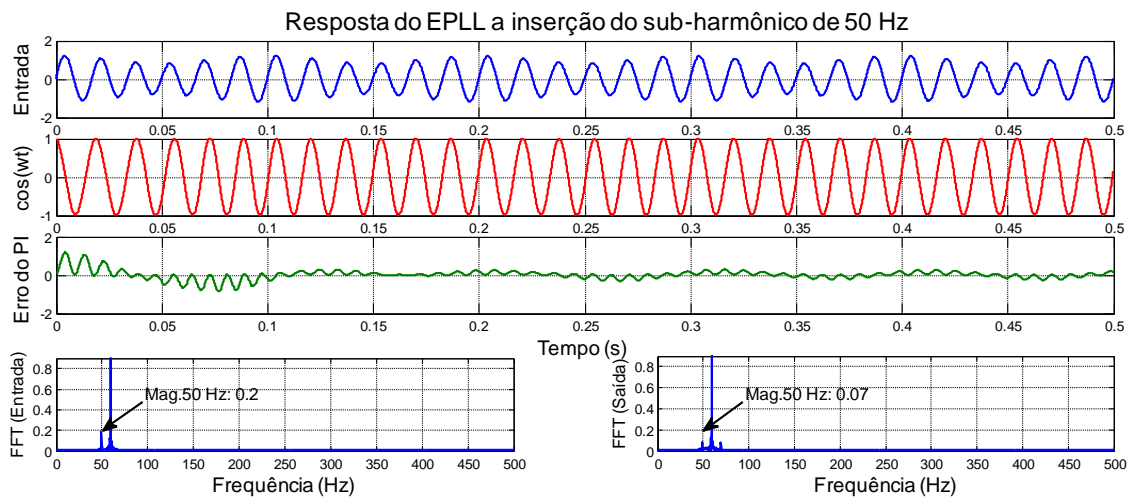


Figura 5.7 - Comportamento do *EPLL* com parâmetros  $\omega_c^2 = 3000$  e  $Q = 1$  e 20% de perturbação sub-harmônica de 50 Hz.

### 5.2.2.3 Comportamento do *EPLL* com parâmetros $\omega_c^2 = 300$ e $Q = 1$

A partir das equações (5.2) e (5.3), é possível obter os valores de  $k_p = 17.3$  e  $t_i = 0.057$  s. De posse destes valores, implementa-se o *EPLL* digital, obtendo-se a resposta do algoritmo de PLL a um seno na frequência fundamental (60 Hz) aliado a 20% de perturbação sub-harmônica (50 Hz).

Como é possível verificar pela Figura 5.8, o efeito de batimento oscilatório ocasionado devido à perturbação é agora praticamente inexistente. A partir do espectro harmônico do sinal de saída do *EPLL* digital implementado é possível verificar que o índice espectral de 50 Hz é praticamente igual ao esperado na seção 5.2.1.3. Além disto, o equilíbrio do sistema é dado em aproximadamente 0.5 s, tal como esperado na resposta ao degrau da seção 5.2.1.3.

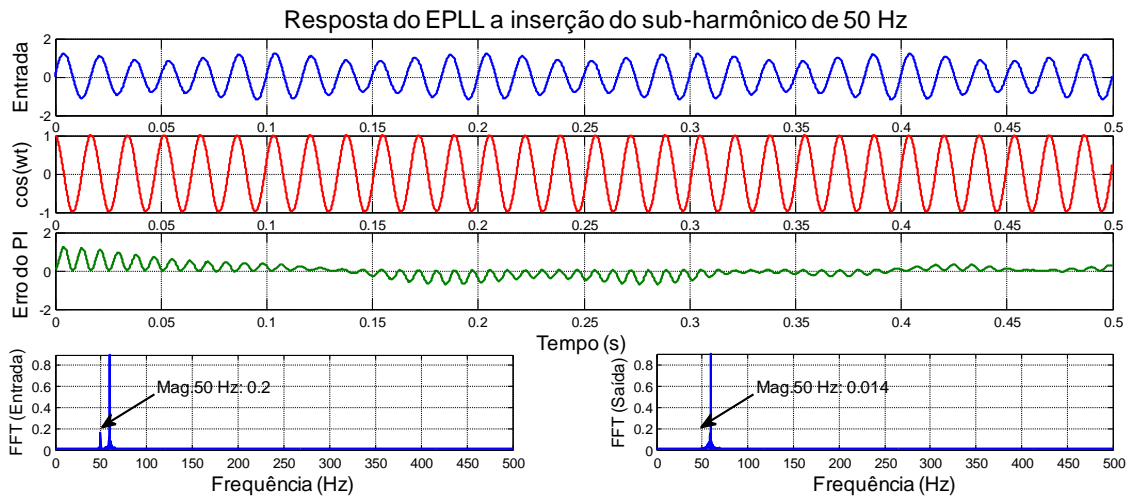


Figura 5.8 - Comportamento do EPLL com parâmetros  $\omega_c^2 = 300$  e  $Q = 1$  e 20% de perturbação sub-harmônica de 50 Hz.

### 5.3 Conclusões parciais

Neste capítulo, foi mostrado que o efeito indesejado de perturbações sub-harmônicas em PLLs de modo geral é possível de ser mitigado. Por outro lado, a correção deste efeito tende a diminuir a margem de fase do sistema como um todo, deixando-o com uma menor margem de estabilidade. Além disto, torna a resposta do sistema extremamente lenta, o que proporciona demora no sincronismo quando da ocorrência de degraus de fase/frequência. A partir do uso de controladores lineares na topologia de PLLs, mostrou-se impossível a implementação de um algoritmo com boa velocidade de rastreo e imune a perturbações. No Capítulo seguinte, far-se-á uso de técnicas de controle não linear adaptativa *fuzzy*, de modo a oferecer o melhor comportamento para a ocorrência de cada uma das situações de perturbação analisadas até agora.

## 6. FAPLL (*Fuzzy-Assisted PLL*)

Estratégias de controle adaptativas utilizando-se da teoria de lógica *fuzzy* tem sido assunto recorrente na literatura nos últimos anos. Esforços como os presentes em [27] e [28] são exemplos atuais de implementação de controladores digitais proporcionais integrais (por vezes, também derivativos) a partir de lógica *fuzzy*.

De modo geral, controladores do tipo proporcional integral *fuzzy* podem ser desenvolvidos de 2 (duas) formas: controladores inteiramente *fuzzy*, como o apresentado em [29] e controladores baseados em teoria de controle clássico com estratégia *fuzzy* de adaptação de ganhos, como o realizado em [30].

A topologia de *Phase-Locked Loop* apresentada neste capítulo utiliza controladores baseados em teoria de controle não-linear aliado ao uso de estratégia *fuzzy* para correção da saída do controlador. O motivo para a utilização desta forma em detrimento de um controlador proporcional integral com estratégia *fuzzy* adaptativa é a maior velocidade e confiabilidade do sistema como um todo.

Nas próximas seções, serão apresentados os principais conceitos de lógica *fuzzy*, bem como sua implementação em sistemas de controle. Além disto, será apresentada a topologia do FAPLL e seu comportamento perante diversos tipos de perturbação em seu sinal de entrada, como analisado no Capítulo 4. A partir dos resultados obtidos, será possível estabelecer comparações com os demais PLLs convencionais estudados neste trabalho.

### 6.1 Introdução à lógica *Fuzzy*

Sistemas de controle fornecem resposta a uma determinada entrada de acordo com sua função de transferência. Os assim chamados “sistemas inteligentes” são aqueles que fornecem respostas que solucionam problemas, as quais são apropriadas as situações específicas destes problemas, mesmo que sejam novas ou inesperadas, fazendo com que tal comportamento seja “único” ou até mesmo considerado “criativo”.

Até o presente momento, existe um grande descompasso entre a capacidade criativa dos seres humanos e a capacidade de solução que as máquinas computacionais nos proporcionam, devido ao fato de que as pessoas raciocinam de forma incerta,

imprecisa, difusa ou nebulosa, enquanto as máquinas são movidas por raciocínio preciso e binário. A eliminação desta restrição faria com que as máquinas fossem inteligentes, ou seja, pudessem raciocinar da mesma maneira imprecisa, como os seres humanos. Tal forma de raciocínio é chamada em inglês por “fuzzy” [31].

Um controlador *fuzzy* típico pode ser projetado para comportar-se conforme o raciocínio dedutivo, isto é, o processo que as pessoas utilizam para inferir conclusões baseadas em informações que elas já conhecem. A lógica *fuzzy* pode aportar um dado conhecimento em um controlador *fuzzy*, possibilitando a implementação de um controlador computacional com desempenho equivalente ao de operador humano.

Outra possível forma de raciocínio é o indutivo, o qual também pode ser incorporado no projeto de controladores *fuzzy*, onde seria possível o aprendizado e generalização através de exemplos particulares provenientes da observação do comportamento do processo numa situação dinâmica, ou variante no tempo [31].

Controladores que combinam técnicas convencionais e inteligentes são geralmente utilizados no controle inteligente de sistemas dinâmicos complexos. Controladores *fuzzy* operacionais ou supervisórios representam um típico exemplo onde apenas uma função de controle, da estratégia global, utiliza o enfoque *fuzzy*. No presente trabalho, o controlador *fuzzy* supervisório é incluído a malha do *Loop Filter* do PLL, tendo como consequência a automática excursão de saída do controlador quando da inserção de perturbações no sinal de entrada do PLL. A análise matemática realizada anteriormente serviu para que uma base de dados possa ser capturada em um controlador *fuzzy*, providenciando-se uma técnica heurística para se projetar os algoritmos de supervisão.

### **6.1.1 Modelagem de plantas e processos**

De modo geral, a modelagem de situações reais é necessária por poder ser objeto de utilização em ensaios e experimentos de uma forma mais fácil (além de menores custos) do que a utilização dos sistemas reais. Os tipos de modelagem podem ser divididos em: Método *experimental*, método *analítico ou matemático* e método *heurístico*.

### 6.1.1.1 Método heurístico

O método heurístico consiste em se realizar uma tarefa de acordo com a experiência prévia, com regras práticas e estratégias já frequentemente utilizadas. As regras heurísticas são implicações lógicas do tipo:

$$SE < condição > ENTÃO < consequência >$$

Estas regras associam conclusões (ou consequências) com condições (ou antecedentes). Em lógica *fuzzy*, tende-se a tratar dados de entrada e saída por entidades *fuzzy* também conhecidas como *funções de pertinência*. Ou seja, dada uma entrada hipotética  $e_{ent}$  e sua respectiva saída  $e_{saída}$ , pode-se exemplificar um sistema *fuzzy* através da seguinte relação:

$$SE e_{ent} = MÉDIO \text{ então } e_{saída} = GRANDE$$

onde MÉDIO e GRANDE são as *variáveis linguísticas* que são definidas através de funções de pertinência que descrevem a imprecisão de tais valores de entrada e saída, respectivamente [31].

Ou seja, ao invés de se utilizar uma expressão numérica, uma expressão linguística é utilizada para descrever a relação entre a variável de entrada e a de saída. Logo, uma vez que a descrição entrada-saída é descrita ponto a ponto, a restrição de linearidade não é mais necessária ou relevante. Portanto, o poder do método heurístico está em sua habilidade de se possibilitar a construção de uma função de entrada *versus* saída útil na descrição de uma planta. Isto pode ser realizado quando um modelo matemático equivalente for complexo de se obter, com a presença de parâmetros incertos, desconhecidos ou mesmo variantes no tempo.

No enfoque *fuzzy*, por exemplo, uma função real de entrada-saída é, de modo geral, aproximada por seções que cobrem regiões da função. Esta análise será vista com mais detalhes em seções posteriores. Em vez de se especificar um valor exato de um ponto na função de entrada-saída, a seção indicaria um conjunto de valores possíveis. De acordo com o *Teorema de Aproximação Fuzzy*, uma curva pode ser sempre convertida em descrições verbais através de um número finito de regiões *fuzzy*.

### 6.1.2 Recursos da lógica *fuzzy*

Um dos recursos principais da lógica *fuzzy* é a capacidade de utilização em tarefas de tomadas de decisão, onde as variáveis individuais não são definidas em termos exatos. No caso do presente trabalho, o sistema *fuzzy* supervisor é responsável por tomar a decisão de tornar o *Loop Filter* mais rápido (maior banda passante) ou mais lento (menor banda passante). Portanto, devido à característica multivalente do controlador *fuzzy*, é possível excursionar os ganhos de maneira não linear, ou seja, com todos os “graus de verdade entre o ser e o não-ser”, ou todos os “graus de duvidoso entre o claro e o obscuro”.

Na metodologia convencional de projetos de sistemas de controle, o que é geralmente modelado é a planta ou processo que está sendo controlado. Este procedimento é conhecido como identificação de sistemas. Tais sistemas são assumidos como lineares (ou aproximadamente lineares) e caracterizados por um conjunto de funções de transferência, cujas respostas indicariam ao controlador como os parâmetros devem ser ajustados de modo a obter-se um determinado comportamento do sistema (amortecimento, velocidade de resposta e tempo de acomodação, por exemplo). Esta estratégia foi utilizada com sucesso no Capítulo 3.

O controlador *fuzzy* baseado no modelo identificado do operador humano torna-se um modelo lógico da forma com que o operador raciocina em manipular o sistema, conforme ilustra a Figura 6.1.

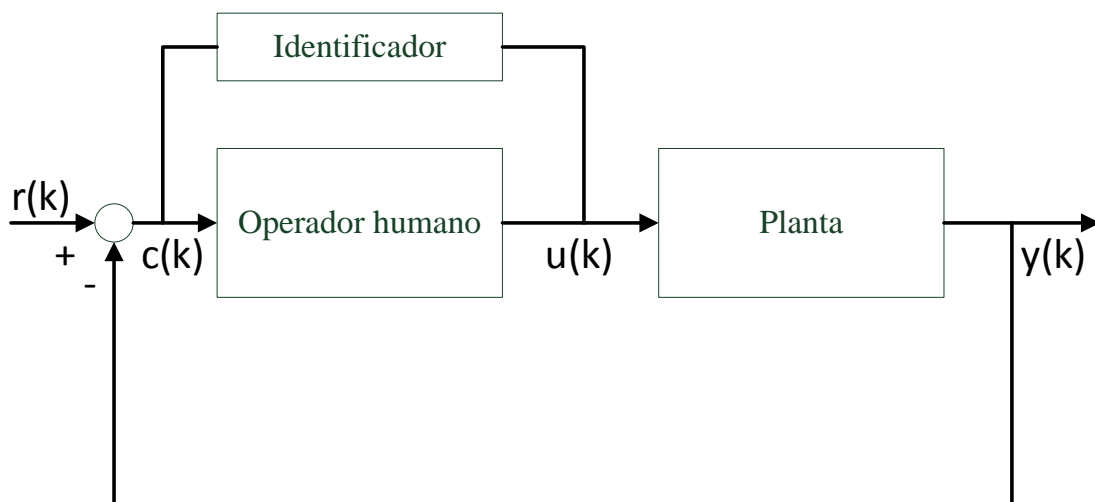


Figura 6.1 – Identificação do comportamento do operador. Adaptado de [31].

## 6.2 Pré-modelagem do controlador proporcional integral *Fuzzy*

De forma a projetar o controlador proporcional integral *fuzzy* que será agregado ao bloco de *Loop Filter* do PLL, faz-se necessário estabelecer, de forma matemática, o comportamento da malha do PLL como um todo. Deste modo, será possível criar regras heurísticas de tomada de decisão para o bloco identificador, dando início a implementação da estratégia supervisória do sistema.

Em [7], introduziu-se uma análise no domínio discreto de um modelo de segunda ordem de PLL. Em [4], foi apresentado o desenvolvimento matemático deste modelo. A Figura 6.2 apresenta o diagrama de blocos do modelo discreto de PLL.

A função de transferência discreta do diagrama de blocos apresentado na Figura 6.2 pode ser expressa por:

$$H_{DPLL}(z) = \frac{\phi_{saída}(z)}{\phi_{ref}(z)} = \frac{(K_1 + K_2)z^{-1} - K_1z^{-2}}{1 + (K_1 + K_2 - 2)z^{-1} + (1 - K_1)z^{-2}} \quad (6.1)$$

Os parâmetros  $K_1$  e  $K_2$  fazem alusão aos ganhos proporcional e integral, respectivamente.

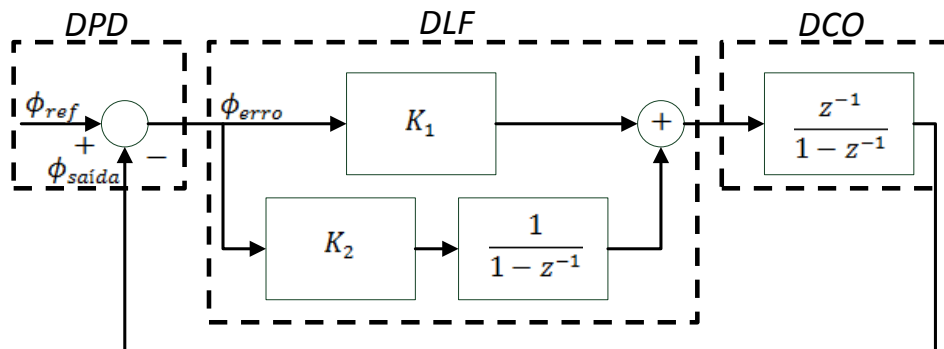


Figura 6.2 – Modelo discreto de PLL. Adaptado de [7].

Aplicando-se a transformação bilinear presente na equação (III.19) em (3.39) e considerando-se  $k_v = 1$ , obtém-se a seguinte relação:

$$H_{DPLL}(z) = \frac{\left(\frac{2k_p}{T_s} + k_i\right) + 2k_i z^{-1} + \left(k_i - \frac{2k_p}{T_s}\right) z^{-2}}{\left(\frac{4}{T_s^2} + \frac{2k_p}{T_s} + k_i\right) + \left(2k_i - \frac{8}{T_s^2}\right) z^{-1} + \left(\frac{4}{T_s^2} + k_i - \frac{2k_p}{T_s}\right) z^{-2}} \quad (6.2)$$

Considerando-se as relações (6.3) e (6.4) e substituindo-as na equação (6.2), chega-se a (6.5).

$$k_p = 2\xi\omega_n \quad (6.3)$$

$$k_i = \omega_n^2 \quad (6.4)$$

$$H_{DPLL}(z) = \frac{(4\xi\omega_n T_s + \omega_n^2 T_s) + (2\omega_n^2 T_s^2) z^{-1} + (\omega_n^2 T_s^2 - 4\xi\omega_n T_s) z^{-2}}{(4 + 4\xi\omega_n T_s + \omega_n^2 T_s^2) + (2\omega_n^2 T_s^2 - 8) z^{-1} + (4 + \omega_n^2 T_s^2 - 4\xi\omega_n T_s) z^{-2}} \quad (6.5)$$

A partir da comparação entre (6.1) e (6.5), chega-se na relação de ganhos  $K_1$  e  $K_2$  em função dos parâmetros  $\xi$  e  $\omega_n$ .

$$K_1 = \frac{8\xi\omega_n T_s}{4 + 4\xi\omega_n T_s + \omega_n^2 T_s^2} \quad (6.6)$$

$$K_2 = \frac{4(\omega_n T_s)^2}{4 + 4\xi\omega_n T_s + (\omega_n T_s)^2} \quad (6.7)$$

De modo a obter um parâmetro para a excursão dos ganhos do controlador, utiliza-se o conceito de *loop bandwidth/noise hemi-bandwidth* apresentado em [32] e [33]. A formulação matemática para uma função semelhante à descrita em (3.39) é apresentada em [34], como pode ser visto na equação (6.8).

$$B_L = \int_0^{\infty} |H(j\omega)|^2 d\omega = \frac{\omega_n}{2} \left( \xi + \frac{1}{4\xi} \right) \quad (6.8)$$

A relação entre o *loop bandwidth* normalizado (em relação à frequência natural de oscilação) e o fator de amortecimento  $\xi$  obtido a partir da equação (6.8) pode ser analisada na Figura 6.3. É possível verificar que a melhor relação sinal ruído se encontra em  $\xi \cong 0.5$ . Em uma primeira análise, seria bastante tentador projetar o sistema supervisor *fuzzy* em torno deste valor. No entanto, pode-se verificar que para valores



de  $\xi$  próximos de 0.7, o sistema se encontra ainda com ótimo valor de *loop bandwidth*. Para  $\xi = 0.7$ ,  $B_L = 0.5286\omega_n$ . Desta forma, obtém-se uma relação sinal ruído quase tão boa quanto para  $\xi \cong 0.5$  e resposta transitória mais rápida.

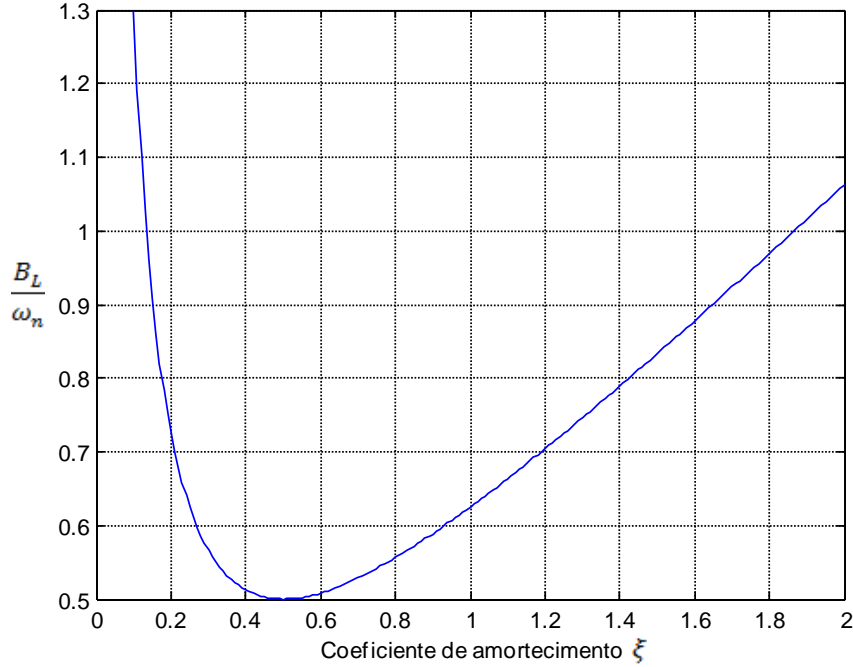


Figura 6.3 – Loop Bandwidth normalizado em relação ao coeficiente de amortecimento  $\xi$ .

De modo a obter a relação entre  $K_1$  e  $K_2$  e o fator de *Loop Bandwidth*  $B_L$ , substitui-se a equação (6.8) em (6.6) e (6.7), alcançando-se assim, as equações (6.9) e (6.10).

$$K_1 = \frac{64\xi^2 B_L T_s}{4(4\xi^2 + 1)^2 + 32\xi^2 B_L T_s(1 + 4\xi^2) + 64(\xi B_L T_s)^2} \quad (6.9)$$

$$K_2 = \frac{256(\xi B_L T_s)^2}{4(4\xi^2 + 1)^2 + 32\xi^2 B_L T_s(1 + 4\xi^2) + 64(\xi B_L T_s)^2} \quad (6.10)$$

Desta forma, é possível verificar a consequência da excursão do parâmetro de controle  $B_L$  nos ganhos proporcional e integral, como descrito em (6.9) e (6.10), respectivamente. É possível verificar a partir da Figura 6.4 que a excursão do ganho proporcional não é linear em relação a variação do *loop bandwidth*. De modo a obter

uma excursão mais simétrica entre valores positivos e negativos de  $\Delta K_1$ , definiu-se o máximo de excursão por  $-1000 \leq \Delta B_L \leq 1000$ .

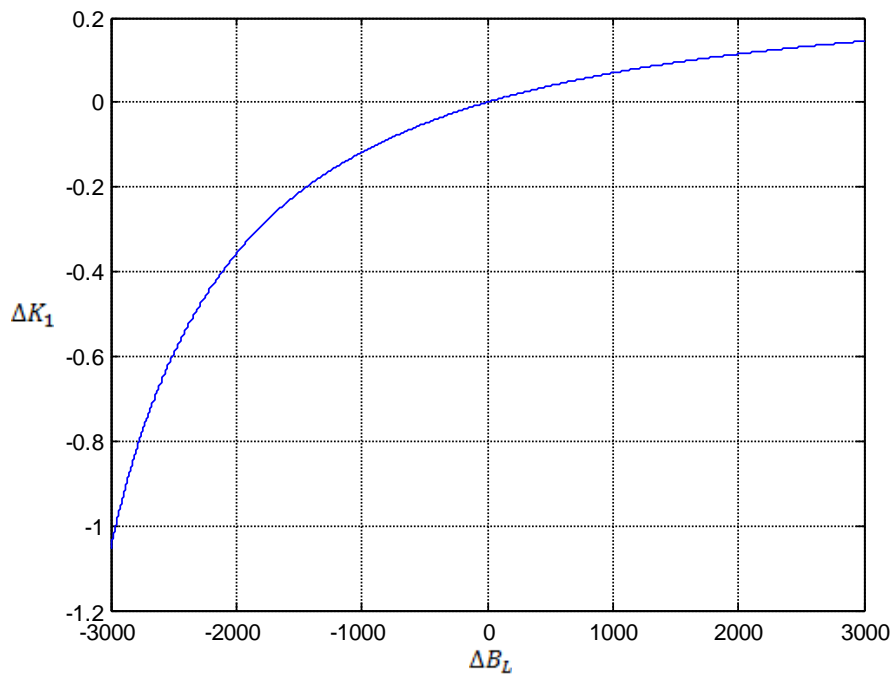


Figura 6.4 – Excursão do ganho proporcional do controlador em relação ao loop bandwidth.

De modo análogo, na Figura 6.5, estudou-se a consequência da excursão do parâmetro  $B_L$  no ganho integral do sistema. É possível verificar que novamente que a excursão do  $K_2$  não é linear em relação à variação do *loop bandwidth*. Portanto, definiu-se também neste caso o máximo de excursão por  $-1000 \leq \Delta B_L \leq 1000$ .

Deste modo, é possível implementar a tabela de regras do controlador *fuzzy* baseado no comportamento expostos nas Figura 6.4 e Figura 6.5. Esta análise será efetuada com clareza na seção 6.3.2.

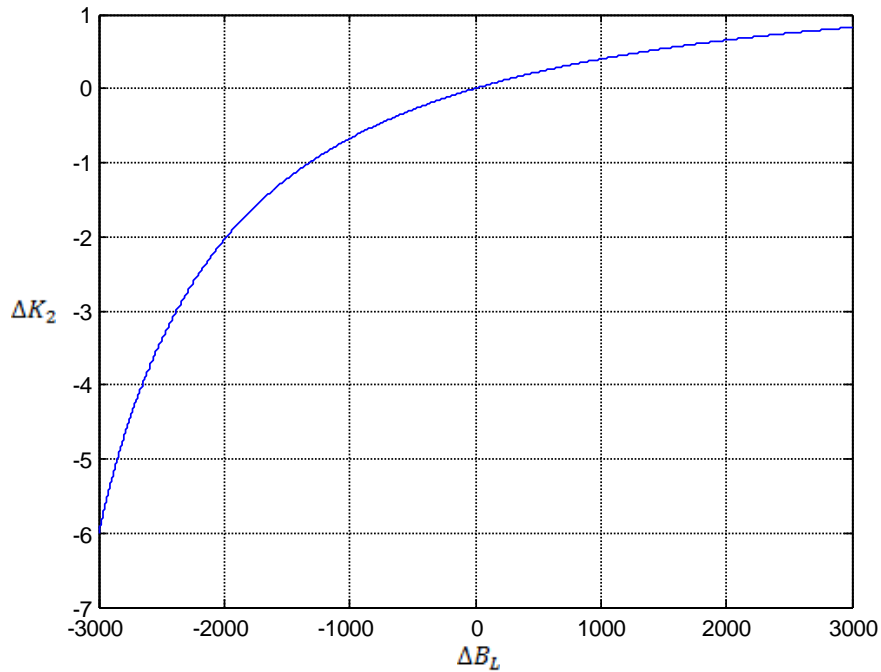


Figura 6.5 - Excursão do ganho integral do controlador em relação ao loop bandwidth.

### 6.3 Modelagem do controlador *fuzzy*

O diagrama esquemático do controlador *fuzzy* a ser utilizado no *loop filter* do FAPLL está ilustrado na Figura 6.6. Basicamente, o controlador com estratégia *fuzzy* supervisorio pode ser dividido em 3 (três) etapas distintas:

- Etapa de *fuzzificação*
- Composição da inferência a partir de uma tabela de regras
- Processo de *defuzzificação*

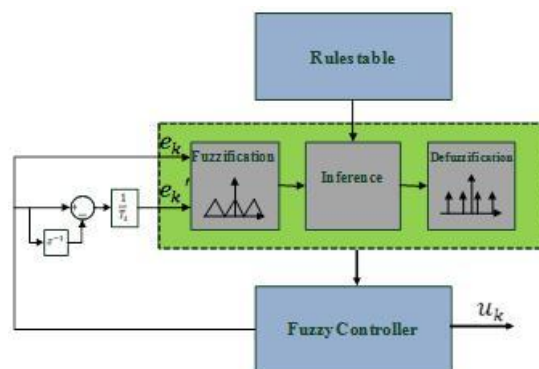


Figura 6.6 – Diagrama de blocos do controlador *fuzzy* proposto para o Loop Filter do FAPLL.

### 6.3.1 Fuzzificação

A *fuzzificação* consiste, geralmente, da primeira etapa a ser implementada em um sistema baseado em lógica *fuzzy*. Após a pré-modelagem do sistema, como descrito na seção 6.2, faz-se necessário aplicar uma transformação linear de modo inserir as variáveis de controle ( $e_k, e_k'$  e  $u_k$ ), em um *universo de discurso fuzzy*. O universo de discurso de uma variável representa o intervalo numérico de todos os possíveis valores que uma função específica pode assumir.

Trata-se, portanto, de inserir tais variáveis dentro de um contexto linguístico, onde a todo instante, cada uma dessas variáveis estará incluída dentro de uma função de pertinência (negativo grande, positivo pequeno, por exemplo) e possuirá um grau de pertinência variando de 0 (zero) a 1 (um), significando possibilidade nula ou total, respectivamente, de estar contido dentro daquela função de pertinência.

Para o controlador *fuzzy* supervisor, foram utilizados 5 (cinco) *sets* triangulares e 2 (dois) *sets* trapezoidais, em um total de 7 (sete) *sets fuzzy* tanto para a variável de erro (entrada) do controlador *fuzzy*  $e_k$  quanto para a derivada do erro  $e_k'$ . A função que descreve a derivada do erro é obtida em (6.11).

$$e_k' = \frac{e_k - z^{-1}e_k}{T_s} \quad (6.11)$$

As funções de pertinência do controlador se dividem da seguinte forma:

- Zero (ZE)
- Positivo pequeno (PS)
- Positivo médio (PM)
- Positivo grande (PB)
- Negativo pequeno (NS)
- Negativo médio (NM)
- Negativo grande (NB)

A partir da Figura 6.7, é possível verificar que a *fuzzificação* abrange todo o universo de discurso, de  $-T$  a  $T$ . O grau de superposição utilizado foi de 50%. Ou seja, uma função com grau de pertinência igual a 80% em PS também irá apresentar grau de pertinência igual a 20% de ZE ou de PM.

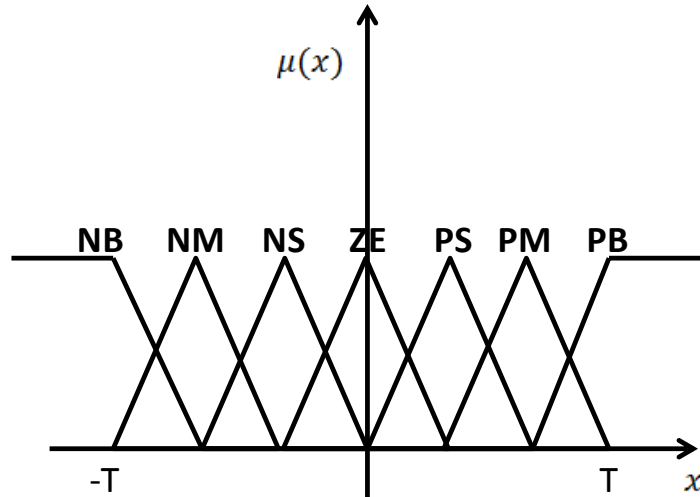


Figura 6.7 – Funções de pertinência do erro e derivada do erro.

### 6.3.2 Composição da inferência a partir de uma tabela de regras

Após a *fuzzificação* do sistema, é necessário caracterizar a ação do controlador supervisor *fuzzy* do PLL. Para isto, utiliza-se uma tabela de regras responsável por determinar qual deverá ser a ação do sistema supervisor na variação da saída  $u_k$  do controlador *fuzzy*.

A tabela de regras proposta possuiria, portanto, 49 (quarenta e nove) regras heurísticas. Ou seja, compara-se o grau de pertinência de cada função de pertinência da *fuzzificação* do erro com o análogo da derivada do erro, obtendo-se, portanto, o menor entre os dois. Esta composição é caracterizada como *mín-max*. De forma a detalhar o procedimento relatado acima, será exposto um pequeno exemplo de apenas duas regras.

Observação: Ao longo do trabalho, observou-se que a eficiência de 5 (cinco) funções de pertinência *fuzzy* para a entrada  $e_k$  e 5 (cinco) funções de pertinência *fuzzy* para a entrada  $e_k'$ , contabilizando-se 25 (vinte e cinco) regras heurísticas, foi plenamente satisfatória, com praticamente metade do custo computacional. Por esta razão, a Tabela 6.1 foi baseada em uma matriz 5x5, obtendo-se ainda 7 (sete) funções de pertinência de saída  $u_k$ .

Exemplo: Regra 1: SE erro for ZERO e a derivada do erro for NEGATIVO GRANDE, ENTÃO, saída será POSITIVO PEQUENO.

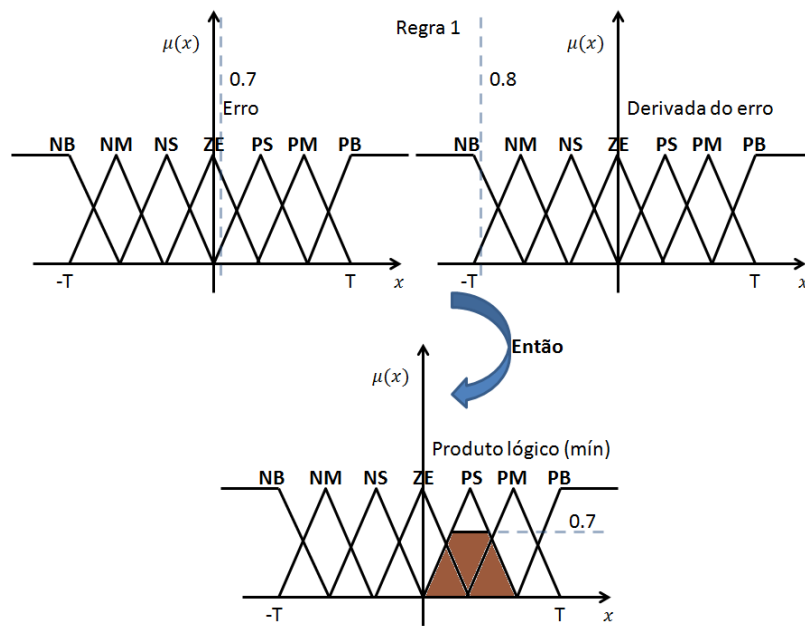


Figura 6.8 – Produto lógico fuzzy proveniente da regra 1.

Regra 2: SE erro for NEGATIVO PEQUENO e a derivada do erro for NEGATIVO GRANDE, ENTÃO, saída será POSITIVO MÉDIO.

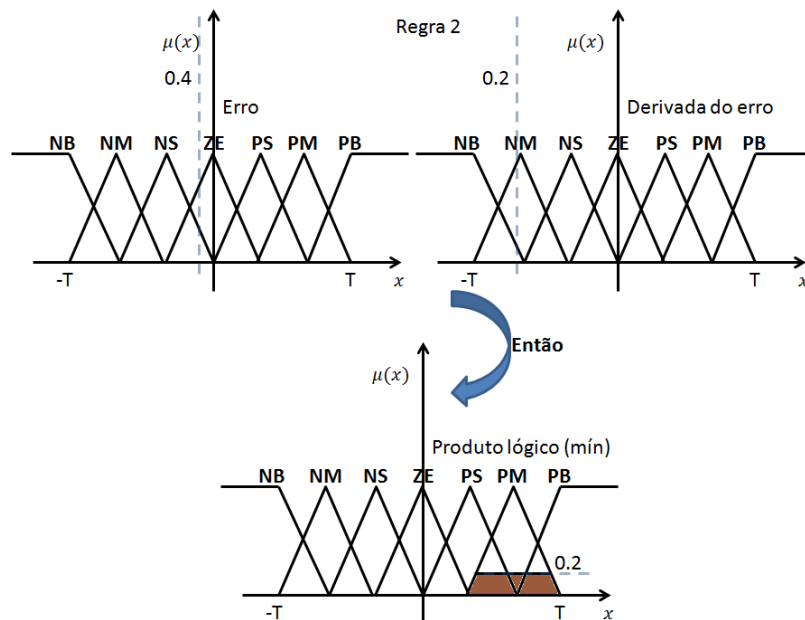


Figura 6.9 – Produto lógico fuzzy proveniente da regra 2.

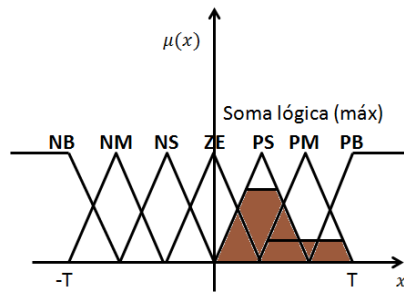


Figura 6.10 – Soma lógica (máx) fuzzy da inferência mín-máx.

Deste modo, em seguida, deve ser aplicado o processo da *defuzzificação*, para que o valor da variável linguística de saída inferida pelas regras *fuzzy* seja traduzido em um valor discreto. O objetivo é obter um único valor numérico que represente os valores *fuzzy* inferidos da variável linguística de saída, ou seja, a distribuição de possibilidades. Desta forma, a *defuzzificação* é uma transformação inversa que traduz a saída do domínio *fuzzy* para o domínio *crisp* (discreto).

### 6.3.3 Defuzzificação do sistema a partir de uma tabela de regras

A tabela de regras descrita na seção anterior pode ser analisada a partir da Tabela 6.1. Esta tabela foi implementada a partir da experiência adquirida ao longo deste trabalho e representa a base de inteligência do algoritmo *fuzzy* supervisor.

Basicamente, a partir da Tabela 6.1, ditam-se as regras de funcionamento do controlador supervisor. É interessante ressaltar que, em circunstâncias normais, o sistema supervisor baseado em regras torna o controlador mais rápido. Em instantes de grandes variações harmônicas e sub-harmônicas, o sistema supervisor torna o controlador *fuzzy* mais lento, de modo a estabelecer uma melhor filtragem, tornando a saída do FAPLL mais próxima da referência fundamental.

Tabela 6.1 – Tabela de regras do FAPLL.

Rate of error ( $e_k'$ )	PB	NM	NS	PM	PB	PB
	PS	NM	NS	PS	PM	PB
	ZE	NM	NM	ZE	PM	PM
	NS	NB	NM	NS	PS	PM
	NB	NB	NB	NM	PS	PM
$u_k$	NB	NS	ZE	PS	PB	
	Error ( $e_k$ )					

A partir da *defuzzificação* do sistema, é possível obter o quanto se deve excursionar o valor de  $u_k$ . A Figura 6.11 mostra claramente a excursão de  $u_k$  a partir do valor *fuzzificado* da variável erro  $e_k$  e derivada do erro  $e_k'$ .

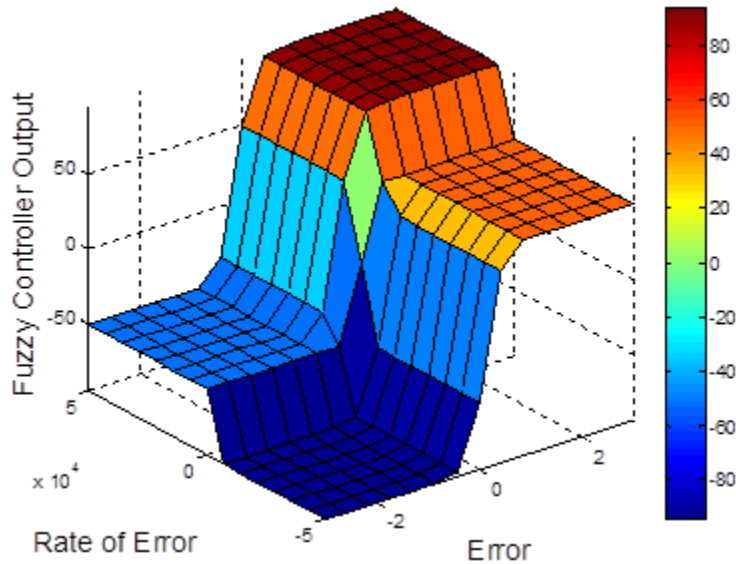


Figura 6.11 – Excursão da variável  $u_k$  a partir do valor *fuzzificado* de  $e_k$  e  $e_k'$ .

De modo a obter um valor discreto de  $u_k$  a partir da *fuzzificação* dos valores de erro e de sua derivada, faz-se necessário utilizar um processo de *defuzzificação*. Utilizou-se a *defuzzificação* conhecida por *centro do máximo*. Neste método, os picos das funções de pertinência representados no universo de discurso da variável de saída são usados, enquanto ignora-se as áreas das funções de pertinência. As contribuições múltiplas de regras são consideradas por este método. É proposta neste trabalho a utilização de sete *sets singleton* no universo de discurso da saída, como apresentado na Figura 6.12. A saída a partir da *defuzzificação pelas alturas* pode ser obtida por:

$$u^* = \frac{\sum_{i=1}^N u_i \sum_{k=1}^N \mu_{saída}(u_i)}{\sum_{i=1}^N \sum_{k=1}^N \mu_{saída}(u_i)} \quad (6.12)$$

onde  $u_i$  é o valor de cada função de pertinência *singleton* de saída e  $\mu_{saída}(u_i)$  é a pertinência referente a cada um dos índices da Tabela 6.1. Ou seja, a saída discreta é calculada como uma média ponderada dos máximos dos *singletons*, cujos pesos são os resultados da inferência *máx-mín* utilizada.



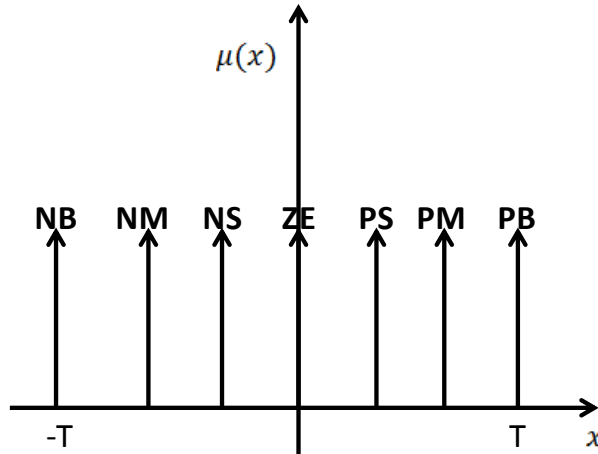


Figura 6.12 – Funções de pertinência singleton de saída.

#### 6.4 Simulação do modelo linearizado de FAPLL

Assim como realizado na seção 4.3, faz-se necessário verificar a resposta dinâmica do modelo linear do FAPLL proposto a partir da inserção de perturbações como degraus de fase, harmônicos e sub-harmônicos. De modo a validar a eficácia de seu comportamento, optou-se por comparar os resultados obtidos para o FAPLL com os obtidos a partir dos sistemas com ganhos ótimos apresentados no Capítulo 4. Desta forma, espera-se demonstrar a superioridade do FAPLL perante os demais PLLs apresentados neste trabalho.

##### 6.4.1 Comparações do modelo linearizado do FAPLL com o pPLL/parkPLL e EPLL linear

Para a modelagem digital do *pPLL/parkPLL* e *EPLL*, bem como do FAPLL, serão utilizados os seguintes parâmetros:

- Mapeamento com aproximação de *Tustin*
- Frequência de amostragem  $f_s = 10020 \text{ Hz}$

A comparação será efetuada a partir dos seguintes testes:

- Degrau de  $30^\circ$  (0.523 rad) em  $t = 0.0 \text{ s}$ .

- Degrau de  $15^\circ$  (0.261 rad) aliado a 10% de perturbação de 2º harmônico em  $t = 0.5$  s.
- Degrau de  $-30^\circ$  (-0.523 rad) aliado a 10% de perturbação sub-harmônica de 50 Hz em  $t = 1.0$  s.

Será comparada a resposta dinâmica do modelo linear do FAPLL com as configurações de *pPLL/parkPLL* e *EPLL*. Primeiramente, serão utilizados o *pPLL/parkPLL* e *EPLL* com os ganhos ótimos obtidos no Capítulo 3. Em seguida, utiliza-se o *Fuzzy PLL* sintonizado para o mesmo tempo de assentamento dos algoritmos supracitados.

A Figura 6.13 mostra a comparação direta entre o FAPLL e o *pPLL/parkPLL* e *EPLL*. É possível verificar que, em  $t = 0.0$ s, quando não há perturbações, a resposta do FAPLL é mais rápida do que de ambos *pPLL/parkPLL* e *EPLL*, além de não possuir *overshoot*.

No instante  $t = 0.25$ s, além do degrau de fase, aplicou-se 10% de perturbação harmônica de 120 Hz. Como é possível verificar pelo *zoom* presente na segunda coluna da Figura 6.13, a resposta do FAPLL foi superior mesmo ao *pPLL/parkPLL*, os quais obtiveram os -20 dB de atenuação propostos. Em  $t = 0.50$ s, aplicou-se, aliado ao degrau de fase, 10% de perturbação sub-harmônica de 50 Hz. Novamente a resposta do FAPLL foi superior às demais, como pode ser analisado pelo *zoom* na terceira coluna obtendo-se o dobro de atenuação para o sub-harmônico, caracterizando assim, a capacidade adaptativa do *Loop Filter* do FAPLL de se adequar a diversos tipos de perturbação.

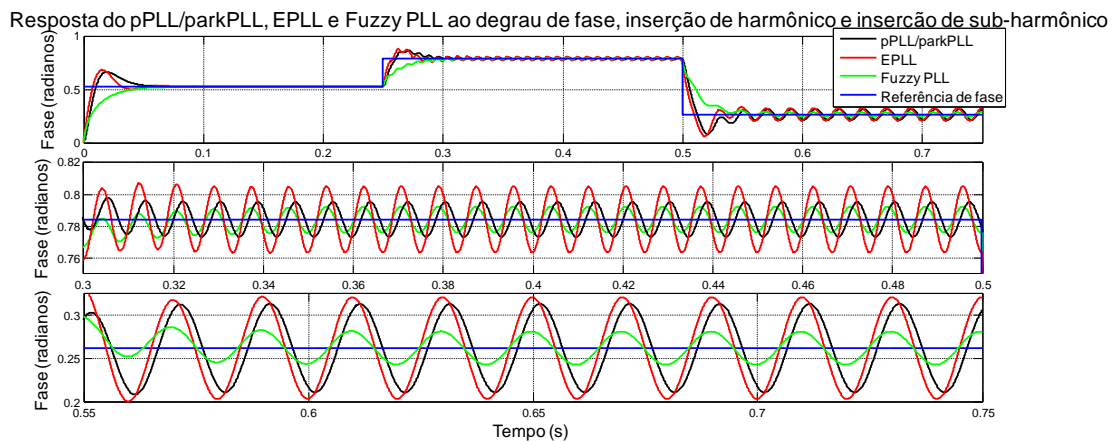


Figura 6.13 – Comparação entre FAPLL e demais PLLs otimamente sintonizados.

## 6.5 Simulação do modelo digital de FAPLL

Assim como realizado na seção 4.3, faz-se necessário verificar a resposta dinâmica do modelo digital do FAPLL proposto a partir da inserção de perturbações como degraus de fase, harmônicos e sub-harmônicos. De modo a validar a eficácia de seu comportamento, optou-se por comparar os resultados obtidos para o FAPLL com os obtidos a partir dos sistemas com ganhos ótimos apresentados no Capítulo 4.

Na modelagem digital do FAPLL, serão utilizados os parâmetros da seção 6.4.1. Além disto, as comparações obtidas serão efetuadas nos moldes das seções 4.3.5, 4.3.6 e 4.3.7.

### 6.5.1 Resposta ao degrau de fase para o modelo digital do FAPLL

Para esta análise foi verificada a resposta do *FAPLL* a um degrau de  $30^\circ$  em sua entrada para  $t = 0.15$  s. É possível verificar que, em aproximadamente 50 ms, tal como os demais PLLs, tanto o algoritmo linearizado quanto o digital efetuam o sincronismo, confirmando que a modelagem linear também é válida para o FAPLL. Desta forma, conclui-se que, como esperado, o modelo linearizado prevê com segurança o tempo de assentamento e dinâmica do erro do controlador *fuzzy*.

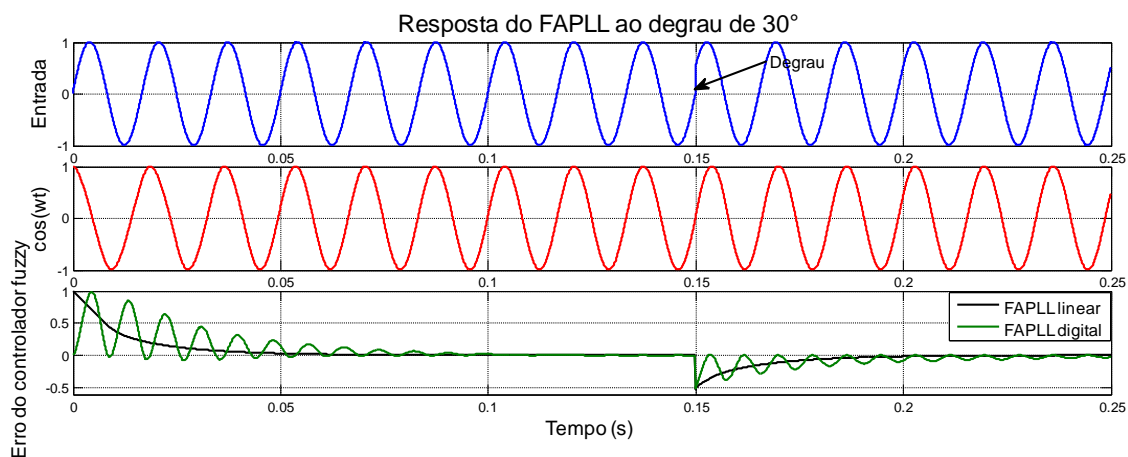


Figura 6.14 – Resposta do FAPLL ao degrau de  $30^\circ$  em  $t = 0.15$ s.

### 6.5.2 Resposta à inserção de harmônicos para o modelo digital do FAPLL

Para esta análise, verificou-se a resposta do *FAPLL* a inserção de 1.0 p.u de perturbação harmônica de 120 Hz em  $t = 0.15s$ . A Figura 6.15 mostra que, assim como esperado pela seção 4.3.6, o *FAPLL* melhorou ainda mais a atenuação para perturbações harmônicas, ultrapassando os -20 db do *pPLL/parkPLL*. Novamente, é possível concluir que os resultados para o modelo linearizado do *FAPLL* foram coerentes com a realidade.

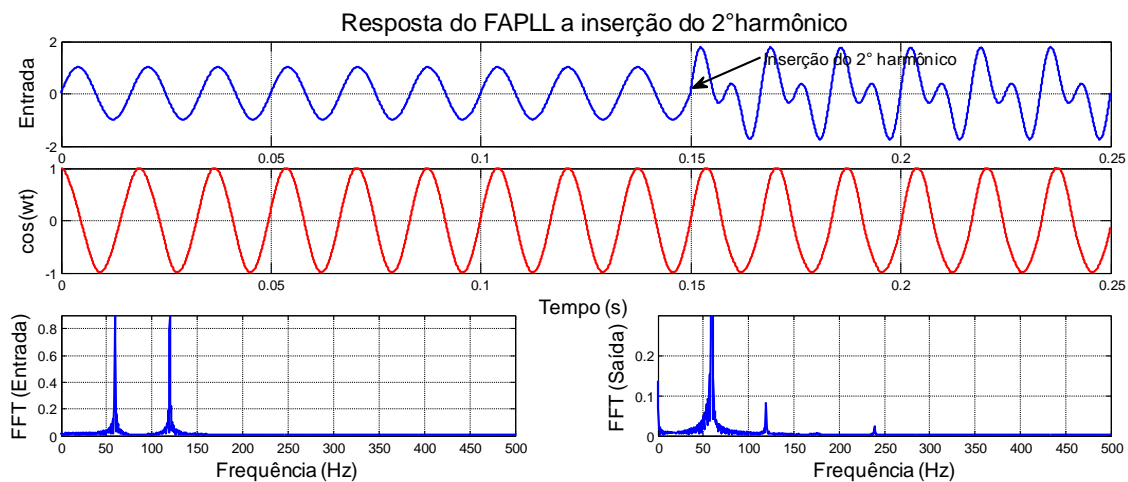


Figura 6.15 - Resposta do *FAPLL* a inserção de 1.0 p.u de 2º harmônico em  $t = 0.15s$ .

### 6.5.3 Resposta à inserção de sub-harmônicos para o modelo digital do FAPLL

Para esta análise, verificou-se a resposta do *FAPLL* a inserção de 1.0 p.u de perturbação sub-harmônica de 50 Hz em  $t = 0.15s$ . A Figura 6.16 mostra que, assim como esperado pela seção 4.3.7, o *FAPLL* melhorou ainda mais a atenuação para perturbações sub-harmônicas, assim como visto no modelo linearizado na Figura 6.13. Novamente, é possível concluir que os resultados para o modelo linearizado do *FAPLL* foram coerentes com a realidade.

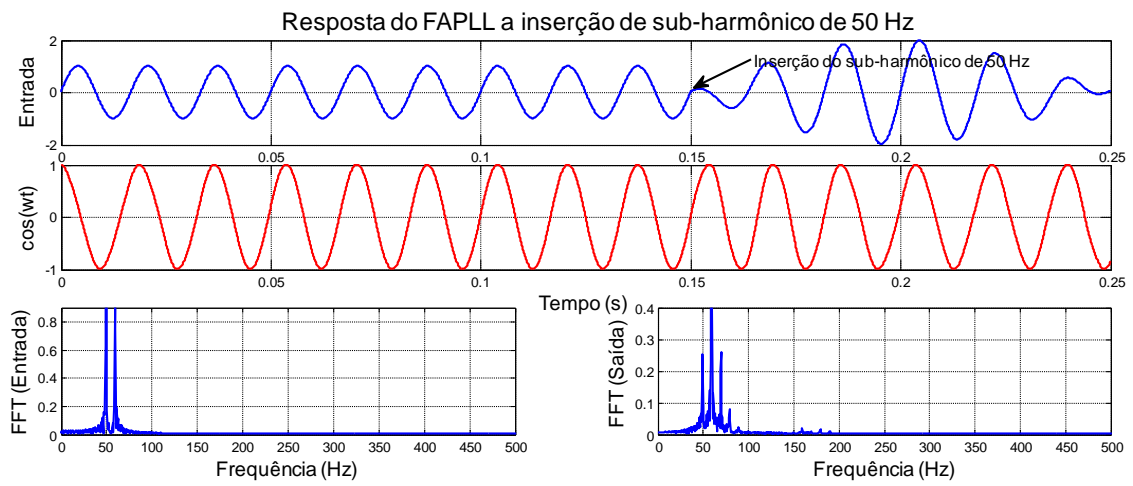


Figura 6.16 - Resposta do FAPLL a inserção de 1.0 p.u de sub-harmônico de 50 Hz em  $t = 0.15$ s.

## 7. Conclusões e Trabalhos Futuros

### 7.1 Conclusões gerais

O principal objetivo deste trabalho foi o estudo de algoritmos *Phase-Locked Loop* monofásicos baseados em controladores proporcional integral bem como seus processos de sintonia para obtenção de ganhos ótimos.

Foi desenvolvida uma metodologia para sintonia dos ganhos proporcional e integral do controlador PI do PLL, bem como as constantes de tempos dos filtros existentes no *pPLL* e *parkPLL*. A partir desta metodologia, é possível obter um sistema de equações (a ser resolvido em um *software* de análise matemática como MATLAB) que garante um sistema com a máxima margem de fase possível, aliado a um nível predeterminado de atenuação para o 2º harmônico.

Após a determinação dos ganhos ótimos utilizando-se da metodologia supracitada, analisou-se o comportamento dinâmico de cada um dos PLLs estudados a partir da inserção de anomalias tais como: degraus de fase, degraus de frequência, perturbações harmônicas e perturbações sub-harmônicas. Constatou-se, portanto, que tais algoritmos, apesar de otimamente sintonizados, não são completamente imunes a tais anomalias. Pelo contrário, verificou-se, por exemplo, que a resposta a perturbações sub-harmônicas próximas ao final da faixa de passagem (neste caso, 50 Hz) era extremamente oscilatória para o *pPLL*, *parkPLL* e *EPLL*.

Desta forma, fez-se necessário averiguar, a partir da teoria clássica de controle, a causa de tais oscilações. Encontrado o motivo, propôs-se a adequação de forma a corrigir este problema. Entretanto, apesar de tal adequação ser suficiente em nível de regime permanente, seu transitório apresenta uma dinâmica extremamente deficitária.

Portanto, a partir deste trabalho, verificou-se a necessidade de implementar um algoritmo *Phase-Locked Loop* baseado em teoria adaptativa/supervisória para que tal algoritmo realize seu auto ajuste de forma automática e dinâmica. Ou seja, implementou-se uma estratégia *fuzzy* supervisória visando a modificação em tempo real da saída do *Loop Filter* de forma a sempre proporcionar uma resposta que integre boa dinâmica e alta imunidade a ruídos.

A partir dos resultados obtidos ainda com o modelo linearizado do FAPLL, ficou claro que uma estratégia supervisória *fuzzy* é extremamente interessante no tocante à adaptação do sistema para diferentes casos de contingência. Além disto, devido à natureza subjetiva da tabela de regras *fuzzy*, é possível obter uma modelagem do comportamento do controlador supervisor que se adéque a necessidade específica de cada projeto de PLL.

## 7.2 Trabalhos futuros

Possíveis trabalhos futuros compreendem, por exemplo, a necessidade de averiguar a consequência da utilização de outros métodos de inferência, tais como *mín-produto* no controlador supervisor *fuzzy*. Além disto, uma modelagem mais precisa do sistema através do uso de *sets* sigmóides como funções de pertinência, bem como a utilização de graus de superposição diferentes de 50% podem gerar resultados ainda melhores para o FAPLL.

Outra possível melhoria no FAPLL compreende o seu processo de aprendizagem. A partir da utilização de algoritmos *neuro-fuzzy*, é possível garantir ao sistema a possibilidade de aprendizado contínuo, atribuindo à tabela de regras *fuzzy* a capacidade de autoajuste com o tempo, tornando o FAPLL ainda mais robusto perante perturbações diversas.

## Referências Bibliográficas

- [1] D. Jovcic, “Phase locked loop system for FACTS,” *Power Systems, IEEE Transactions on*, vol. 18, n. 3, pp. 1116-1124, 2003.
- [2] S. Golestan, M. Monfared, F. Freijedo e J. Guerrero, “Design and Tuning of a Modified Power-Based PLL for Single-Phase Grid-Connected Power Conditioning Systems,” *Power Electronics, IEEE Transactions on*, vol. 27, n. 8, pp. 3639-3650, 2012.
- [3] S. Silva, B. Lopes, B. J. C. Filho, R. Campana e W. Bosventura, “Performance evaluation of PLL algorithms for single-phase grid-connected systems,” em *Industry Applications Conference, 2004. 39th IAS Annual Meeting. Conference Record of the 2004 IEEE*, 2004.
- [4] W. Li e J. Meiners, “Introduction to Phase-Locked Loop System Modeling,” 2005.
- [5] M. Karimi-Ghartemani, S. Khajehoddin, P. Jain, A. Bakhshai e M. Mojiri, “Addressing DC Component in PLL and Notch Filter Algorithms,” *Power Electronics, IEEE Transactions on*, vol. 27, n. 1, pp. 78-86, 2012.
- [6] M. Karimi-Ghartemani, M. Mojiri, A. Safaee, J. Walseth, S. Khajehoddin, P. Jain e A. Bakhshai, “A New Phase-Locked Loop System for Three-Phase Applications,” *Power Electronics, IEEE Transactions on*, vol. 28, n. 3, pp. 1208-1218, 2013.
- [7] C. Shan, Z. Chen, Y. Li e H. Yuan, “All DPLLs based on fuzzy PI control algorithm,” em *Mechanic Automation and Control Engineering (MACE), 2011 Second International Conference on*, 2011.
- [8] W. Lindsey e C. M. Chie, “A survey of digital phase-locked loops,” *Proceedings of the IEEE*, vol. 69, n. 4, pp. 410-431, 1981.
- [9] S. C. Gupta, “Phase-locked loops,” *Proceedings of the IEEE*, vol. 63, n. 2, pp. 291-306, 1975.
- [10] D. Abramovitch, “Phase-locked loops: a control centric tutorial,” em *American Control Conference, 2002. Proceedings of the 2002*, 2002.
- [11] R.E.Best, *Phase-Locked Loops: Design, Simulation and Applications*, New York: McGraw-Hill, 1997.
- [12] J.A.Crawford, *Frequency Synthesizer Design Handbook*, 1994.
- [13] K. Ogata, *Engenharia de Controle Moderno*, Pearson Brasil, 2011.



- [14] K. J. Astrom, Automatic Tuning of PID Controllers, Instrument Society of America, 1988.
- [15] B. C. Kuo, Automatic Control Systems, Prentice-Hall, 1995.
- [16] M. Iravani e M. Karimi-Ghartemani, "A method for synchronization of power electronic converters in polluted and variable-frequency environments," *Power Systems, IEEE Transactions on*, vol. 19, n. 3, pp. 1263-1270, 2004.
- [17] A. Timbus, M. Liserre, R. Teodorescu e F. Blaabjerg, "Synchronization methods for three phase distributed power generation systems - An overview and evaluation," em *Power Electronics Specialists Conference, 2005. PESC '05. IEEE 36th*, 2005.
- [18] M.H.J. Bollen, Understanding Power Quality Problems: Voltage Sags and Interruptions, IEEE Press, 2000.
- [19] R. Santos Filho, P. Seixas, P. Cortizo, L. A. B. Torres e A. Souza, "Comparison of Three Single-Phase PLL Algorithms for UPS Applications," *Industrial Electronics, IEEE Transactions on*, vol. 55, n. 8, pp. 2923-2932, 2008.
- [20] L. Rolim, D. da Costa e M. Aredes, "Analysis and Software Implementation of a Robust Synchronizing PLL Circuit Based on the pq Theory," *Industrial Electronics, IEEE Transactions on*, vol. 53, n. 6, pp. 1919-1926, 2006.
- [21] V. Kaura e V. Blasko, "Operation of a phase locked loop system under distorted utility conditions," *Industry Applications, IEEE Transactions on*, vol. 33, n. 1, pp. 58-63, 1997.
- [22] R. M. Palhares. [Online]. Available: [http://www.cpdee.ufmg.br/~palhares/aula6\\_csl.pdf](http://www.cpdee.ufmg.br/~palhares/aula6_csl.pdf). [Acesso em 2 Julho 2013].
- [23] L. F. A. Pereira, "Universidade Federal do Rio Grande do Sul - UFRGS," [Online]. Available: <http://www.ece.ufrgs.br/~eng04006/aulas/aula29.pdf>. [Acesso em 08 Julho 2013].
- [24] Preitl.S e Precup.R.E, "An extension of tuning relations after symmetrical optimum method for PI and PID controller," *Automatica*, pp. 1731-1736, 1999.
- [25] R. R. W. C. Wilson Arnaldo Artuzi Junior, "Introdução à Transformada Z," 2010. [Online]. Available: <http://www.eletrica.ufpr.br/artuzi/te043/capitulo6.pdf>. [Acesso em 11 Agosto 2013].
- [26] W. F. Lages, "Aproximações Discretas para Funções de Transferência Contínuas," Rio Grande do Sul, 2011.

- [27] K. Pal, R. Mudi e N. Pal, “A new scheme for fuzzy rule-based system identification and its application to self-tuning fuzzy controllers,” *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 32, n. 4, pp. 470-482, 2002.
- [28] A. Amer, E. A. Sallam e W. M. Elawady, “Fuzzy pre-compensated fuzzy self-tuning fuzzy PID controller of 3 DOF planar robot manipulators,” em *Advanced Intelligent Mechatronics (AIM), 2010 IEEE/ASME International Conference on*, 2010.
- [29] Q. Li e D. Shen, “A New Incremental Fuzzy PD+Fuzzy ID Fuzzy Controller,” em *Information Engineering, 2009. ICIE '09. WASE International Conference on*, 2009.
- [30] E. Bolat e H. Ertunc, “Control of single phase UPS inverter using Fuzzy Gain Scheduling of PI Controller,” em *Electrical and Electronics Engineering (ELECO), 2011 7th International Conference on*, 2011.
- [31] I. S. Marcelo Godoy Simões, *Controle e Modelagem Fuzzy*, São Paulo: Blucher, 2007.
- [32] B. Chatterjee, S. Ray e B. Biswas, “Performance enhancement in a fuzzy controlled digital phase locked loop,” em *Communications, Devices and Intelligent Systems (CODIS), 2012 International Conference on*, 2012.
- [33] Y. Chen, H. Zheng e Y. Wang, “Adaptive bandwidth PLL design based on fuzzy logic control,” em *Microwave, Antenna, Propagation, and EMC Technologies for Wireless Communications (MAPE), 2011 IEEE 4th International Symposium on*, 2011.
- [34] Agilent, “Loop Bandwidth and Clock Data Recovery (CDR) in Oscilloscope Measurements,” 2002. [Online]. Available: <http://literature.agilent.com/litweb/pdf/5988-6682EN.pdf>. [Acesso em 12 Agosto 2013].
- [35] F. Meneghetti, Março 2003. [Online]. Available: <http://www.dca.ufrn.br/~meneghet/FTP/Controle1/Controle%20I%20-%20Apostila.pdf>. [Acesso em 1 Junho 2013].
- [36] M. Karimi-Ghartemani, B.-T. Ooi e A. Bakhshai, “Application of Enhanced Phase-Locked Loop System to the Computation of Synchrophasors,” *Power Delivery, IEEE Transactions on*, vol. 26, n. 1, pp. 22-32, 2011.
- [37] P.V.Brennan, *Phase-Locked Loops: Principles and Practice*, New York: McGraw Hill, 1996.

[38] “ENG04006 Sistemas e Sinais,” [Online]. Available:  
<http://www.ece.ufrgs.br/~eng04006/aulas/aula29.pdf>. [Acesso em 08 Julho 2013].

# I. Anexo 1

## a. Sistemas de primeira ordem

Como analisado em [35], considerando-se a equação diferencial de primeira ordem descrita abaixo:

$$a \frac{d}{dt} c(t) + bc(t) = dr(t) \quad a \neq 0 \quad (1.1)$$

onde  $c(t)$  é a saída do sistema e  $r(t)$  é a entrada do sistema..

Definindo-se:

$$\frac{a}{b} = T \quad (1.2)$$

$$\frac{d}{b} = K \quad (1.3)$$

tem-se que:

$$T \frac{d}{dt} c(t) + c(t) = Kr(t) \quad (1.4)$$

onde T e K são a constante de tempo e o ganho do sistema, respectivamente.

Desta forma, tem-se que a função de transferência é dada por:

$$\frac{C(s)}{R(s)} = G(s) = \frac{K}{Ts + 1} \quad (1.5)$$

Resposta ao degrau unitário:

Considerando-se a entrada  $R(s) = \frac{1}{s}$ , a qual é a transformada de *Laplace* do degrau unitário e sendo o ganho do sistema  $K = 1$ , chega-se a:

$$C(s) = \frac{1}{sT + 1} \times \frac{1}{s} = \frac{1}{s} - \frac{1}{s + \frac{1}{T}} \quad (1.6)$$

Aplicando-se a transformação inversa de *Laplace*, é possível estabelecer que:

$$c(t) = 1 - e^{-\frac{t}{T}} \quad (1.7)$$

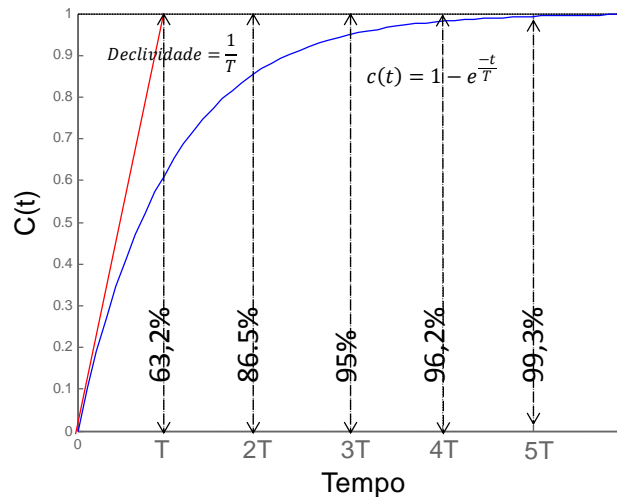


Figura I.1 - Resposta ao degrau de um sistema de primeira ordem de ganho  $K=1$ .

Resposta a rampa unitária:

Outra forma bastante comum de verificação de um sistema é a partir da sua resposta a uma rampa unitária. A partir desta, é possível verificar o erro de estado estacionário do sistema em questão.

Lembrando-se que a transformada de *Laplace* da rampa unitária é dada por  $R(s) = \frac{1}{s^2}$ . Portanto, a resposta a rampa unitária do sistema de primeira ordem em questão é dado por:

$$C(s) = \frac{1}{s^2} \times \frac{1}{Ts + 1} \quad (1.8)$$

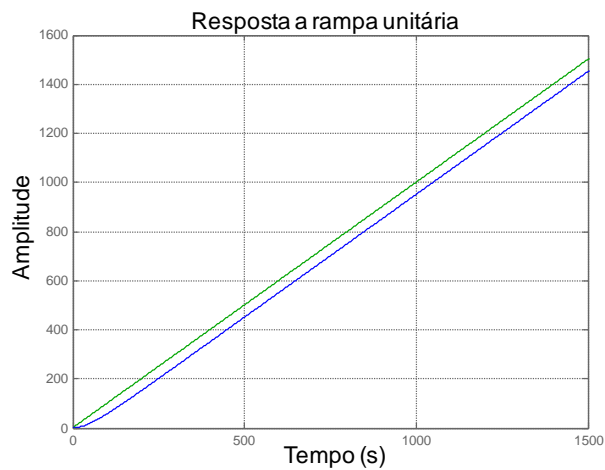
Aplicando-se a transformação inversa de *Laplace*, é possível estabelecer que:

$$c(t) = t - T + Te^{\frac{-t}{T}} \quad (1.9)$$

Desta forma, o erro de estado estacionário pode ser calculado da seguinte maneira:

$$e(t) = r(t) - c(t) = T \left( 1 - e^{\frac{-t}{T}} \right) \quad (1.10)$$

$$e(\infty) = T \quad (1.11)$$



*Figura 1.2 – Resposta a rampa unitária de uma função de primeira ordem com  $K=1$  e  $T=50$  (azul). Em verde, a rampa unitária de referência.*

## **b. Sistemas de segunda ordem**

Novamente, como analisado em [35], considerando-se a equação diferencial de primeira ordem descrita abaixo:

$$a \frac{d^2}{dt^2} c(t) + b \frac{d}{dt} c(t) + dc(t) = er(t) \quad (1.12)$$

Definindo-se:

$$\frac{b}{a} = 2\xi\omega_n \quad (I.13)$$

$$\frac{d}{a} = \omega_n^2 \quad (I.14)$$

$$\frac{e}{a} = K \quad (I.15)$$

onde  $\xi$  é conhecido como o fator de amortecimento,  $\omega_n$  é a frequência natural de oscilação e  $K$  é o ganho do sistema.

Desta forma, substituindo-se (I.13), (I.14), (I.15) em (I.12) e aplicando-se a transformada de Laplace com condições iniciais nulas e  $K = \omega_n^2$ , chega-se a:

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (I.16)$$

que é a função de transferência de um sistema de controle de segunda ordem.

Sistemas de segunda ordem podem ser divididos em três casos distintos:

Caso  $0 < \xi < 1$ : O sistema possui dois pólos complexos conjugados e apresenta oscilações. É conhecido como *subamortecido*.

Caso  $\xi = 1$ : O sistema apresenta dois pólos reais e iguais. A partir deste valor de  $\xi$ , o sistema passa a não ter mais oscilações. É conhecido como *criticamente amortecido*.

Caso  $\xi > 1$ : O sistema apresenta dois pólos reais e distintos. A medida que  $\xi$  aumenta, o comportamento do sistema se aproxima do comportamento de um sistema de primeira ordem. É conhecido como *sobreamortecido*.

Para cada um destes casos, a resposta temporal a uma entrada (por exemplo, degrau unitário) será diferente, como analisado abaixo:

Resposta ao degrau unitário:

- Caso subamortecido:

$$c(t) = 1 - \frac{e^{-\xi\omega_n t}}{\sqrt{1-\xi^2}} \operatorname{sen} \left( \omega_d t + tg^{-1} \frac{\sqrt{1-\xi^2}}{\xi} \right) \quad (1.17)$$

onde  $\omega_d = \sqrt{1-\xi^2}$  é a frequência natural amortecida.

- Caso criticamente amortecido:

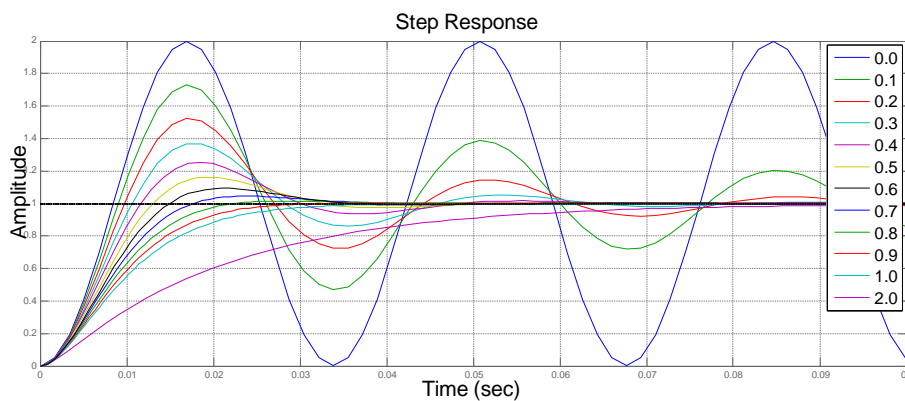
$$c(t) = 1 - e^{-\omega_n t} (1 + \omega_n t) \quad (1.18)$$

- Caso sobreamortecido:

$$c(t) = 1 + \frac{\omega_n}{2\sqrt{\xi^2-1}} \left( \frac{e^{-s_1 t}}{s_1} - \frac{e^{-s_2 t}}{s_2} \right) \quad (1.19)$$

onde  $s_1 = (\xi + \sqrt{\xi^2-1})\omega_n$  e  $s_2 = (\xi - \sqrt{\xi^2-1})\omega_n$ .

A Figura I.3 demonstra como as curvas de resposta ao degrau unitário de um sistema de segunda ordem se comportam em função do fator de amortecimento  $\xi$ .



*Figura I.3 - Comportamento da resposta ao degrau de uma função de segunda ordem em função do fator de amortecimento  $\xi$ .*

**c. Especificações de resposta transitória**

Em muitos casos práticos, as características de desempenho desejadas de sistemas de controle são especificadas em termos de grandezas no domínio do tempo.



Sistemas com armazenamento de energia não podem responder instantaneamente e, portanto, terão respostas transitórias sempre que submetidos a perturbações e excitações [13].

Em geral, as características de desempenho de um sistema de controle são específicas em termos da resposta transitória a uma entrada (excitação) em degrau unitário, pois o mesmo é muito fácil de ser gerado e corresponde a uma solicitação suficientemente severa. Basicamente, a partir da resposta de um sistema ao degrau, é matematicamente possível computar a resposta para qualquer outro tipo de sinal de entrada.

A resposta transitória de um sistema a uma excitação em degrau unitário depende das condições iniciais. Por uma questão de conveniência na comparação de respostas transitórias de vários sistemas, constitui-se uma praxe usar a condição inicial padrão de que o sistema está inicialmente em repouso com valor nulo da variável de saída e de todas as suas derivadas. Desta forma, as características do sinal de resposta podem ser facilmente comparadas [13].

De fato, a resposta transitória de um sistema de controle geralmente apresenta oscilações amortecidas antes de alcançar o estado de regime permanente. Ao realizar as especificações das características de resposta transitória de um sistema de controle a uma entrada de degrau unitário, é frequente especificar-se as seguintes definições:

- Tempo de atraso,  $t_d$
- Tempo de subida,  $t_r$
- Instante de pico,  $t_p$
- Máximo valor de ultrapassagem,  $M_p$
- Tempo de acomodação,  $t_s$

As especificações definidas acima são mostradas graficamente na Figura L4.

O tempo de atraso,  $t_d$  é o tempo necessário para que a resposta alcance, pela primeira vez, a metade do valor final.

O tempo de subida,  $t_r$  é o tempo necessário para que a resposta passe de 10% a 90%, de 5% a 95% ou de 0% a 100% do seu valor final. No caso de sistemas de segunda ordem subamortecidos, normalmente usa-se o tempo de 0% a 100%. No caso

de sistemas de segunda ordem superamortecidos, o intervalo utilizado é o de 10% a 90%.

$$t_r = \frac{\pi - tg^{-1}\left(\frac{\sqrt{1-\xi^2}}{\xi}\right)}{\omega_d} \quad (1.20)$$

O instante de pico é o tempo necessário para que a resposta alcance o primeiro pico de ultrapassagem.

$$t_p = \frac{\pi}{\omega_d} \quad (1.21)$$

A máxima ultrapassagem é o máximo valor de pico da curva de resposta medido a partir do valor unitário. Quando o valor final de regime estacionário da resposta difere da unidade, é comum usar o conceito da máxima ultrapassagem percentual, definida por:

$$M_p = \frac{c(t_p) - c(\infty)}{c(\infty)} \times 100\% \quad (1.22)$$

$$c(t_p) = 1 + e^{-\left(\frac{\xi\pi}{\sqrt{1-\xi^2}}\right)} \quad (1.23)$$

Substituindo-se (I.23) em (I.22), tem-se que:

$$M_p(\%) = 100e^{-\left(\frac{\xi\pi}{\sqrt{1-\xi^2}}\right)} \quad (1.24)$$

O valor de máxima ultrapassagem indica diretamente a estabilidade relativa do sistema [13].

O tempo de acomodação é o tempo necessário para que a curva de resposta alcance valores dentro de uma faixa em torno do valor final e neste estado permaneça. O tempo de acomodação está associado a uma maior constante de tempo do sistema de controle.

$$t_s = \frac{4}{\xi\omega_n} \quad (1.25)$$

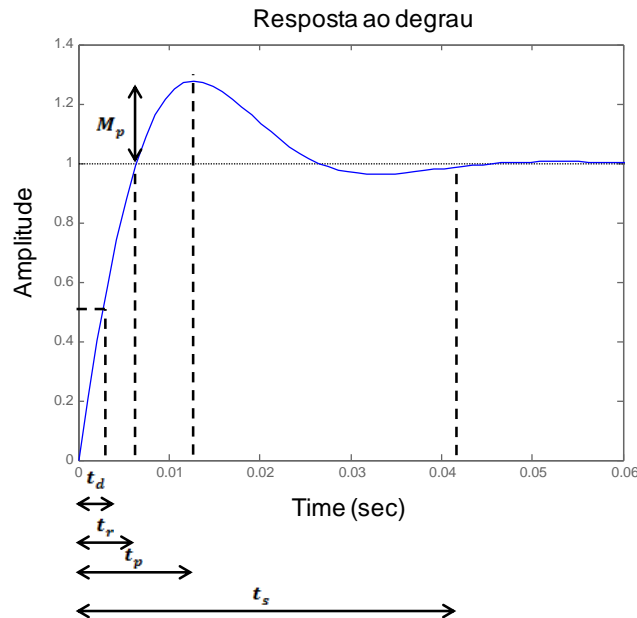


Figura 1.4 – Curva de resposta ao degrau unitário mostrando  $t_d$ ,  $t_r$ ,  $t_p$ ,  $M_p$  e  $t_s$ .

Deve-se lembrar, no entanto, que nem todas as especificações supracitadas se aplicam necessariamente a qualquer caso dado. Para um sistema superamortecido, por exemplo, os critérios *instante de pico* e *máxima ultrapassagem* não se aplicam.

Como será visto mais adiante neste trabalho, exceto em algumas aplicações, onde não se podem tolerar oscilações, é desejável que a resposta transitória seja suficientemente rápida e amortecida. De forma a ter uma resposta transitória aceitável de um sistema de segunda ordem, o coeficiente de amortecimento  $\xi$  deve estar situado entre 0.4 e 0.8, preferencialmente. Valores abaixo desta faixa acarretam em valores de máxima ultrapassagem excessivos na resposta transitória, enquanto um sistema com valor acima desta faixa tende a responder de forma lenta.

Ou seja, sem a utilização de uma estratégia de controle adaptativa, os parâmetros *máxima ultrapassagem* e *tempo de subida* serão especificações sempre conflitantes. Ou seja, não será possível minimizar a máxima ultrapassagem e o tempo de subida simultaneamente.

## II. Anexo 2

### a. Controladores

Iniciando-se de um diagrama de processo tal como o mostrado na Figura II.1, a implementação de um sistema de controle envolve os três seguintes passos:

- Determinação do que o sistema deve efetuar.
- Determinação da configuração do controlador em relação ao processo a ser controlado.
- Determinação dos parâmetros do controlador de modo a atingir os objetivos propostos.

De modo geral, as especificações de critérios de desempenho descrevem o que o sistema de controle deve realizar e como deve realizar. Estas especificações são únicas para cada aplicação individual e, comumente, incluem especificações em relação à estabilidade relativa, erro de estado estacionário, resposta transitória e características de resposta em frequência. Em algumas aplicações, conceitos como o de robustez ou rejeição a perturbações também são especificações adicionais [15].



*Figura II.1 - Diagrama de bloco de um processo.*

Na literatura, é possível encontrar 2 (dois) tipos de controladores: os controladores série e os controladores por realimentação. Comumente, o projeto de controladores série é mais simples que os controladores por realimentação. Têm-se como exemplo de controladores série os controladores lineares proporcional, integral e derivativo. Além disto, são considerados controladores série os controladores *lead* (avanço de fase) e os controladores *lag* (atraso de fase). Os controladores por

realimentação são aqueles responsáveis pela realimentação de estados, como por exemplo, realimentação tacométrica [35].

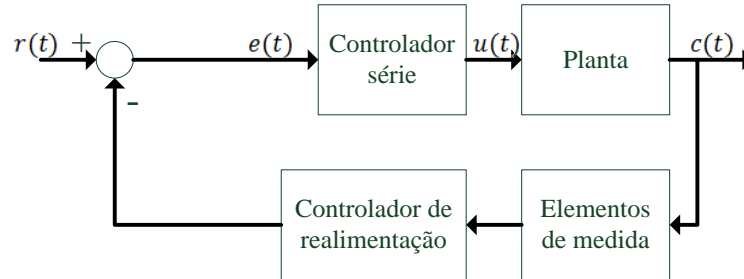


Figura II.2 – Diagrama de controladores série e por realimentação.

A seguir, será realizada uma breve análise sobre controladores proporcional, integral e derivativa, apresentando posteriormente um foco maior no controlador proporcional integral, visto que o mesmo é largamente utilizado na concepção de *Phase-Locked Loops* e sua dinâmica apresenta um fator preponderante em toda a teoria de PLLs.

#### b. Controladores proporcional, integral e derivativo

Nesta seção, serão abordados os detalhes das ações de controle básicas utilizadas na maior parte dos controladores industriais.

Controle proporcional: Para um controlador com ação de controle proporcional, a relação entre o sinal de saída do controlador  $u(t)$  e o sinal de erro atuante  $e(t)$  é dada por:

$$u(t) = K_p e(t) \quad (\text{II.1})$$

ou no domínio da transformada de *Laplace*,

$$\frac{U(s)}{E(s)} = K_p \quad (\text{II.2})$$

onde  $K_p$  é denominado ganho proporcional.

Basicamente, qualquer que seja o mecanismo real ou a forma de energia utilizada na operação, o controlador proporcional é essencialmente um amplificador com ganho ajustável [13].

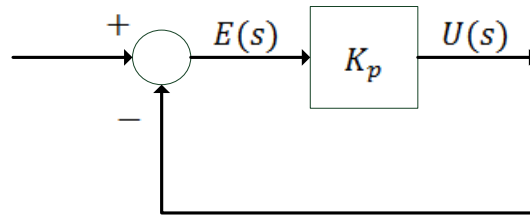


Figura II.3 - Diagrama de blocos de um controlador proporcional. Adaptada de [13].

Controle integral: Em um controlador com ação integral, o valor da saída do controlador  $u(t)$  é variado segundo uma taxa proporcional ao sinal de erro atuante  $e(t)$ , ou seja,

$$\frac{du(t)}{dt} = K_i e(t) \quad (\text{II.3})$$

ou

$$u(t) = K_i \int_0^t e(t) dt \quad (\text{II.4})$$

onde  $K_i$  é uma constante ajustável. A função de transferência do controlador integral é:

$$\frac{U(s)}{E(s)} = \frac{K_i}{s} \quad (\text{II.5})$$

Caso o valor de  $e(t)$  seja dobrado, então o valor de  $u(t)$  é variado 2 (duas) vezes mais rápido. Para erro atuante nulo, o valor de  $u(t)$  permanece estacionário. A Figura II.4 apresenta um diagrama de blocos deste controlador.

A razão principal da utilização de controladores integrais se dá devido ao fato de que o controle meramente proporcional de um processo cuja função de transferência não possui um integrador existe um erro de regime permanente, residual, na resposta a uma excitação em degrau. Este erro residual apenas pode ser eliminado caso seja incluída no controlador uma parcela de ação integral [13]. O efeito pode ser observado na Figura II.7.

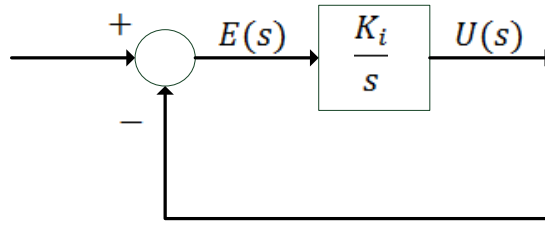


Figura II.4 – Diagrama de blocos de um controlador integral. Adaptada de [13].

Controle proporcional e integral: A ação de controle de um controlador proporcional e integral é dada por:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt \quad (II.6)$$

A função de transferência do controlador é:

$$\frac{U(s)}{E(s)} = K_p \left( 1 + \frac{1}{T_i s} \right) \quad (II.7)$$

onde  $K_p$  é o ganho proporcional e  $T_i$  é denominado o *tempo integral*. Tanto os parâmetros  $K_p$  quanto  $T_i$  são ajustáveis. O tempo integral tem a função de ajustar a ação de controle integral. Por outro lado, uma mudança no valor do ganho proporcional  $K_p$  afeta tanto a parcela proporcional como a parte integral da ação de controle. O inverso do tempo integral  $T_i$  é conhecido como *taxa de restabelecimento*, a qual é caracterizada pelo número de vezes por minuto que a parcela proporcional da ação de controle é duplicada. Em certas literaturas, é muito comum a utilização do parâmetro  $K_i$ , o qual é dado por:

$$K_i = K_p / T_i \quad (II.8)$$

A Figura II.5 apresenta um diagrama de blocos de um controlador proporcional integral. A saída deste controlador para a excitação de um degrau unitário pode ser vista através da Figura II.6.

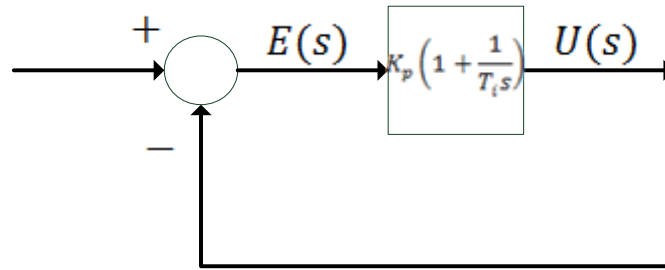


Figura II.5 – Diagrama de blocos de um controlador proporcional integral. Adaptado de [13].

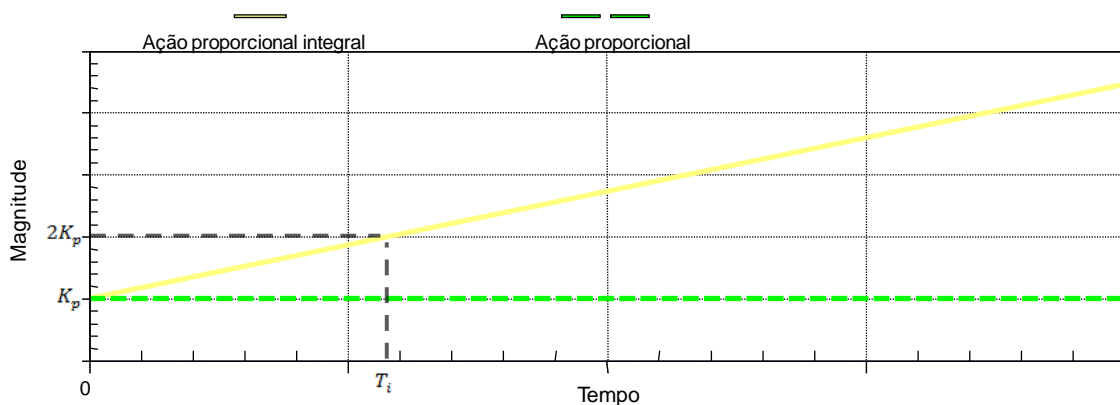


Figura II.6 – Diagramas esboçando a resposta ao degrau de um controlador proporcional (verde) e proporcional integral(amarelo).

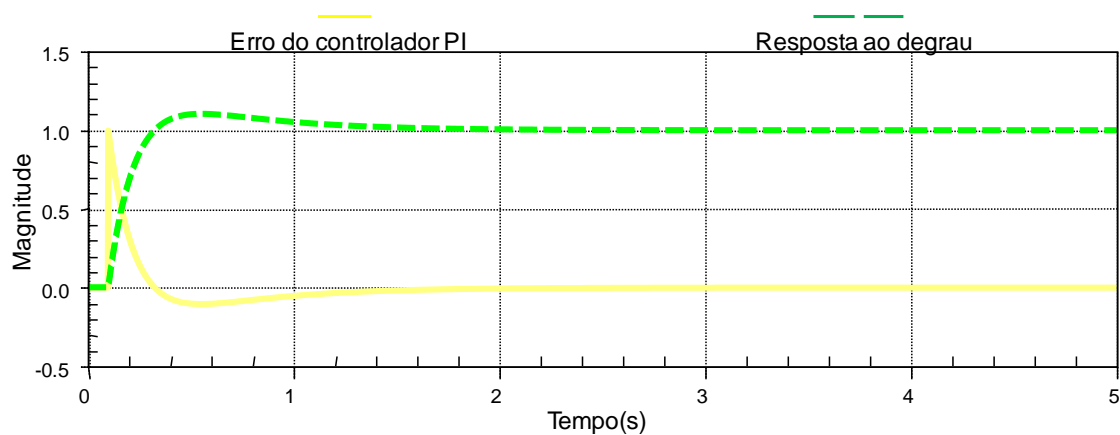


Figura II.7 – Resposta ao degrau unitário (verde) perante o erro atuante do controlador integral (amarelo).

Controle proporcional e derivativo: A ação de controle de um controlador proporcional e derivativo é dada pela seguinte equação:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt} \quad (\text{II.9})$$

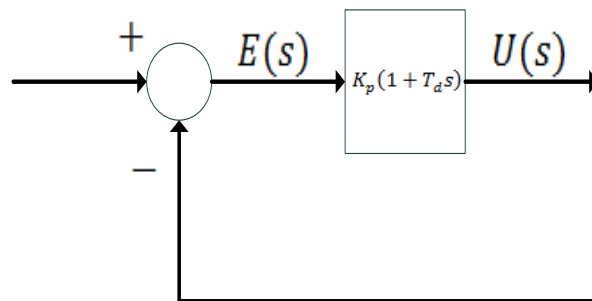


e sua função de transferência é dada por:

$$\frac{U(s)}{E(s)} = K_p(1 + T_d s) \quad (\text{II.10})$$

onde  $K_p$  novamente é dado como o ganho proporcional e  $T_d$  é uma constante chamada de tempo derivativo. Ambos os parâmetros são ajustáveis. A ação de controle derivativa presente neste controlador também é conhecida como controle de taxa. Neste caso, a magnitude de saída do controlador é proporcional a taxa de variação do sinal de erro atuante. O termo derivativo  $T_d$  é caracterizado pelo intervalo de tempo pelo qual a ação derivada avança o efeito da ação de controle proporcional [13]. A Figura II.8 mostra o diagrama de blocos do controlador proporcional e derivativo. Para uma entrada  $e(t)$  de rampa unitária, a saída do controlador proporcional e derivativo será a presente na Figura II.9.

Apesar da ação de controle derivativo possuir a vantagem de ser antecipatória, também apresenta a grave desvantagem de amplificação de sinais de ruídos. A ação derivativa nunca é utilizada sozinha, já que apenas é eficiente durante períodos transitórios.



*Figura II.8 – Diagrama de blocos de um controlador proporcional derivativo. Adaptado de [13].*

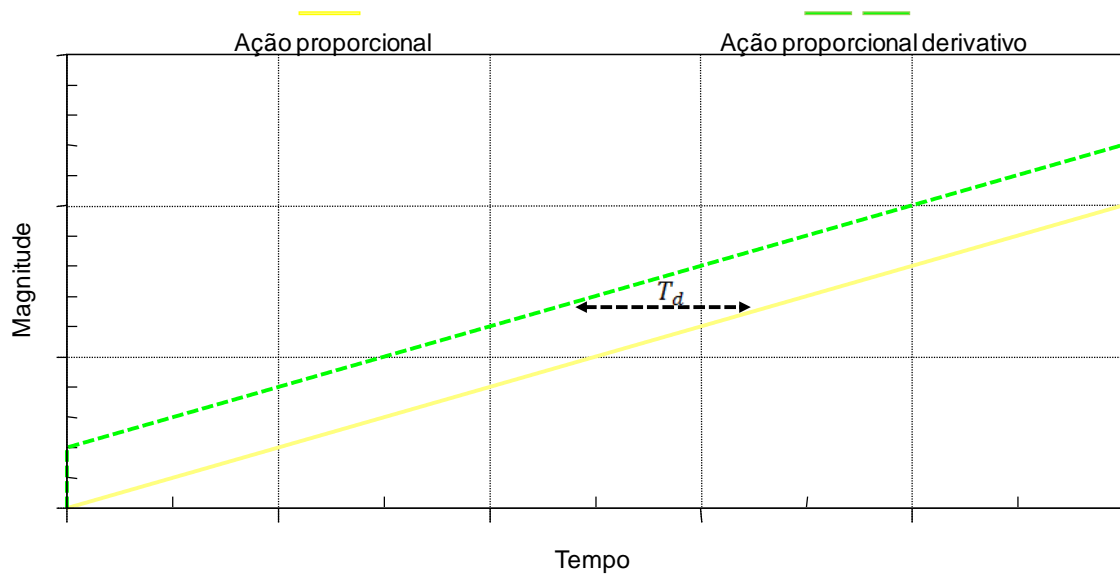


Figura II.9 - Diagramas esboçando a resposta a rampa unitária de um controlador proporcional (amarelo) e proporcional e derivativo (verde).

Controle proporcional-integral e derivativo: A combinação de todas as ações proporcional, integral e derivativa é chamada de proporcional-integral e derivativa. Esta ação de controle apresenta as vantagens de cada uma das ações de controle individualmente. A equação do controlador proporcional-integral e derivativo é dada por:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \quad (\text{II.11})$$

e sua função de transferência é dada por:

$$\frac{U(s)}{E(s)} = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (\text{II.12})$$

onde  $K_p$  e  $T_i$  são o ganho proporcional e o tempo integral, respectivamente e  $T_d$  é o tempo derivativo. O diagrama de blocos do controlador proporcional-integral e derivativo (também conhecido como PID) pode ser visto na Figura II.10.

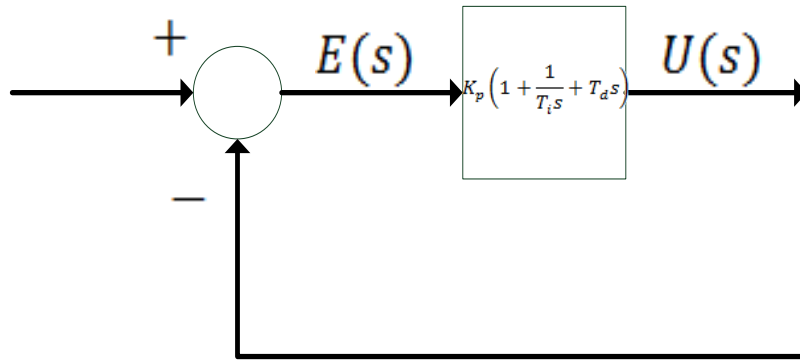


Figura II.10 - Diagrama de blocos de um controlador proporcional – integral e derivativo. Adaptado de [13].

**c. Análise da implementação do controlador proporcional integral**

Foi analisado anteriormente no Anexo 1 como é possível melhorar o amortecimento e o tempo de subida de um sistema de controle em detrimento de uma menor capacidade de rejeição a perturbações. A seguir, será analisado com mais atenção o comportamento de um controlador proporcional integral perante um sistema de segunda ordem.

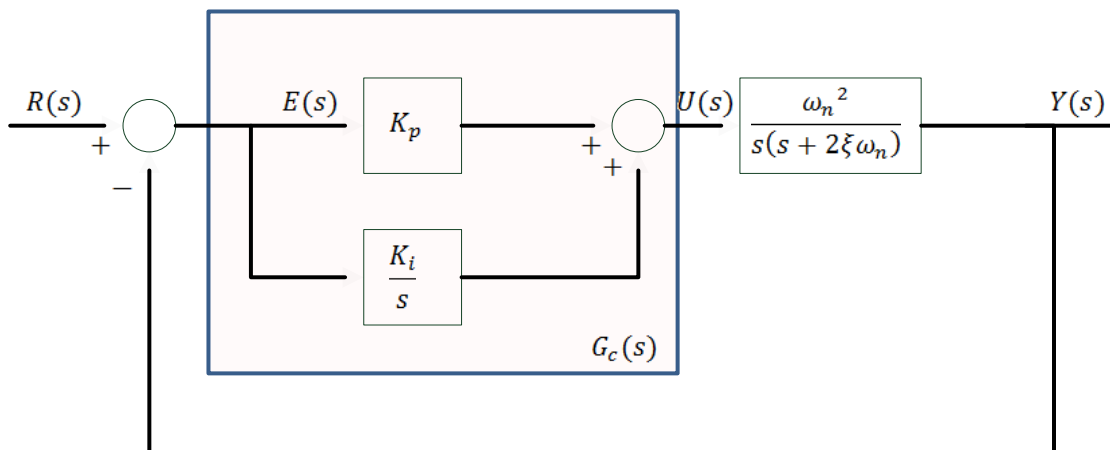


Figura II.11 - Sistema de controle com controlador PI.

A partir da Figura II.11, é possível reparar que como efeitos imediatos do controlador PI têm-se que:

- Adiciona um zero em  $s = -\frac{K_i}{K_p}$  a função de transferência de malha aberta.
- Adiciona um pólo em  $s = 0$  a função de transferência de malha aberta.

Desta forma, o sistema aumenta sua ordem em uma unidade. Assim, o erro de regime permanente do sistema original é aumentado em uma ordem. Ou seja, caso o erro de regime permanente para uma dada entrada é constante, o controlador PI o reduz a zero, visto que o sistema a ser compensado se mantenha estável.

Além disto, verifica-se que a função de transferência de malha aberta  $G(s)$  é dada por:

$$G(s) = G_c(s) \frac{\omega_n^2}{s(s + 2\xi\omega_n)} = \frac{\omega_n^2(K_p s + K_i)}{s^2(s + 2\xi\omega_n)} \quad (\text{II.13})$$

Ao trabalhar-se com um sistema de terceira ordem, o ganho  $K_p$  não mais afeta o erro de estado estacionário, sendo este sempre nulo para um sistema estável com uma entrada em forma de rampa. O problema agora se trata de escolher uma combinação apropriada entre  $K_p$  e  $K_i$  para que a resposta transiente seja satisfatória [15].

### III. Anexo 3

#### a. Forward Differences (Aproximação de Euler)

Na aproximação *Forward Differences*, a derivada de uma função  $y(kT_s)$  é dada por:

$$\dot{y}(kT_s) = \frac{y(k+1) - y(k)}{T_s} \quad (\text{III.1})$$

Considerando o sistema de primeira ordem

$$\dot{y}(t) + ay(t) = au(t) \quad (\text{III.2})$$

tem-se que:

$$sY(s) + aY(s) = aU(s) \quad (\text{III.3})$$

e, portanto:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{a}{s + a} \quad (\text{III.4})$$

A aproximação discreta de (III.2) utilizando (III.1), é dada, portanto, como:

$$\frac{y(k+1) - y(k)}{T_s} + ay(k) = au(k) \quad (\text{III.5})$$

Obtêm-se a partir de (III.5) que:

$$zY(z) - Y(z) + aT_s Y(z) = aT_s U(z) \quad (\text{III.6})$$

e portanto:

$$G_f(z) = \frac{Y(z)}{U(z)} = \frac{aT_s}{z - 1 + aT_s} \quad (\text{III.7})$$
$$G_f(z) = \frac{a}{\frac{z-1}{T_s} + a}$$

Através da comparação de (III.7) com (III.4), é possível verificar-se que, para obter  $G(z)$  a partir de  $G(s)$ , basta realizar a substituição de variáveis:

$$s = \frac{z - 1}{T_s} \quad (\text{III.8})$$

**b. Backward Differences (Aproximação de Euler)**

Na aproximação *Backward Differences*, a derivada de uma função  $y(kT_s)$  é dada por:

$$\dot{y}(kT_s) = \frac{y(k) - y(k - 1)}{T_s} \quad (\text{III.9})$$

A aproximação discreta de (III.2) utilizando-se de (III.9) é dada por:

$$\frac{y(k) - y(k - 1)}{T_s} + ay(k) = au(k) \quad (\text{III.10})$$

de onde tem-se :

$$Y(z) - z^{-1}Y(z) + aT_sY(z) = aT_sU(z) \quad (\text{III.11})$$

e portanto:

$$G_b(z) = \frac{Y(z)}{U(z)} = \frac{aT_s}{1 - z^{-1} - aT_s}$$

$$G_b(z) = \frac{a}{\frac{z - 1}{T_s z} + a} \quad (\text{III.12})$$

Através da comparação de (III.12) com (III.4), é possível verificar-se que, para obter  $G(z)$  a partir de  $G(s)$ , basta realizar a substituição de variáveis:

$$s = \frac{z - 1}{T_s z} \quad (\text{III.13})$$

**c. Transformação bilinear (Aproximação de Tustin)**

A aproximação de *Tustin* é dada por:

$$\frac{\dot{y}(k) + \dot{y}(k-1)}{2} = \frac{y(k) - y(k-1)}{T_s} \quad (\text{III.14})$$

A versão amostrada de (III.2) é:

$$\dot{y}(k) = au(k) - ay(k) \quad (\text{III.15})$$

ou atrasado no tempo:

$$\dot{y}(k-1) = au(k-1) - ay(k-1) \quad (\text{III.16})$$

Substituindo-se (III.15) e (III.16) em (III.14) e aplicando-se a transformada Z, obtém-se que:

$$Y(z) \left( 1 - z^{-1} + \frac{aT_s}{2} + \frac{aT_s}{2} z^{-1} \right) = U(z) (1 + z^{-1}) \frac{aT_s}{2} \quad (\text{III.17})$$

e portanto:

$$G_t(z) = \frac{Y(z)}{U(z)} = \frac{\frac{aT_s}{2} (z+1)}{\left( 1 + \frac{aT_s}{2} \right) z + \frac{aT_s}{2} - 1}$$

$$G_t(z) = \frac{a}{\frac{2}{T_s} \frac{(z-1)}{(z+1)} + a} \quad (\text{III.18})$$

Através da comparação de (III.18) com (III.4), é possível verificar-se que, para obter G(z) a partir de G(s), basta realizar a substituição de variáveis:

$$s = \frac{2}{T_s} \frac{z-1}{z+1} \quad (\text{III.19})$$

Tabela III.1 - Tabela de aproximações por integração numérica.

Aproximação	$s \rightarrow z$	$z \rightarrow s$
<i>Forward Differences</i>	$s = \frac{z - 1}{T_s}$	$z = 1 + T_s s$
<i>Backward Differences</i>	$s = \frac{z - 1}{T_s z}$	$z = \frac{1}{1 - T_s s}$
Transformação Bilinear	$s = \frac{2z - 1}{T_s z + 1}$	$z = \frac{1 + \frac{T_s s}{2}}{1 - \frac{T_s s}{2}}$

**d. Mapeamento com Forward Differences**

Utilizando-se da aproximação *Forward Differences*, tem-se que  $z = 1 + T_s s$ . Portanto, o limite de estabilidade  $s = j\omega$  mapeia-se em  $z = 1 + jT_s \omega$ . Esta região é mostrada na Figura III.1. Desta forma, é possível verificar que sistemas estáveis no plano  $s$  podem ser tornar instáveis quando mapeados para  $z$  por este método.

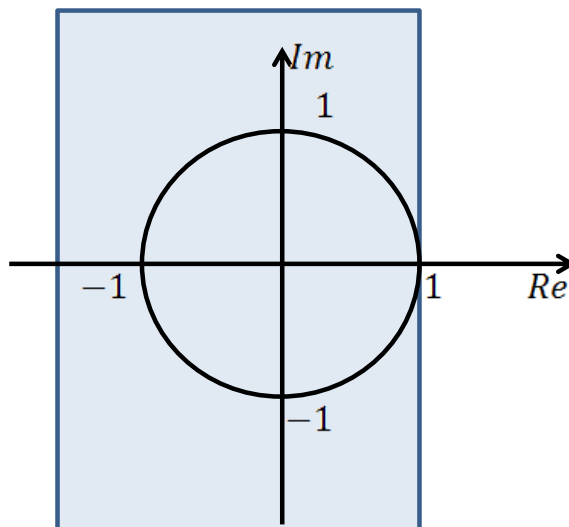


Figura III.1 - Mapeamento da região de estabilidade do plano  $s$  no plano  $z$  utilizando *Forward Differences*. Adaptado de [25].

**e. Mapeamento com Backward Differences**

Utilizando-se da aproximação *Backward Differences*, tem-se que:



$$z = \frac{1}{1 - T_s s} + \frac{1}{2} - \frac{1}{2} = \frac{1}{2} + \frac{1 T_s s + 1}{2(1 - T_s s)} \quad (\text{III.20})$$

onde:

$$\left| \frac{T_s s + 1}{1 - T_s s} \right|_{s=j\omega} = 1 \quad (\text{III.21})$$

Logo,  $\left| z - \frac{1}{2} \right| = \frac{1}{2}$ , o que representa um círculo de raio  $\frac{1}{2}$  centrado em  $\frac{1}{2}$ , como pode ser visto na Figura III.2. Esta transformação mapeia sistemas estáveis em  $s$  em sistemas estáveis em  $z$ . Por outro lado, nem toda a região de estabilidade no plano  $z$  é utilizada no mapeamento. Ou seja, esta característica faz com que a utilização desta transformação restrinja o espaço de projeto, já que especificações de desempenho não serão mapeadas de modo completamente coerente a partir desta aproximação.

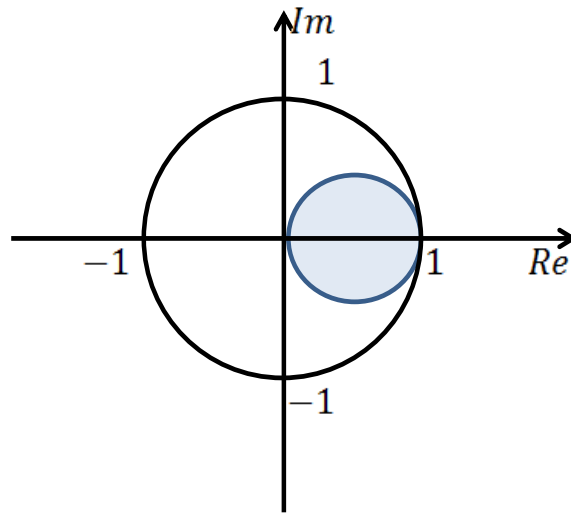


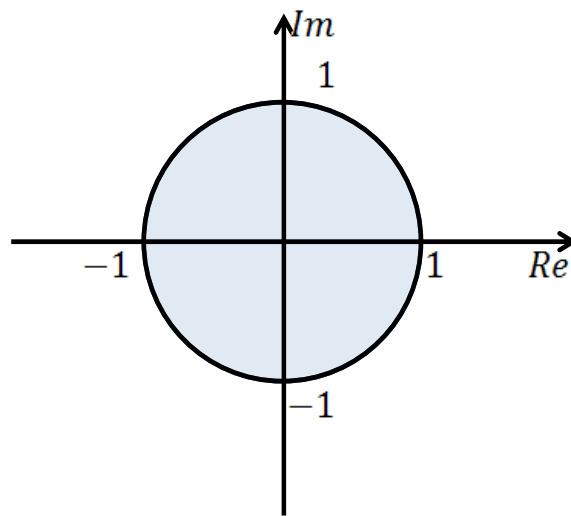
Figura III.2 - Mapeamento da região de estabilidade do plano  $s$  no plano  $z$  utilizando Backward Differences. Adaptado de [25].

#### f. Mapeamento com aproximação de Tustin

De modo análogo aos outros mapeamentos anteriores, tem-se que:

$$\left| \frac{1 + \frac{T_s s}{2}}{1 - \frac{T_s s}{2}} \right|_{s=j\omega} = 1 \quad (\text{III.22})$$

Logo,  $|z| = 1$ , o que representa o círculo unitário, como visto na Figura III.3. A aproximação de *Tustin* mapeia todo o semiplano esquerdo de  $s$  no interior do círculo unitário em  $z$ . Esta aproximação é a mais fidedigna de todas as citadas, apesar de possuir um custo computacional mais elevado que as demais. Atualmente, devido ao advento de microprocessadores cada vez mais rápidos, tornou-se a mais aconselhável para uso em algoritmos cujo mapeamento deve ser o mais próximo possível do domínio  $s$ . Por este motivo, todas as digitalizações presentes neste trabalho serão realizadas utilizando-se esta aproximação.



*Figura III.3 - Mapeamento da região de estabilidade do plano  $s$  no plano  $z$  utilizando aproximação de Tustin. Adaptado de [25].*

## IV. Anexo 4

### a. Script do MATLAB para cálculo dos parâmetros do *pPLL*

% Algoritmo para cálculo do  $w_p, w_z$  e  $k_i$ , além da plotagem do bode de  $G(s)$

```
kmax=sym('0.1');  
>> syms wp positive;  
>> syms wz;  
>> kp=sym('150');  
>> w2=sym('753.96');  
>> kv=sym('1.0');  
>> wz=(kp^2)/wp;  
>> Eq=(kmax^2)-((((kp*wp*kv*w2)^2)+((kp^3+kv)^2))/(((w2^2*wp)^2)+((w2^3)^2)));  
>> pretty(Eq);  
>> WP=solve(Eq,wp)  
>> Ki=(kp^3)*(WP^-1)  
WZ=(kp^2)/WP  
num=sym2poly([0 0 kp*WP*kv kp^2*kv]);  
>> den=sym2poly([1 WP 0 0]);  
>> system=tf(num,den);  
>> margin(system)  
>> grid
```

% Resposta ao degrau de  $G(s)$

```
>> num2=sym2poly([0 0 kv*kp kv*Ki]);  
>> den2=sym2poly([1/WP 1 kp*kv Ki*kv]);  
>> system2=tf(num2,den2);  
>> step(system2)  
>> grid
```

## b. Script do MATLAB para cálculo dos parâmetros do *parkPLL*

% Algoritmo para cálculo do  $w_p, w_z$  e  $k_i$ , além da plotagem do bode de  $G(s)$

```
kmax=sym('0.1');  
  
>> syms wp positive;  
  
>> syms wz;  
  
>> kp=sym('150');  
  
>> w2=sym('753.96');  
  
>> kv=sym('1.0');  
  
>> wz=(kp^2)/wp;  
  
>> Eq=(kmax^2)-((((kp*wp*kv*w2)^2)+((kp^3+kv)^2))/(((w2^2*wp)^2)+((w2^3)^2)));  
  
>> pretty(Eq);  
  
>> WP=solve(Eq,wp)  
  
>> TAU=1/(2*WP)  
  
>> Ki=(kp^3)*(WP^-1)  
  
WZ=(kp^2)/WP  
  
num=sym2poly([0 0 kp*WP*kv kp^2*kv]);  
  
>> den=sym2poly([1 WP 0 0]);  
  
>> system=tf(num,den);  
  
>> margin(system)  
  
>> grid  
  
% Resposta ao degrau de  $G(s)$   
  
>> num2=sym2poly([0 0 kv*kp kv*Ki]);  
  
>> den2=sym2poly([1/WP 1 kp*kv Ki*kv]);  
  
>> system2=tf(num2,den2);  
  
>> step(system2)  
  
>> grid
```