



Ministério da
**Ciência, Tecnologia
e Inovação**



sid.inpe.br/mtc-m19/2013/07.25.18.55-TDI

ESTUDO DOS EFEITOS DA RECONFIGURAÇÃO SOBRE O TRANSITÓRIO E A ESTABILIDADE DE SISTEMAS DE CONTROLE RECONFIGURÁVEIS

Jairo Cavalcanti Amaral

Tese de Doutorado do Curso de Pós-Graduação em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle, orientada pelo Dr. Marcelo Lopes de Oliveira e Souza, aprovada em 27 de maio de 2013.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/3EGT738>>

INPE
São José dos Campos
2013

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE (RE/DIR-204):

Presidente:

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Membros:

Dr. Antonio Fernando Bertachini de Almeida Prado - Coordenação Engenharia e Tecnologia Espacial (ETE)

Dr^a Inez Staciarini Batista - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

Dr. Germano de Souza Kienbaum - Centro de Tecnologias Especiais (CTE)

Dr. Manoel Alonso Gan - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr^a Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

EDITORAÇÃO ELETRÔNICA:

Maria Tereza Smith de Brito - Serviço de Informação e Documentação (SID)

Luciana Manacero - Serviço de Informação e Documentação (SID)



Ministério da
**Ciência, Tecnologia
e Inovação**



sid.inpe.br/mtc-m19/2013/07.25.18.55-TDI

ESTUDO DOS EFEITOS DA RECONFIGURAÇÃO SOBRE O TRANSITÓRIO E A ESTABILIDADE DE SISTEMAS DE CONTROLE RECONFIGURÁVEIS

Jairo Cavalcanti Amaral

Tese de Doutorado do Curso de Pós-Graduação em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle, orientada pelo Dr. Marcelo Lopes de Oliveira e Souza, aprovada em 27 de maio de 2013.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/3EGT738>>

INPE
São José dos Campos
2013

Dados Internacionais de Catalogação na Publicação (CIP)

Amaral, Jairo Cavalcanti.

Am14e Estudo dos efeitos da reconfiguração sobre o transitório e a estabilidade de sistemas de controle reconfiguráveis / Jairo Cavalcanti Amaral. – São José dos Campos : INPE, 2013.

xx + 172 p. ; (sid.inpe.br/mtc-m19/2013/07.25.18.55-TDI)

Tese (Doutorado em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2013.

Orientador : Dr. Marcelo Lopes de Oliveira e Souza.

1. controle reconfigurável. 2. cross-fading. 3. Plataforma Multimissão (PMM).4. critérios integrais I.Título.

CDU 629.7:681.51

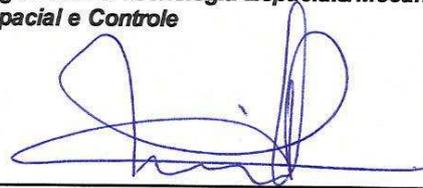


Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](https://creativecommons.org/licenses/by-nc/3.0/).

This work is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](https://creativecommons.org/licenses/by-nc/3.0/).

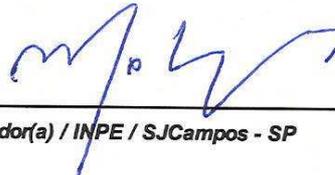
Aprovado (a) pela Banca Examinadora
em cumprimento ao requisito exigido para
obtenção do Título de **Doutor(a)** em
**Engenharia e Tecnologia Espaciais/Mecânica
Espacial e Controle**

Dr. Paulo Giacomio Milani



Presidente / INPE / SJC Campos - SP

Dr. Marcelo Lopes de Oliveira e Souza



Orientador(a) / INPE / SJC Campos - SP

Dr. Luiz Carlos Gadelha de Souza



Membro da Banca / INPE / SJC Campos - SP

Dr. Fernando José de Oliveira Moreira



Convidado(a) / EMBRAER / SJC Campos - SP

Dr. Henrique Mohallem Paiva



Convidado(a) / MECTRON / SJC Campos - SP

Este trabalho foi aprovado por:

maioria simples

unanimidade

Aluno (a): **Jairo Cavalcanti Amaral**

São José dos Campos, 27 de Maio de 2013

“A Natureza não dá saltos.”

Autor desconhecido

AGRADECIMENTOS

Agradeço a meus pais, por todo o apoio incondicional; aos membros da Banca Dr. Paulo Giácomo Milani, Dr. Luís Carlos Gadelha de Souza, Dr. Fernando José de Oliveira Moreira e Dr. Henrique Mohallem Paiva; a todos os meus professores; à Capes, pelo auxílio financeiro; e ao meu orientador Dr. Marcelo Lopes de Oliveira e Souza, pela inspiração e ensino.

Agradeço também aos amigos Guilherme Matter e Rachel Queiroz, pela ajuda vinda de longe; e aos amigos Alain Giacobini, Igor Mainenti, Jairo Moraes e Ximena Cubillos, pela ajuda vinda de perto.

Este trabalho não foi um desafio meramente intelectual.

RESUMO

Neste trabalho, foram utilizadas e abordagens teóricas de “cross-fading”, “anti-windup” e rastreamento de saída para reduzir a instabilidade durante a transição entre modos de controle; foram utilizados critérios integrais de uma maneira original, para determinar trajetórias de transição do sinal de controle durante o chaveamento entre modos de controle; e foi desenvolvido um novo método para descrever a Função-Transferência de um sistema durante o período de transição entre modos de controle usando “cross-fading”. Isso foi verificado pela utilização das ferramentas MatrixX/SystemBuild e MatLab para modelagem e simulação, enfatizando casos de aplicação prática, como um par de motores de corrente contínua e a PMM (Plataforma Multimissão).

Study of the effects of the reconfiguration on the transient and stability of reconfigurable control systems

ABSTRACT

In this work, we used theoretical approaches of cross-fading, anti-windup and output tracking to reduce the instability during the transition between control modes; we used integral criteria in an original way, to determine transition trajectories for the control signal during a switching between control modes; and we developed a new method to describe the Transfer Function of a system during the transition between control modes using cross-fading. This was verified using the MatrixX/Systembuild and MatLab tools to model and simulate, emphasizing cases of practical application, such as a pair of DC motors and the MMP (Multimission Platform).

LISTA DE FIGURAS

	<u>Pág.</u>
Figura 1-1: Resposta de torque mostrando os efeitos das transições do controlador. Fonte: Diallo et al. (2004).	3
Figura 1-2: Respostas de torque e velocidade mostrando os efeitos da transição do controle vetorial para o controle escalar em um motor de indução Fonte: Diallo et al. (2004).	4
Figura 2-1: Diagrama de um sistema chaveado. [Liberzon et Morse 1999].	9
Figura 2-2: Diagrama em blocos de um controlador PID com “anti-windup” do tipo “back calculation”	11
Figura 2-3: Resposta no tempo da velocidade angular durante três rotações consecutivas para aquisição da atitude nominal durante o Modo de Navegação Fina da PMM. Fonte: Amaral (2008).	12
Figura 2-4: Modelo de motor CC com PLL entre malhas.	13
Figura 2-5: ExoMars Rover Fonte: Vinci (2012).	14
Figura 2-6: Exemplo onde dois sistemas individualmente assintoticamente estáveis podem produzir um sistema composto instável Fonte: Liberzon et Morse (1999).	20
Figura 3-1: – Plataforma MultiMissão com painéis solares abertos e espaço destinado à carga útil. Fonte: INPE (2001b).	24
Figura 3-2: Lógica de Transição dos modos de Operação da PMM.	27
FONTE: INPE (2001a).	27
Figura 3-3: Diagrama de blocos dos submodos de uma única rotação.....	30
Figura 3-4: Modelo de motor CC com PLL entre malhas.	32
Figura 3-5: Diagrama de blocos de um motor CC genérico (Oliveira et Souza 2012).	33
Figura 3-6: Imagens do rover usado em obstáculos Fonte: Höpflinger et al. (2008).	34
Figura 3-7: Modos e submodos de controle Fonte: Höpflinger et al. (2008).	35
Figura 3-8: Diagrama em blocos mostrando o controlador C2 sendo obrigado a rastrear o sinal do controlador C1, antes do chaveamento.	36
Figura 3-9: Diagrama em blocos mostrando o controlador C2 introduzido na malha principal, após o chaveamento.	36
Figura 3-10: Diagrama de blocos de um sistema de controle durante a etapa de inicialização do controlador C2.....	37
Figura 3-11: Diagrama de blocos de um sistema de controle após a etapa de chaveamento para o controlador C2.	38
Figura 3-12: Diagrama dos sinais de controle primário (C1, contínuo) e secundário (C2, tracejado).	38

Figura 3-13: Diagrama dos sinais de controle primário (C1, azul) e secundário (C2, vermelho).....	39
Figura 3-14: Uma de várias possíveis implementações “anti-windup” para um controlador PID.	40
Figura 3-15: Diagrama de um sistema com “cross-fading” do sinal de controle.	41
Figura 3-16: Gráfico da função de multiplexação at genérica ao longo do tempo.	42
Figura 3-17: Diagrama de um sistema com “cross-fading” no sinal de erro.	42
Figura 4-1: Diagrama em blocos de um sistema genérico.	46
Figura 4-2: Diagrama em blocos de um sistema que chaveia entre dois controladores.....	47
Figura 4-3: Sinal de erro reduzido segundo um critério integral durante a etapa de transição.....	48
Figura 4-4: Diagrama em blocos do sistema durante a etapa de transição entre modos. A referência original deixa de ser usada, e uma nova referência do supervisor é introduzida.	48
Figura 4-5: Diagrama em blocos do controle usado imediatamente antes da etapa de transição, com o bloco supervisor medindo as saídas dos blocos C1 e C1.....	50
Figura 4-6: Diagrama em blocos do controle usado durante a etapa de transição, com a saída do controlador C1/2 sendo usada pela planta, mas guiada apenas pela referência do supervisor.....	50
Figura 4-7: Diagrama em blocos do controle usado após a etapa de transição, com o controlador C2 sendo usado como um bloco de controle comum.	51
Figura 4-8: Modelo de um sistema que pode realizar uma transição em “cross-fading” entre os controladores C1 e C2.....	61
Figura 4-9: Gráfico da função $a(t)$ para uma forma de “cross-fading” linear.....	62
Figura 4-10: Função $a(t)$ para o “cross-fading” linear ilimitado, onde não há a modelagem do fim da transição.	63
Figura 4-11: Função at para o “cross-fading” em decaimento exponencial.	66
Figura 5-1: Sinal de Controle com rastreo de saída.	72
Figura 5-2: Sinal de Controle sem rastreo de saída.	73
Figura 5-3: Velocidade Angular com rastreo de saída.....	73
Figura 5-4: Velocidade Angular sem rastreo de saída.....	74
Figura 5-5: Sinal de erro com rastreo de saída.	75
Figura 5-6: Sinal de erro sem rastreo de saída.	76
Figura 5-7: Módulo da velocidade angular (acima) e sinal de controle (abaixo) sem “cross-fading” nem “anti-windup”.	78
Figura 5-8: Módulo da velocidade angular (acima) e sinal de controle (abaixo) sem “cross-fading” nem “anti-windup”.	79
Figura 5-9: Módulo da velocidade angular (acima) e sinal de controle no eixo y (abaixo) sem “cross-fading” nem “anti-windup”.	80
Figura 5-10: Módulo da velocidade angular total (acima) e sinal de controle no eixo y (abaixo) com “anti-windup”.....	81

Figura 5-11: Módulo da velocidade angular (acima) e o sinal de controle (abaixo) com “cross-fading”	82
Figura 5-12: Módulo da velocidade angular (acima) e o sinal de controle (abaixo) com “cross-fading”	83
Figura 5-13: Diferença de posição entre os motores com 1 PLL mas sem “cross-fading”, em graus.....	85
Figura 5-14: Mostra a diferença de posição entre os motores com 1 PLL e 1 PID, mas sem “cross-fading”	86
Figura 5-15: Mostra a diferença de posição entre os motores (acima), o sinal de chaveamento original (meio), e o sinal de chaveamento suavizado (abaixo) com 1 PID e 1 PLL+1 PID, mas sem “cross-fading”	87
Figura 5-16: Sinal de controle ao longo de vários chaveamentos entre modos de controle sem nenhuma estratégia de atenuação de impacto.	89
Figura 5-17: Sinal de controle ao longo de vários chaveamentos entre modos de controle com controle PID + Integrador intermediário não ótimo.	90
Figura 5-18: Sinal de controle ao longo de vários chaveamentos entre modos de controle com controle PID + Integrador intermediário não ótimo, lento em comparação com o segundo controlador.	91
Figura 5-19: Funcional J (ISE) ao longo de ζ	92
Figura 5-20: Funcional J (ISE) ao longo de ζ	93
Figura 5-21: Funcional J (ITSE) ao longo de ζ	94
Figura 5-22: Funcional J (ITSE) ao longo de ζ	95
Figura 5-23: Funcional J (IME) ao longo de ζ	96
Figura 5-24: Funcional J (IME) ao longo de ζ	97
Figura 5-25: Funcional J (ITME) ao longo de ζ	98
Figura 5-26: Funcional J (ITME) ao longo de ζ	99
Figura 5-27: comparação entre as transições no modelo de 1 eixo da PMM para os quatro critérios integrais para $J = 10e-5$	101
Figura 5-28: comparação entre as transições no modelo de 1 eixo da PMM para os quatro critérios integrais para $J = 10e-5$	102
Figura 5-29: comparação entre as transições no modelo de 1 eixo da PMM para os quatro critérios integrais para $J = 10e-5$	103
Figura 5-34: Sinal de controle visto pela planta durante uma transição de modos (ISE).	109
Figura 5-35: Sinal de controle visto pela planta durante uma transição de modos (IME).	110
Figura 5-36: Sinal de controle visto pela planta durante uma transição de modos (ISE).	112
Figura 5-37 Sinal de controle visto pela planta durante uma transição de modos (ITSE).	113
Figura 5-38: Sinal de controle visto pela planta durante uma transição de modos (IME).	114
Figura 5-39: Sinal de controle visto pela planta durante uma transição de modos (ITME).	115
Figura 6-1: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 100$ s.	118

Figura 6-2: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 50$ s.	119
Figura 6-3: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 20$ s.	120
Figura 6-6: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 0,1$ s.	123
Figura 6-7: Polos do sistema da PMM durante “cross-fading” exponencial $\Delta t = 100$ s.	125
Figura 6-8: PMM exponencial $\Delta t = 50$ s.....	126
Figura 6-9: PMM exponencial $\Delta t = 10$ s.....	127
Figura 6-10: PMM exponencial $\Delta t = 5$ s.....	128
Figura 6-11: PMM exponencial $\Delta t = 1$ s.....	129
Figura 6-12: PMM exponencial $\Delta t = 0,1$ s.....	130
Figura 6-13: PMM exponencial $\Delta t = 0,1$ s, mostrando grupos de polos mais à esquerda, associados ao impacto da transição.....	131
Figura 6-14: PMM exponencial $\Delta t = 0,1$ s, detalhe no grupos de polos mais à esquerda, associados ao impacto da transição.....	132

LISTA DE SIGLAS E ABREVIATURAS

INPE	Instituto Nacional de Pesquisas Espaciais
PMM	Plataforma Multimissão
CC	Corrente Contínua
PLL	Phase Locked Loop
ISE	Integral of Squared Error
IAE	Integral of Absolute Error
ITAE	Integral of Time and Absolute Error
PID	Proporcional Integral Derivativo
VLHL	Vertical Local Horizonte Local

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
1.1. Contexto	1
1.2. Motivação	2
1.3. Objetivo.....	4
1.4. Originalidade, Generalidade e Utilidade.....	5
1.5. Organização deste Trabalho.....	6
2 CONCEITOS BÁSICOS E REVISÃO BIBLIOGRÁFICA	7
2.1. Sistemas de Controle Reconfiguráveis, Chaveados, Adaptativos, Clássicos, Modernos.....	7
2.2. Definição de um sistema chaveado:.....	8
2.3. Efeitos do Transitório Causado pela Reconfiguração.....	9
2.4. Abordagem para Tratamento do Transitório Durante a Reconfiguração.....	14
2.5. Sensibilidade de Trajetórias e Rastreamento de Trajetórias.....	16
2.6. Convergência de Algoritmos de Adaptação.....	16
2.7. Problemas em aberto sobre o gerenciamento de transitórios.....	16
2.8. Plataforma de Controle Aberta (GeorgiaTech).....	18
2.9. Efeitos da Reconfiguração sobre a Estabilidade.....	19
2.10. Ferramentas de Simulação.....	21
2.11. Conexão da Literatura com este Trabalho.....	21
3 FORMULAÇÃO DO PROBLEMA.....	23
3.1. Sistemas em Estudo.....	23
3.1.1. A Plataforma Multimissão.....	24
3.1.2. Par de Motores de Corrente Contínua.....	32
3.1.3. Rovers como o ExoMars Rover.....	33
3.2. Abordagens para sua Solução.....	35
3.2.1. Rastreio de Saída.....	35
3.2.2. “Anti-windup”.....	39
3.2.3. “Cross-Fading”.....	41
3.2.4. Critérios Integrais.....	43
4 ESTUDO E REDUÇÃO DOS EFEITOS DA RECONFIGURAÇÃO SOBRE OS TRANSITÓRIOS.....	45
4.1. Exemplos.....	45
4.2. Aplicação.....	47
4.2.1. Exclusão do Atuador, Planta e Sensor.....	49

4.2.2. Critério Integral do Quadrado do Erro (ISE)	51
4.2.3. Critério Integral do Tempo Vezes o Quadrado do Erro (ITSE).....	56
4.2.4. Critérios Integral do Módulo do Erro (IME)	58
4.2.5. Critérios Integral do Tempo Vezes o Módulo do Erro (ITME).....	60
4.2.6. Estabilidade Geral Durante Transições.....	61
5 SIMULAÇÕES E ANÁLISES DA REDUÇÃO DOS EFEITOS DA RECONFIGURAÇÃO SOBRE OS TRANSITÓRIOS.....	71
5.1. Uso de rastreo de saída para a transição de submodos dentro do modo de navegação fina da PMM.....	71
5.2. Combinações de “cross-fading” e “anti-windup” no Modo de Navegação Fina..	77
5.3. Par de Motores de Corrente Contínua.....	84
5.4. Critérios integrais	88
5.4.1. Critério Integral do Quadrado do Erro (ISE).	92
5.4.2. Critério Integral do Tempo vezes o Quadrado do Erro (ITSE)	94
5.4.3. Critério Integral do Módulo do Erro (IME)	96
5.4.4. Critério Integral do Tempo vezes o Módulo do Erro (ITME)	98
5.4.5. Comparação dos Critérios no Modelo da PMM	100
5.4.6. Comparação dos Critérios no Modelo dos Motores de Indução.	111
6 ESTUDO DE ESTABILIDADE SOBRE A RECONFIGURAÇÃO NO MODELO DA PMM.....	117
6.1. “Cross-fading” Linear Ilimitado.....	117
6.2. “Cross-fading em Decaimento Exponencial	124
7 CONCLUSÕES E SUGESTÕES.....	133
7.1 CONCLUSÕES	133
7.2 SUGESTÕES PARA TRABALHOS FUTUROS.....	135
REFERÊNCIAS BIBLIOGRÁFICAS	137
APÊNDICE A: CÓDIGOS EM MATLAB PARA CALCULAR E PLOTAR OS GRÁFICOS DOS CRITÉRIOS INTEGRAIS.....	143
APÊNDICE B: CÓDIGOS EM MATLAB PARA CALCULAR E PLOTAR OS GRÁFICOS DA ESTABILIDADE GERAL.....	147
APÊNDICE C: CÓDIGO DO MATRIXX/SYSTEMBUILD PARA O BLOCO DE SIMULAÇÃO DE TRANSIÇÃO USANDO CRITÉRIOS INTEGRAIS.....	148

1 INTRODUÇÃO

1.1. Contexto

Muitos sistemas de engenharia encontrados na prática exibem diversos modos de operação que são dependentes de mudanças na planta ou no ambiente. Sua correta operação usualmente exige que sejam controlados por sistemas de controle baseadas em chaveamento entre diversos modos de operação ou leis de controle. Ao longo do tempo, diversas abordagens foram aplicadas sucessiva e extensivamente, em particular:

- 1) O uso de um mesmo modo ou lei de controle para controlar os vários modos de operação da planta, escolhido segundo algum critério fixo (i.e., **não considera a mudança durante a operação**; e.g., desempenho médio, etc.) **(Controle Clássico e Controle Moderno) (Ogata, 1985)**. Sua vantagem é a simplicidade, mas sua desvantagem é o desempenho apenas aceitável ou até satisfatório.
- 2) O uso de um mesmo modo ou lei de controle para controlar os vários modos de operação da planta, escolhido segundo algum critério adaptativo (i.e., **considera a mudança durante a operação**; e.g., desempenho instantâneo, identificação da planta, etc.) **(Controle Adaptativo) (Åström, 2006)**. Sua vantagem é melhorar o desempenho na presença das mudanças. A importância desses métodos de controle também vem em parte da existência de sistemas que não podem ser estabilizados assintoticamente por uma única lei de controle contínua com realimentação **(Liberzon et Morse, 1999)**. Sua desvantagem é a escolha e aplicação do critério adaptativo, e a prova das suas propriedades no transitório e na estabilidade.
- 3) O uso de um modo ou lei de controle para controlar cada um os modos de operação da planta, escolhidas segundo algum critério (desempenho local máximo, etc.) e chaveados segundo alguma lógica de chaveamento (i.e., a

mudança é descontínua e entre um número inteiro positivo de opções de valores ou topologias ≥ 2 ;) (**Controle Chaveável**) (**Liberzon et Morse 1999**). Sua vantagem é a possibilidade de segmentar sistemas complexos tornando mais viável a linearização de operação de cada subsistema, uma vez que cada um deles é usado numa região de operação. Isso permite maior flexibilidade na escolha das diversas teorias, mais bem desenvolvidas, para sistemas lineares, com as quais se ocupa boa parte da literatura. Sua desvantagem é o repetido projeto para cada modo de controle e os problemas com transitórios e com a estabilidade causados pelo chaveamento.

- 4) O uso de um modo ou lei de controle para controlar cada um os modos de operação da planta, escolhidas segundo algum critério (desempenho local máximo, etc.) e reconfigurados segundo alguma lógica de reconfiguração (i.e., é **mudança contínua ou descontínua** e entre um número real positivo ou inteiro positivo de opções de valores ou topologias ≥ 2); (**Controle Reconfigurável**) (**Guler et al. 2003**). Sua vantagem é a possibilidade de modificar sistemas continuamente. Sua desvantagem é a dificuldade ainda maior no projeto de uma quantidade incontável modos de controles, e os problemas com transitórios e com a estabilidade causados pela reconfiguração.

1.2. Motivação

- 1) Mais recentemente, a área de sistemas de controle reconfiguráveis está se desenvolvendo rapidamente, e isso tem numerosas aplicações nas indústrias aeroespacial, automotiva, de processos, no controle de aeronaves e do tráfego aéreo, e várias outras áreas (Todman et al. 2005).
- 2) Embora sistemas de controle reconfiguráveis proporcionem vantagens no projeto e desempenho de sistemas de engenharia e sejam cada vez mais utilizados, a reconfiguração introduz transitório, no mínimo

temporariamente, que pode comprometer o desempenho e até introduzir instabilidades. O problema é conhecido e já foi abordado de diversas formas, mas está longe de ser esgotado. (Simon et Al. 2002) Por isso, o estudo dos transitórios e da estabilidade é importante para assegurar as vantagens desse tipo de sistema.

- 3) Como exemplo de caso, em (Diallo et al. 2004), uma estratégia de controle tolerante a falhas foi simulada num motor de indução, onde uma sequência de falhas de sensores foi implementada, e as transições para modos de falha introduziram sobre-elevações e oscilações duradouras. As Figuras 1.1 e 1.2 mostram respostas durante transições de modos de controle.

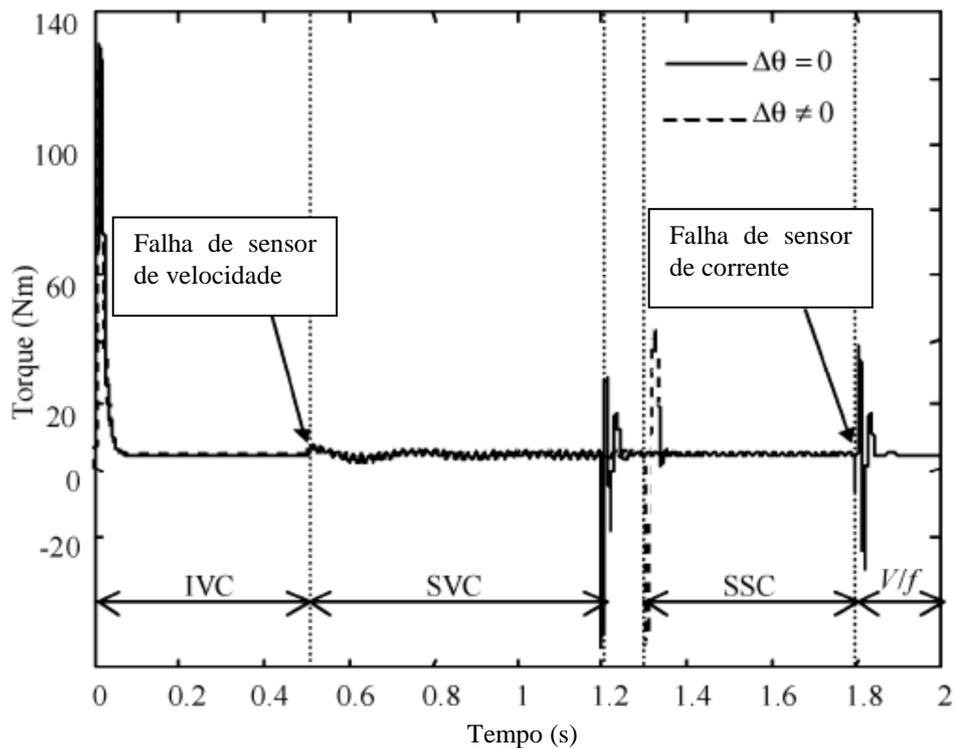


Figura 1-1: Resposta de torque mostrando os efeitos das transições do controlador.
Fonte: Diallo et al. (2004).

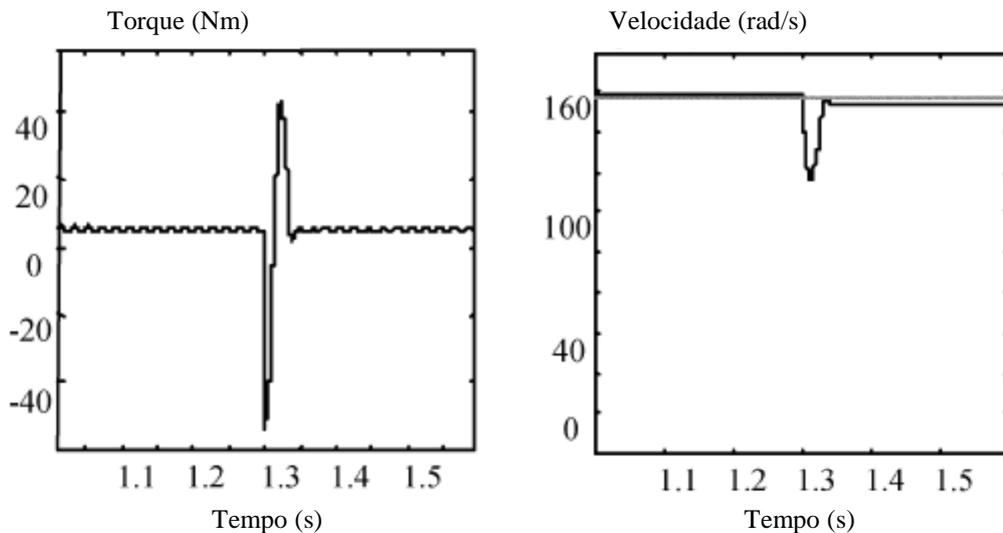


Figura 1-2: Respostas de torque e velocidade mostrando os efeitos da transição do controle vetorial para o controle escalar em um motor de indução
 Fonte: Diallo et al. (2004).

1.3. Objetivo

O objetivo desta Tese é o estudo dos efeitos da reconfiguração sobre o transitório e a estabilidade de sistemas de controle reconfiguráveis. Para isto foram adotados os seguintes passos:

- 1) Rever a literatura sobre os efeitos da reconfiguração sobre o transitório e a estabilidade de sistemas de controle reconfiguráveis, e sobre sua redução;
- 2) Investigar e buscar melhorias em técnicas de redução dos efeitos da reconfiguração sobre o transitório de sistemas de controle reconfiguráveis que usem uma das seguintes abordagens existentes na literatura:
 - a. Reconfiguração de passos múltiplos (série de reconfigurações de passo único);
 - i. Métodos de “cross-fading” de saídas;
 - ii. Métodos de “cross-fading” de entradas;
 - b. Métodos de rastreamento das saídas (“output tracking”): ou seja, pelo rastreamento de uma saída do controlador principal por uma

saída do controlador secundário envolvidos na reconfiguração, baseada no espelhamento entre elas;

- c. Minimização do erro entre saídas (“Output error minimization”): pela minimização de uma função do erro entre uma saída do controlador principal (1) e uma saída do controlador secundário (2) envolvidos na reconfiguração, baseada numa extensão do critério ISE (“Integral of the Squared Error”);

1.4. Originalidade, Generalidade e Utilidade

É esperado que todo trabalho de doutorado cumpra essas qualidades com rigor. Referente ao nosso trabalho, pode se dizer que:

- 1) A utilidade vem do fato de que isso permite que tal tipo de sistema de controle pode ser projetado com mais rapidez e/ou complexidade, sem prejuízo de sua eficácia. A originalidade vem do fato de encontrar das abordagens analíticas para atribuir um valor qualitativo a uma transição entre dois modos de controle e modelar a função transferência durante um “cross-fading” entre dois modos de controle, para verificar e garantir o funcionamento de sistemas que reconfigurem entre várias leis de controle.
- 2) A generalidade vem do fato de que esse tipo de sistema de controle se faz cada vez mais frequente em diversos sistemas de engenharia, em vez de se limitar a casos extremos. (esperava a aplicação em vários casos e mostrar uso em casos gerais)
- 3) A utilidade vem do fato de que isso permite que tal tipo de sistema de controle pode ser projetado com mais rapidez e/ou complexidade, sem prejuízo de sua eficácia. Seja específico: aplicação a PMM.

1.5. Organização deste Trabalho

Os capítulos restantes desta Tese estão organizados da seguinte maneira:

- o Capítulo 2 apresenta os Conceitos Básicos e a Revisão Bibliográfica, o que inclui a definição de sistemas chaveados, e exemplos de tecnologias que utilizam formas de reconfiguração ou estratégias para melhorar a estabilidade durante transições entre modos de controle;
- o Capítulo 3 apresenta a Formulação do Problema e as Abordagens para sua Solução, o que inclui a descrição dos modelos de sistemas utilizados nesta Tese, a Plataforma Multimissão e o par de motores CC, e as estratégias “anti-windup”, “cross-fading” e rastreamento de saída;
- o Capítulo 4 trata do Estudo e Redução dos Efeitos da Reconfiguração sobre os Transitórios, onde serão apresentadas quatro formas originais de determinar uma forma de transição entre dois modos de controle, utilizando quatro critérios integrais, e também duas formas originais de analisar a Função Transferência de um sistema durante uma transição em “cross-fading”;
- o Capítulo 5 apresenta as Simulações e Análises da Redução dos Efeitos da Reconfiguração Sobre os Transitórios, onde são mostrados os resultados dos métodos “anti-windup”, “cross-fading”, rastreamento de saída e a transição por critério integral, nos modelos da Plataforma Multimissão e/ou dos motores CC;
- o Capítulo 6 apresenta um Estudo de Estabilidade Sobre a Reconfiguração e mostra como a transição em “cross-fading” altera a quantidade e posição de polos e zeros no plano complexo, segundo os métodos exponencial e linear;
- e o Capítulo 7 apresenta as conclusões finais sobre os resultados obtidos e oferece sugestões para trabalhos futuros.

2 CONCEITOS BÁSICOS E REVISÃO BIBLIOGRÁFICA

2.1. Sistemas de Controle Reconfiguráveis, Chaveados, Adaptativos, Clássicos, Modernos

Um sistema de controle reconfigurável é aquele capaz de modificar qualitativamente e quantitativamente sua lei de controle, comandado por um controle supervisor. Embora a literatura encontrada enfatize a reconfigurabilidade devida a falhas, ela também pode se dever a mudanças no ambiente ou a etapas no funcionamento da planta.

Exemplos disso podem ser vistos em alguns usos dos “Field Programmable Gate Arrays” (FPGAs) (Papakonstantinou et al. 2011), que possuem um grande paralelismo em seus recursos lógicos, e alcançam um grande “throughput” computacional mesmo em baixas frequências de relógio. A flexibilidade dos FPGAs permite um desempenho ainda mais alto trocando a precisão e alcance do formato de número por uma maior quantidade de unidades aritméticas. Isso levou a um tipo de processamento chamado **Computação Reconfigurável** (Compton et Hauck 2002), onde tarefas que exigiriam muito tempo são executadas em FPGAs.

Num caso mais simples, porém frequente, a modificação da lei de controle pode ser simplesmente feita através do chaveamento entre blocos com leis de controle diferentes (Liberzon & Morse 1999). Este caso é chamado de **Controle Chaveado**.

Num outro caso ainda mais simples, a modificação da lei de controle pode ser simplesmente feita através de mudanças quantitativas nos parâmetros de controle de uma única lei de controle (Åström 2006). Este caso é chamado de **Controle Adaptativo**.

No caso mais simples de todos, a lei de controle pode ser escolhida durante o projeto, mas após isto, é mantida. (Ogata 1993). Este caso é chamado de **Controle Clássico ou Moderno**.

Como o caso de controle chaveado é o mais simples e o mais frequente na literatura dentre os controles de interesse desta tese, começaremos por ele a seguir.

2.2. Definição de um sistema chaveado:

De acordo com (Liberzon & Morse 1999), um **sistema chaveado** (Figura 2-1) pode ser matematicamente descrito por uma equação diferencial da forma:

$$\dot{x}(t) = f_{\sigma}(x(t)) \quad (2.1)$$

onde $\{f_p: p \in P\}$ é uma família de funções suficientemente regulares de \mathbb{R}^n para \mathbb{R}^n , parametrizada por um parâmetro $p \in P =$ conjunto índice, e $\sigma: [0, \infty) \times \mathbb{R}^n \rightarrow P$ tal que $p = \sigma(t, x(t))$ é uma função discreta no tempo, chamada de **sinal chaveador**. Complementando tal definição, esclarecemos que $P \subset \{2, 3, 4, \dots\}$ e representa os blocos de controle que o supervisor pode usar na malha do sistema.

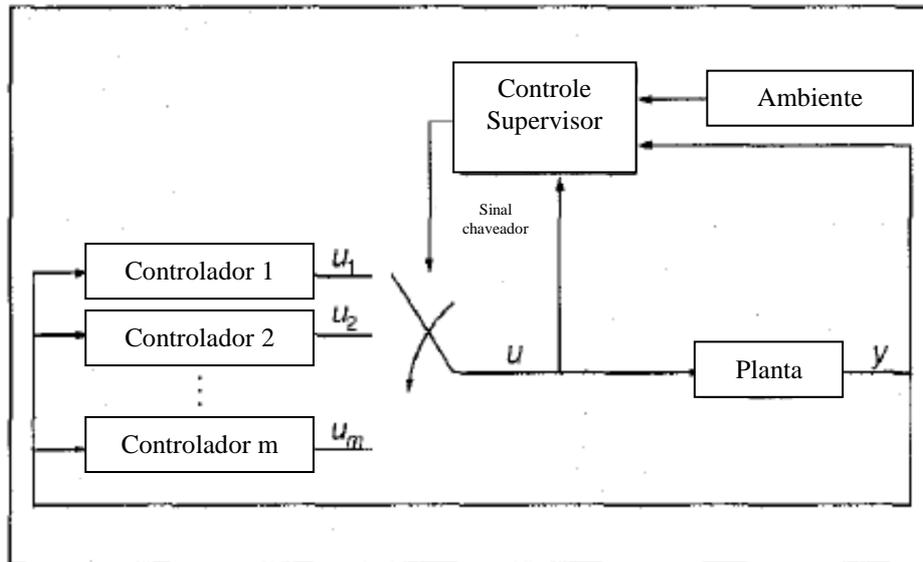


Figura 2-1: Diagrama de um sistema chaveado. [Liberzon et Morse 1999].

Em situações específicas, o valor de $p = \sigma$ pode depender de t , de $x(t)$, de ambos, e pode ser gerado com técnicas mais sofisticadas. Assumimos que o estado $x(t)$ da Equação (2.1) não tem descontinuidades durante os instantes de chaveamento, isto é, a solução $x(\cdot)$ é contínua em todos os locais. A situação onde o chaveamento é muito rápido (“chattering”) **não é abordada nesta definição**. No caso particular onde todos os subsistemas são lineares, temos um **sistema linear chaveado**:

$$\dot{x}(t) = A_p x(t), \quad (2.2)$$

que é uma das formas mais abordadas na literatura [Liberzon et Morse 1999].

2.3. Efeitos do Transitório Causado pela Reconfiguração

Ainda que a estabilidade geral do sistema possa ser garantida, isso por si só não leva em conta o transitório que ocorre na mudança entre modos. A irregularidade no transitório (“**impacto**”) é referida na literatura como “**bump**” (Simon et al. 2002), e pode introduzir sobrelevações e instabilidades

temporárias, podendo degradar a performance do sistema, ou até mesmo danificar a planta.

De acordo com (Görçek et al. 2002), a questão da **transferência “bumpless”** tem sido considerada usualmente como semelhante ao problema de **“anti-windup”**, que atraiu muita atividade de pesquisa por vários anos. O fenômeno de **“windup”** é bem conhecido, e é causado pelo acúmulo na saída de um componente integrador, durante uma abertura da malha de controle, como no caso da saturação de um atuador (Packard 2005).

Há várias estratégias para eliminar seu efeito, e uma delas é simplesmente recalcular o termo integral de forma que seu valor permaneça não sature o atuador. É vantajoso fazer essa correção dinamicamente com uma constante de tempo T_t (Åström 2006).

A Figura 2-2 mostra o diagrama em blocos de um controlador PID com “anti-windup” do tipo “back calculation”, um de diversos tipos de estratégia “anti-windup”. O sistema apresenta um laço de realimentação adicional. A diferença entre os valores da entrada e da saída do atuador constitui um erro e_s que é realimentado na entrada do integrador com um ganho $1/T_t$. Note que quando não há saturação o erro e_s é igual a zero e, portanto, este laço não tem nenhum efeito quando o controlador está operando linearmente, ou seja, quando sua saída não está saturada.

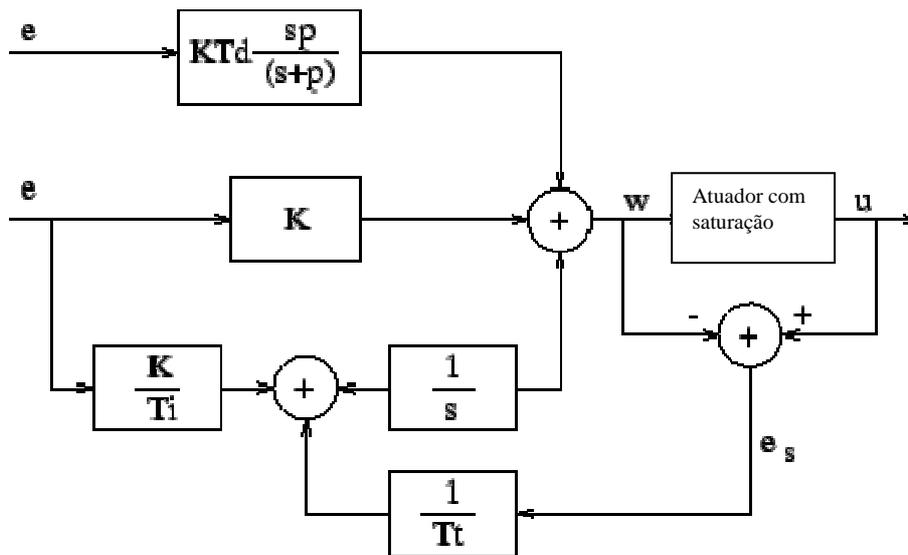


Figura 2-2: Diagrama em blocos de um controlador PID com “anti-windup” do tipo “back calculation”.

No caso do fenômeno “bump”, a irregularidade é diretamente relacionada à diferença entre o valor de saída do controlador ativo antes do chaveamento e o controlador ativo após chaveamento e seu efeito na realimentação.

Como um exemplo do problema de impacto, a Figura 2-3 mostra a resposta no tempo da velocidade angular durante o Modo de Navegação Fina da Plataforma MultiMissão (PMM) (Amaral 2008). Cada pico está associado a uma rotação para a aquisição da atitude nominal. Os primeiros dois picos exibem “bumps”, marcados com setas azuis, durante o chaveamento do controle grosseiro para o controle fino.

Mesmo que não haja impacto na saída da planta, o transiente desnecessário do módulo da velocidade angular implica uma aquisição mais demorada. Testamos diferentes abordagens para melhorar a suavidade e duração da transição.

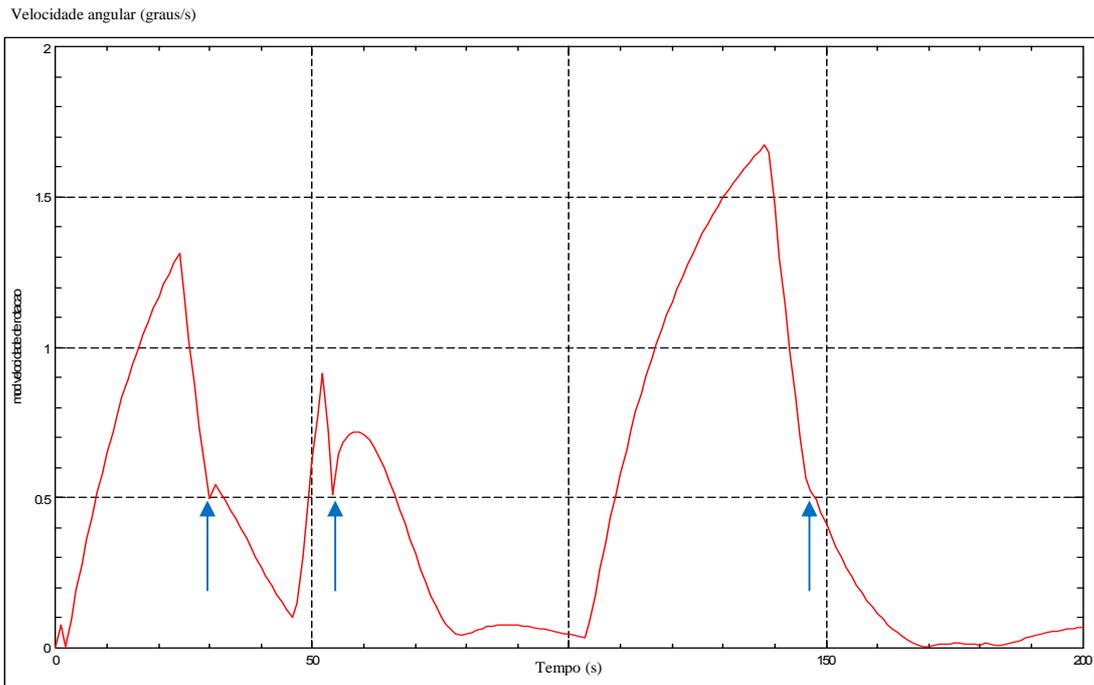


Figura 2-3: Resposta no tempo da velocidade angular durante três rotações consecutivas para aquisição da atitude nominal durante o Modo de Navegação Fina da PMM.

Fonte: Amaral (2008).

O par de motores CC encontrado em (Oliveira et Souza 2012) (Figura 2-4), ou casos como o ExoMars Rover (Höplinger et al. 2008) (Figura 2-5) são exemplos de plantas com menos inércia, onde o impacto está sujeito a não se manifestar apenas no sinal de controle, mas também na saída da planta. Os modelos da PMM e dos motores CC já foram usados em simulações neste trabalho e serão mostrados nos capítulos seguintes, enquanto casos como o último será uma possível extensão para trabalhos futuros.

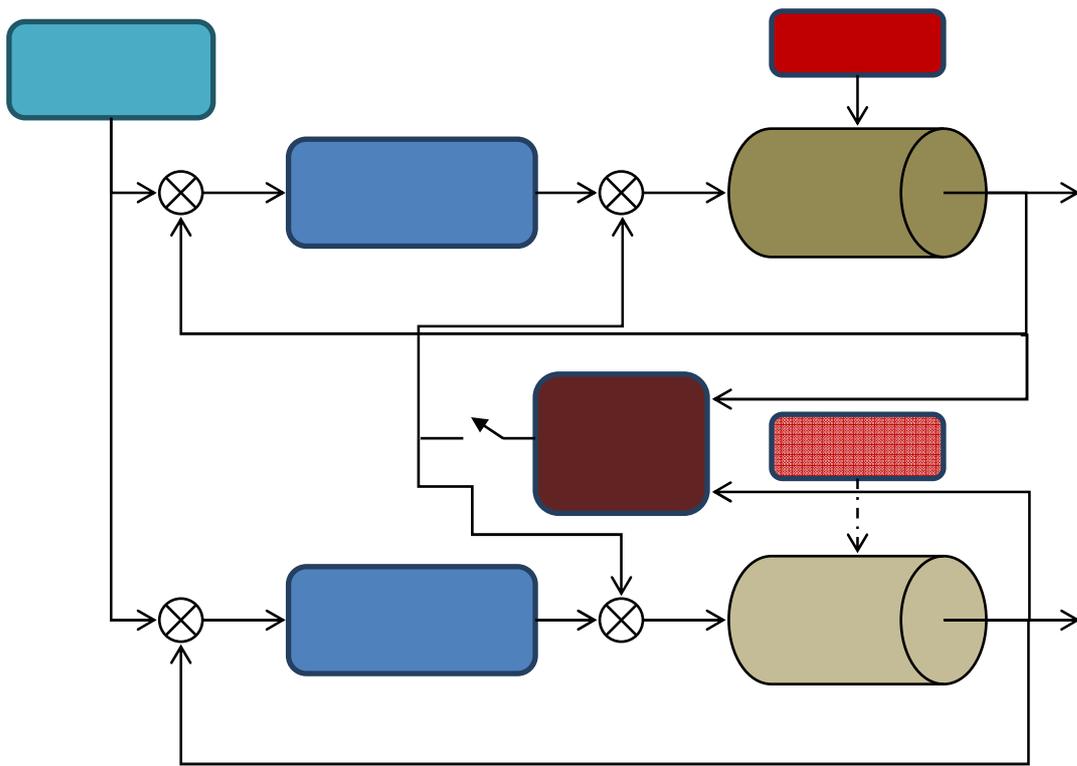


Figura 2-4: Modelo de motor CC com PLL entre malhas.

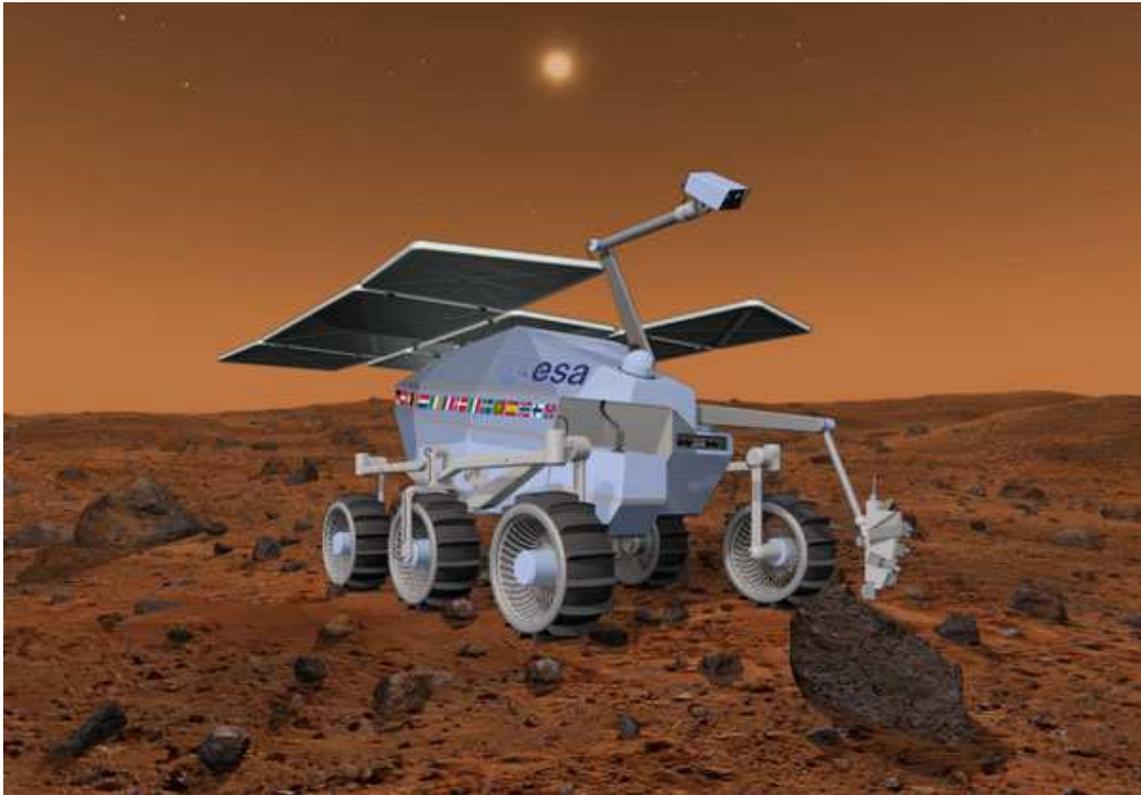


Figura 2-5: ExoMars Rover
Fonte: Vinci (2012).

2.4. Abordagem para Tratamento do Transitório Durante a Reconfiguração

De acordo com (Simon et al. 2002), há várias abordagens para lidar com sistemas híbridos (analógicos-digitais). No entanto, poucos trabalhos lidam com o problema do transitório durante as reconfigurações. As abordagens típicas vistas são:

1. reconfiguração de passo único;
2. reconfiguração de passos múltiplos (série de reconfigurações de passo único);
3. métodos de “cross-fading” de entrada;

4. métodos de “cross-fading” de saída;
5. suavização de sinal;
6. métodos de atualização de variáveis de estado;

A abordagem 1 pode ser considerada como o simples chaveamento de uma configuração para outra. As abordagens 2-5 são técnicas que podem reduzir fenômenos transitórios pagando o preço de um intervalo de transição (muito) mais longo, enquanto a abordagem 6 propõe mudar, sempre que possível, os valores das variáveis de estado, pois os transitórios se devem às diferenças dessas variáveis antes e depois da reconfiguração. O problema dos transitórios de reconfiguração é bem compreendido por comunidades de processamento de sinal de controle e áudio (Simon et al. 2002). No entanto, existem poucos relatos de pesquisa sobre estratégias para suprimir esses transitórios em sistemas de controle reconfiguráveis e/ou variantes no tempo. Os resultados disponíveis são válidos principalmente para sistemas lineares ajustáveis, nos quais os parâmetros são subitamente modificados (Välímäki et Laakso 1998).

(Graebe & Ahlén 1996) apresentam um esquema para transferência “bumpless” bidirecional entre sistemas sem observação de estados nem necessidade de que o primeiro estado esteja estacionário. Isso pode ser aplicado a sistemas onde um controlador analógico com estado apenas parcialmente conhecido é chaveado para um novo controle digital ou adaptativo sem conhecer nenhum estado. Caso o novo controle se mostre instável, isso permite chavear de volta para o controlador antigo.

(Yamé & Kinnaert 2007) basearam-se numa profunda caracterização do fenômeno “bump”, e forneceram uma parametrização simples para grupos de controladores lineares, que podem ter diferentes dimensões de estados. Nela há uma memória em comum e uma dinâmica partilhada por todos os controladores no conjunto, e ela faz cada controlador instável em malha aberta manter-se estável mesmo quando não é usado.

2.5. Sensibilidade de Trajetórias e Rastreamento de Trajetórias

Em reconfigurações de sistemas onde os estados estacionários são fáceis de ser definidos por poderem ser representados por modelos simples, como valores constantes ou periódicos, a interpretação de transitórios é bastante simples. No entanto, em reconfigurações de sistemas onde os estados estacionários são difíceis de ser definidos, a interpretação de transitórios deve ser caracterizada como um movimento complexo e contínuo. Esses sistemas estão permanentemente em “transitório”, e são representados e projetados a partir de trajetórias. Transitórios devidos à reconfiguração modificarão essas trajetórias, e apenas abordagens baseadas em modelos podem ajudar a separar e/ou compensar tais efeitos. Em nosso conhecimento até agora, não há métodos sistemáticos disponíveis para executar tal separação ou compensação de sinal. Os estudos sobre a análise de sensibilidade de trajetórias (Hiskens et Pai 2000), predição de estabilidade de transitório (Liu et al. 2000), e rastreamento de trajetórias (Torfs et al. 1998) também podem ajudar a encontrar soluções para o gerenciamento do problema do transitório.

2.6. Convergência de Algoritmos de Adaptação

Há muitos artigos (Carini et al. 1999), (Moustakides 1997), (Mayyas et Aboulnasr 1998), que investigam as propriedades de estabilidade e convergência de algoritmos adaptativos. O movimento de sistemas adaptativos a partir de seus estados iniciais para estados finais (estacionários) pode ser considerado como um movimento transitório, portanto cada método para a caracterização das propriedades de convergência pode ser útil em lidar com transitórios de reconfiguração.

2.7. Problemas em aberto sobre o gerenciamento de transitórios

De acordo com (Simon et al. 2002) há vários problemas de pesquisa e implementação que lidam com reconfigurações. O que vimos no momento para

reduzir ou eliminar transitórios de reconfiguração pode ser sumarizado da seguinte forma:

- Modelos lineares de mecanismos de computação e seleção suportando atividades durante o projeto.
- Mecanismos de estimação e inicialização de estados de controle para serem implementados em tempo de execução.
- Estratégias de injeção antitransitório baseadas em observação de estado “on-line”.

Na maioria dos desenvolvimentos, consideram-se situações onde reconfigurações são feitas enquanto o sistema geral é mantido em regime permanente. A reconfiguração aparece como uma mudança brusca, e se assume que a especificação exige alcançar um novo regime permanente. A dependência de estrutura foi investigada apenas para modelos lineares de computação, e também os exemplos ilustrativos lidam com sistemas lineares. A maioria dos métodos propostos é baseada em hipóteses válidas para sistemas lineares. Entretanto, devido à natureza não linear dos sistemas chaveados, não se pode generalizar estes métodos para tratar transitórios de reconfigurações de sistemas não lineares.

De acordo com (Simon et al. 2002), os mais importantes problemas de pesquisa nesse campo são os seguintes:

1. Extensão das técnicas já existentes em direção a sistemas de controle não lineares.
2. Investigações para ter mais entendimento em reconfigurações conjuntas de controlador e planta.

3. Investigação de reconfigurações onde, em adição aos parâmetros, a estrutura também é modificada (e.g., mudando para um controlador de ordem diferente.)
4. Investigação de dependência de estrutura em caso de modelos não lineares de computação (estruturas não lineares).
5. Elaboração de métodos para separar e/ou compensar componentes de trajetória no caso de sistemas sem estados estacionários.
6. Extensão dos resultados disponíveis para o caso de reconfigurações de múltiplos passos.
7. Análise da carga computacional de gerenciamento de reconfiguração e transitório.

2.8. Plataforma de Controle Aberta (GeorgiaTech)

De acordo com (Guler et al. 2003), um suporte genérico para a gestão da transição (“Transition Management-TM”) está sendo desenvolvido na Georgia Tech para suportar sistemas de controle híbrido como parte de um esforço maior (em colaboração com a Boeing, Honeywell, e a Universidade da Califórnia em Berkeley) para criar uma Plataforma de Controle Aberta (“Open Control Platform - OCP”) baseada em “middleware” orientado a objetos e computação para objetos distribuídos. Essa plataforma coordena a interação distribuída entre diversos componentes de sistemas de controle e suporta a reconfiguração e customização dinâmica de componentes em tempo real. Ela provê um suporte mais abrangente do que as ferramentas existentes para integrar componentes distribuídos, enquanto esconde detalhes de componentes distribuídos do desenvolvedor do sistema de controle. Isso também vai além do suporte a desenvolvimento apenas, possibilitando a adaptação rápida durante o processo e reconfiguração dinâmica de sistemas

de controle. A OCP está sendo usada no controle autônomo de UAVs na Georgia Tech (UAVs).

2.9. Efeitos da Reconfiguração sobre a Estabilidade

A capacidade de alternar modos de controle cria dificuldades para determinar a estabilidade de um sistema. Se todos os subsistemas forem lineares e assintoticamente estáveis, a estabilidade geral pode ser garantida. No entanto, caso existam não linearidades, isso não pode ser garantido. Determinar isso constitui o **Problema da Hiperestabilidade** (Popov 1973), que permanece em aberto.

Da mesma forma, entre as questões mais relevantes associadas à estabilidade de sistemas chaveados, está a de encontrar condições que garantam que o sistema seja assintoticamente estável para qualquer sinal de chaveamento (Narendra et Balakrishnam 1994), (Simon et al. 2002), (Chatterjee et Liberzon 2007). Em vez de abordar o caso geral, autores frequentemente se dedicam a encontrar soluções em situações particulares, como por exemplo, quando se assume que a sequência de chaveamento é suficientemente lenta (Liberzon et Morse 1999).

Na parte superior da Figura 2-6, são mostrados os planos de fase de dois sistemas não lineares e assintoticamente estáveis, com linhas contínua e tracejada, respectivamente. Eles podem ser chaveados para produzir os sistemas mostrados na parte de baixo da Figura 2-6: o da esquerda converge ainda mais rapidamente, enquanto o da direita é instável.

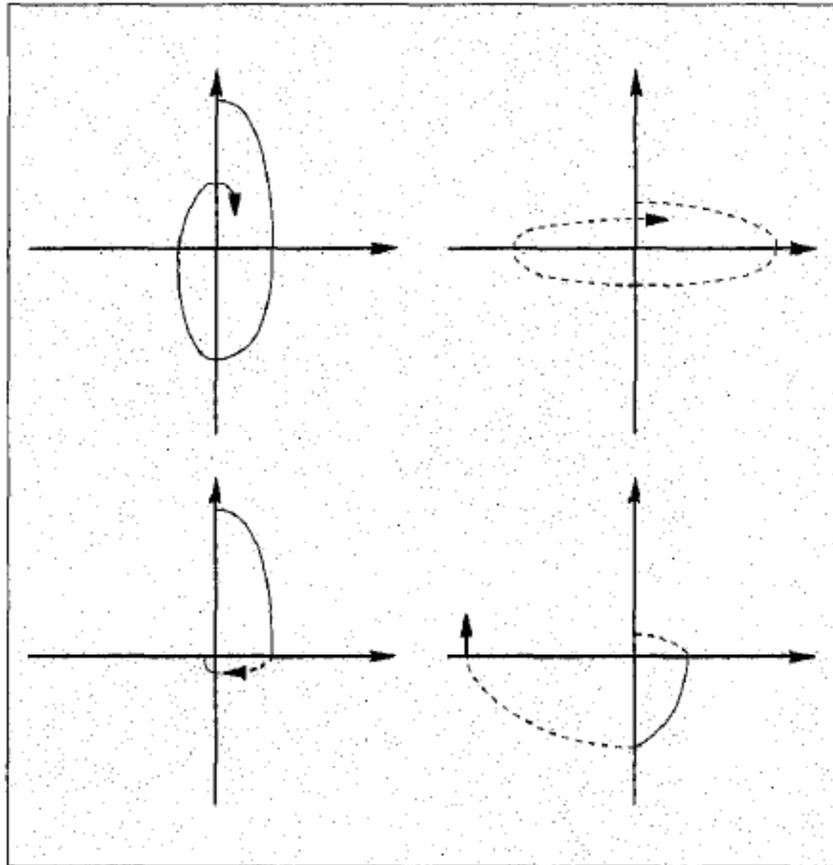


Figura 2-6: Exemplo onde dois sistemas individualmente assintoticamente estáveis podem produzir um sistema composto instável
 Fonte: Liberzon et Morse (1999).

Outras questões incluem identificar que classes de sinais de chaveamento tornam o sistema assintoticamente estável, e construir um sinal de chaveamento que torne o sistema assintoticamente estável.

Em situações onde dois sistemas chaveiam ciclicamente entre estruturas, (Middlebrook et Ćuk 1977) estabeleceram um método para modelar conversores chaveados, criando um espaço de estados unificado, baseando-se na média entre cada subsistema, ponderada pelo seu “duty cycle”. Entretanto, o método não leva em consideração o transitório entre os modos.

2.10. Ferramentas de Simulação

Para realizar as simulações, utilizaremos as ferramentas MATRIXx/SystemBuild por serem conhecidas e adequadas aos sistemas simulados, como visto em (Gobato 2006) e (Moreira 2006). O desenvolvimento de sistemas de tempo real usualmente ocorre em estágios, com ferramentas separadas para o projeto de controle, engenharia de software, aquisição de dados e testes. A família de produtos do MATRIXx integra as ferramentas para cada estágio do desenvolvimento do sistema em um único ambiente. Isso permite a rápida transição de um estágio para o outro tornando possível criar um protótipo funcional ainda cedo durante o processo de desenvolvimento.

2.11. Conexão da Literatura com este Trabalho

Da literatura encontrada, as abordagens mais relevantes para atingirmos o objetivo desta Tese são: a “cross-fading” (Simon et al. 2002), “anti-windup” (Görçek et al. 2002), rastreamento de trajetória (Torfs et al. 1998), e critérios integrais (Ogata 1970).

3 FORMULAÇÃO DO PROBLEMA

O problema que queremos estudar são os efeitos da reconfiguração sobre o transitório e a estabilidade de sistemas de controle reconfiguráveis. Para isto, podemos aplicar uma das abordagens mencionadas anteriormente e detalhadas na Seção 3.2 a um dos sistemas mencionados anteriormente e detalhados na Seção 3.1, segundo a Tabela 3-1. Esta mostra com X quais abordagens puderam ser aplicados em quais sistemas nesta Tese, no tempo disponível. Além do uso comum encontrado na literatura, a abordagem de “cross-fading” foi estendida para ser aplicada no sinal de erro; e a abordagem de rastreo de saída foi simplificada para utilizar um sinal de pré-inicialização. Além disso, elas foram testadas simultaneamente com técnicas de “anti-windup”. A aplicação dos critérios integrais do quadrado do erro (ISE), do tempo vezes quadrado do erro (ITSE), do módulo do erro (IME) e do tempo vezes o módulo do erro (ITME), para determinar uma trajetória ótima para a transição entre sinais de controle, foi uma contribuição original. A análise da função-transferência do sistema durante a etapa de “cross-fading” exponencial ou linear também foi uma contribuição original.

Tabela 3-1 Casos vs. abordagens aplicadas.

caso\abordagem	“cross-fading”	“anti-windup”	Rastreo de saída	Critérios integrais	Análise de transitório
Motores CC	X			X	
PMM	X	X	X	X	X

3.1. Sistemas em Estudo

Esta seção descreve os sistemas em estudo utilizados nas simulações, que foram a Plataforma Multimissão e o par de motores CC.

3.1.1. A Plataforma Multimissão

Segundo [Moreira, 2006], a Plataforma Multi-Missão foi concebida para ser uma plataforma modular capaz de servir como base para várias missões científicas, de comunicação e de observação da Terra em baixas órbitas terrestres. A PMM é constituída de subsistemas básicos que fornecem o essencial para o funcionamento do satélite e um suporte para a integração de uma carga útil que será escolhida de acordo com a missão que o satélite irá desempenhar.

Uma visão da PMM, com os painéis solares abertos e o envelope previsto para a carga útil, se encontra na Figura 3.1.

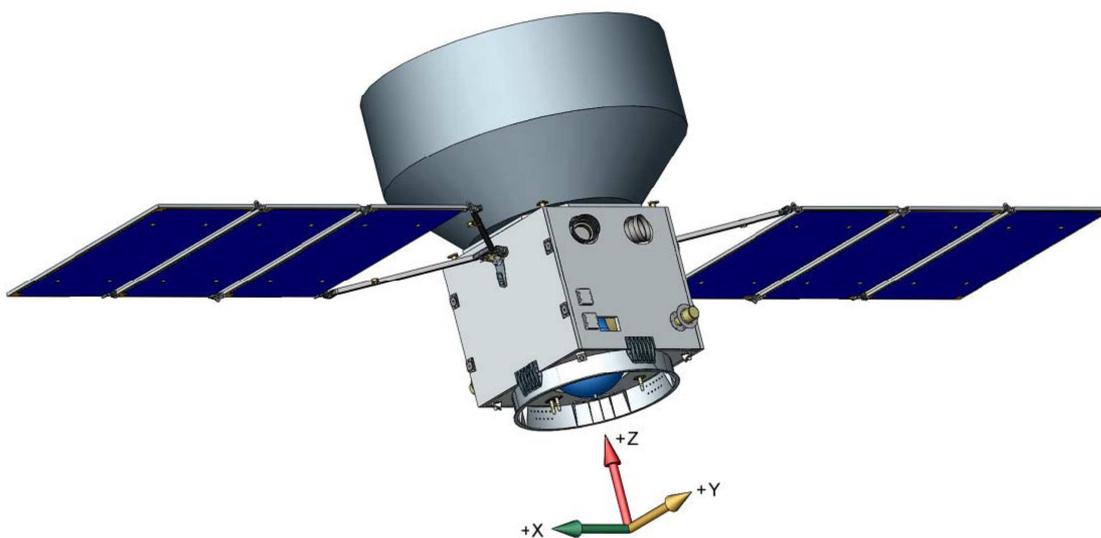


Figura 3-1: – Plataforma MultiMissão com painéis solares abertos e espaço destinado à carga útil.

Fonte: INPE (2001b).

3.1.1.1. Modos de Operação

Devido à diversidade de condições em que um satélite irá encontrar durante toda sua vida, faz-se uma separação em vários Modos de Operação, onde cada modo é definido pelo ambiente e condição em que o satélite se encontra. Estes modos são divididos em dois grupos maiores definidos pelo ambiente em que o satélite encontra-se, como visto abaixo:

- Modos de Solo
 - Modo Desligado (“Off Mode” - OFM) - Nesse modo todos os equipamentos são desligados (com a bateria desconectada). Este modo serve para armazenamento e transporte.
 - Modo de Integração e Teste (“Integration and Test Mode” - ITM) - Esse modo é usado durante a montagem e testes de integração ou na plataforma de lançamento. Durante a montagem e integração, todos os testes são realizados enquanto na plataforma de lançamento só se realizam os testes de verificação de funcionamento.
- Modos de Voo
 - Modo de Inicialização (“Start Mode” - STM) - Esse modo deve ser usado em Terra, durante a fase de vôo e pode ser usado a qualquer momento durante a vida útil do satélite.
 - Modo de Contingência (“Contingency Mode” - COM) - O objetivo desse modo é automaticamente levar o satélite e sua carga útil do STM para um modo seguro após a separação do lançador ou no caso da detecção de uma anomalia.
 - Modo de Navegação Fina (“Fine Navigation Mode” - FNM) - Esse modo é usado para aquisição de atitude, posição e tempo de forma

precisa para permitir a transição do modo de contingência para o modo nominal.

- Modo Nominal (“Nominal Mode” - NOM) - Esse é o modo operacional do satélite, onde a carga útil pode cumprir seus objetivos. Neste modo também ocorre a dessaturação com os atuadores magnéticos.
- Modo de Dessaturação das Rodas com Propulsores (“Wheels Desaturation Mode with Thrusters” - WDM) - Nesse modo é realizada a dessaturação das rodas de reação por intermédio do acionamento dos propulsores. Esse procedimento visa reduzir a velocidade angular das rodas a níveis nominais de operação.
- Modo de Correção de Órbita (“Orbit Correction Mode” - OCM) - Utilizado para realizar manobras orbitais, no plano da órbita ou para fora dele.
- Modo Salvaguarda de Correção de Órbita (“Orbit Correction Mode Backup” - OCMB) - Caso um dos propulsores não redundantes falhe, as manobras orbitais serão realizadas com somente dois dos propulsores simétricos para minimizar os torques de perturbação.

A lógica de transição dos modos de operação pode ser vista na Figura 3-2. As transições podem ser realizadas automaticamente (Auto), via telecomando (TC), ou devidas a alguma anomalia no satélite (Alarme).

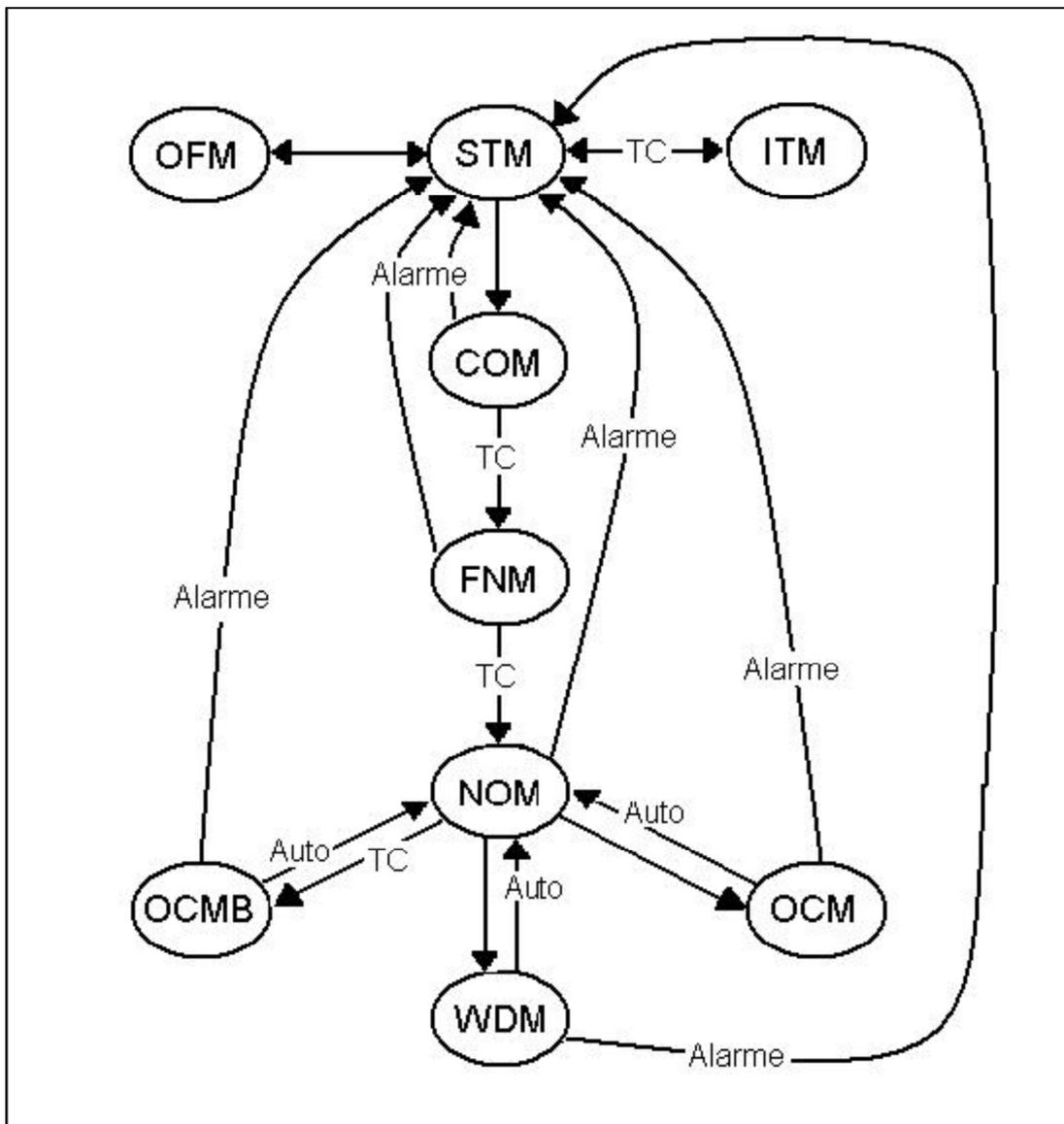


Figura 3-2: Lógica de Transição dos modos de Operação da PMM.

FONTE: INPE (2001a).

Segundo as especificações descritas em (INPE 2001^a), cada magnetômetro mede as projeções do campo magnético local nos três eixos do veículo. As unidades inerciais devem ser capazes de prover as velocidades angulares do satélite em três eixos, de forma independente. O conjunto de sensores solares deve prover informação suficiente para a determinação das componentes da direção do Sol em três eixos com completa cobertura espacial (4π srad). O

receptor GPS fornece o tempo e também a posição e a velocidade do satélite de forma autônoma. Cada sensor de estrelas provê atitude em três eixos de forma completamente autônoma. As rodas de reação, para o controle em três eixos, podem ser comandadas tanto em torque quanto em velocidade. Atuadores magnéticos geram momento magnético em três eixos para geração de torques externos.

3.1.1.2. Dinâmica

Um sistema de referência A do tipo VLHL (Vertical Local Horizonte Local) apresenta origem no centro de massa do satélite e tem vetores unitários (a_1, a_2, a_3) , sendo: a_1 na direção da órbita, a_2 perpendicular ao plano da órbita, e a_3 em direção à Terra (Wie, 1998). A velocidade de A em relação ao referencial inercial N é:

$$\vec{w}^{A/N} = w_0 \vec{a}_2 \quad (3.1)$$

onde w_0 é a velocidade orbital. A velocidade angular do referencial fixo ao corpo B , com vetores base (b_1, b_2, b_3) , é dado por:

$$\vec{w}^{B/N} = w^{B/A} + w^{A/N} = w^{B/A} - w_0 \vec{a}_2 \quad (3.2)$$

onde $w^{B/A}$ é a velocidade angular de B em relação a A .

A equação rotacional do movimento de um corpo rígido com momento angular $\vec{H} = I\vec{w}^{B/N}$ é dada por:

$$\left\{ \frac{d\vec{H}}{dt} \right\}_B \equiv \left\{ \frac{d\vec{H}}{dt} \right\}_N + \vec{w}^{B/N} \times \vec{H} = T_p \quad (3.3)$$

onde $\vec{w}^{B/N}$ é o vetor velocidade angular do satélite (referencial do corpo B em relação ao referencial inercial N), I é a matriz de inércia do satélite e T_p é o torque devido às perturbações externas.

O modelo da Plataforma Multi-Missão segue as equações de movimento de um veículo espacial rígido (WIE 1998), levando em consideração a presença da roda de reação e seu acoplamento com o satélite (Wertz 1978). Portanto a equação dinâmica da planta é:

$$I\dot{\vec{w}} + \vec{w} \times (I\vec{w}) = \vec{w} \times h - \dot{h} + T_p \quad (3.4)$$

onde I é a matriz de inércia do satélite, w é o vetor velocidade angular do satélite (referencial do corpo B em relação ao referencial inercial N), h é o momento angular das rodas de reação, \dot{h} é a variação do momento angular das rodas de reação, que representa a parcela controlável da equação (também chamado de torque de controle), e T_p é o torque devido às perturbações externas.

3.1.1.3. Modo de Navegação Fina

Neste trabalho, os testes se concentraram no Modo de Navegação Fina. O propósito desse modo é levar a PMM para sua atitude nominal, e então chavear para o Modo nominal. Nesta simulação, a atitude nominal é em direção à Terra (vertical local, horizonte local). A PMM executa um máximo de três rotações. Cada rotação (Figura 3-3) começa guiada por um controle grosseiro do tipo “bang-bang”, que é revertido quando o erro absoluto está abaixo de um valor pré-determinado ε e, então, chaveado para um controlador fino do tipo PID quando a velocidade absoluta está abaixo de um valor predeterminado ζ .

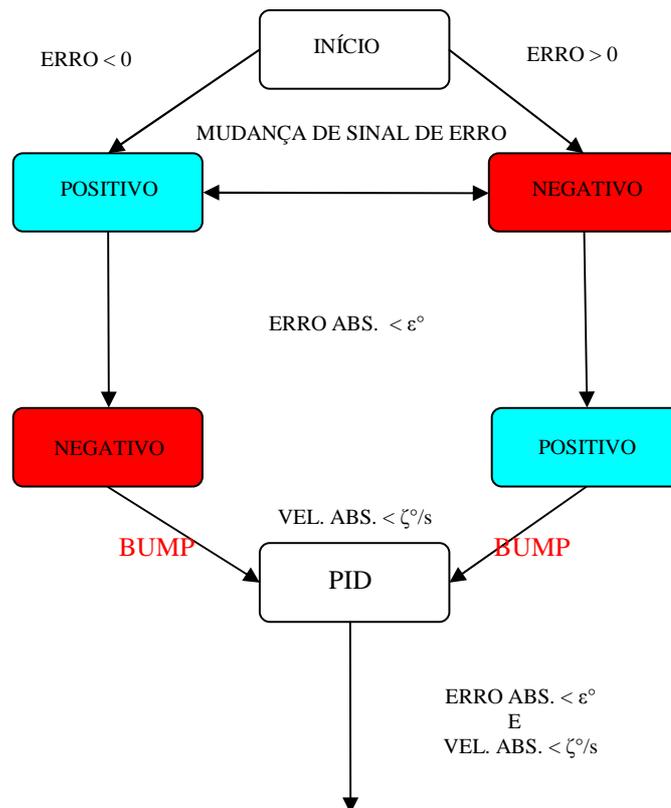


Figura 3-3: Diagrama de blocos dos submodos de uma única rotação.

O chaveamento geralmente causa um impacto, e isso é uma boa oportunidade de testar estratégias anti-impacto. Os casos mostrados se referem à primeira rotação, em torno do eixo z.

Como em (Moreira 2006), nossas simulações foram feitas no ambiente MATRIXx/SystemBuild, que permite projetar em forma de diagrama de blocos. Isso permite uma visão sistêmica do projeto, ao mesmo tempo em que guardam dentro de si os blocos de todos os subsistemas especificados anteriormente no projeto. Dessa forma, é possível alternar facilmente entre dois tipos de visão, detalhada e sistêmica.

Durante a confecção dos sub-blocos, eles podem ser testados antes de ser conectados a outros blocos. Independente de serem conectados a outros

blocos ou não, esses sub-blocos podem ser modificados novamente e/ou testados, ajudando no desenvolvimento do projeto.

A parte que se refere à PMM foi baseada num desenvolvimento anterior em MATRIXx/SystemBuild para o satélite SACI, e descrito em (Prudêncio 2000). O modelo do ambiente espacial, com o propagador de órbita, foi reutilizado completamente, com a adição das características necessárias para o Modo de Correção Orbital, não apresentado neste trabalho.

O modelo da planta, descrito pelas equações cinemáticas de equação e dinâmica, em quatérnions, foi ampliado. Houve modificações que aumentaram a influencia das rodas de reação na dinâmica, e a conservação de momento angular durante variações no momento de inércia, durante a extensão dos painéis solares. Parâmetros específicos da PMM precisaram ser adotados, tais como os momentos de inércia.

O reuso de trabalhos anteriores foi importante para que o trabalho presente pudesse atingir outros níveis de desenvolvimento. O reuso acelerou o processo de modelagem e projeto de simulação.

As simulações com a PMM foram feitas considerando uma órbita polar circular de 7.000 km de raio. Para teste de desempenho, foram usadas diferentes atitude e velocidades angulares iniciais, e diferentes combinações de redução de instabilidade de transiente.

As simulações tentaram demonstrar o cumprimento dos requerimentos de projeto. Os piores casos possíveis de atitude inicial que poderiam ser encontrados na missão proposta foram simulados como exemplo de aplicação deste trabalho.

O caso geral concentra-se em voltas de 90 graus para alinhar o satélite com a desejada referência num dado eixo. No entanto, também foram executados

testes com ângulos menores para checar se o método é menos efetivo ou desnecessário em outros casos. (Amaral 2008).

3.1.2. Par de Motores de Corrente Contínua

Buscando um exemplo de planta mais rápida, também aplicamos o método de “cross-fading” para suavizar a transição entre um PID (controlador Proporcional Integral Derivativo) e PLL (“Phase Locked Loop”) numa simulação de duas malhas de controle CC como descrito em (Oliveira et Souza 2012). O sistema é descrito na Figura 3-4.

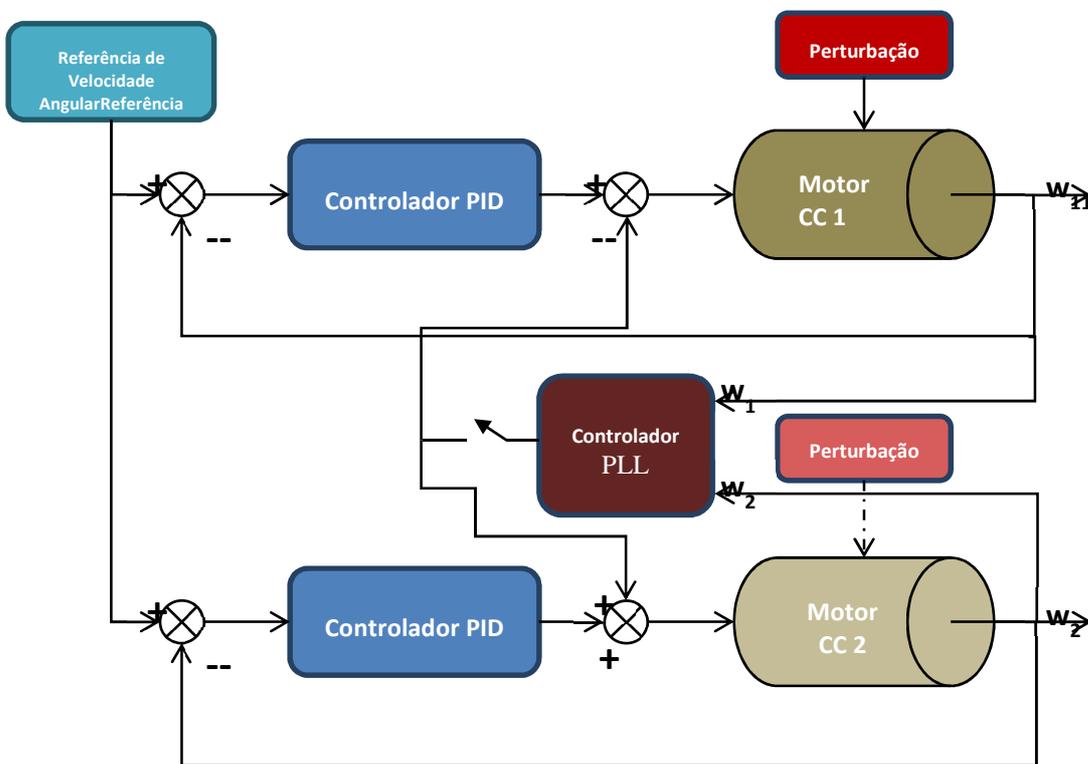


Figura **Erro! Nenhum texto com o estilo especificado foi encontrado no documento.**-1: Modelo de motor CC com PLL entre malhas.

A função de malha fechada do filtro ativo do PLL é:

$$H''(s) = \frac{K_v K_d (s\tau_2 + 1) / \tau_1}{s^2 + s\left(\frac{K_v K_d \tau_2}{\tau_1}\right) + \frac{K_v K_d}{\tau_1}} \quad (3.5)$$

e o diagrama de blocos do motor é mostrado na Figura 3-18:

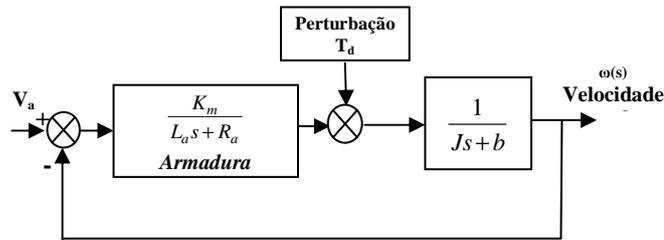


Figura 3-5: Diagrama de blocos de um motor CC genérico (Oliveira et Souza 2012).

O objetivo é manter a sincronização entre os motores. Os PIDs são mais robustos e fazem a velocidade dos motores seguirem uma referência, enquanto os controladores PLL têm uma reação mais rápida e são usados para corrigir a diferença de posição quando esta ultrapassa um módulo pré-estabelecido.

Em trabalhos anteriores como de (Oliveira et Souza 2012), foram feitas simulações com diversas estratégias de chaveamento entre PID e PLL.

3.1.3. Rovers como o ExoMars Rover

De acordo com (Höpflinger 2008), o ExoMars Rover (Figura 3-6) é um veículo com tração e mobilidade em seis rodas. A Arquitetura de Sistema de Controle de Locomoção precisa permitir operar o “rover” como um item “standalone” para testes de desempenho.

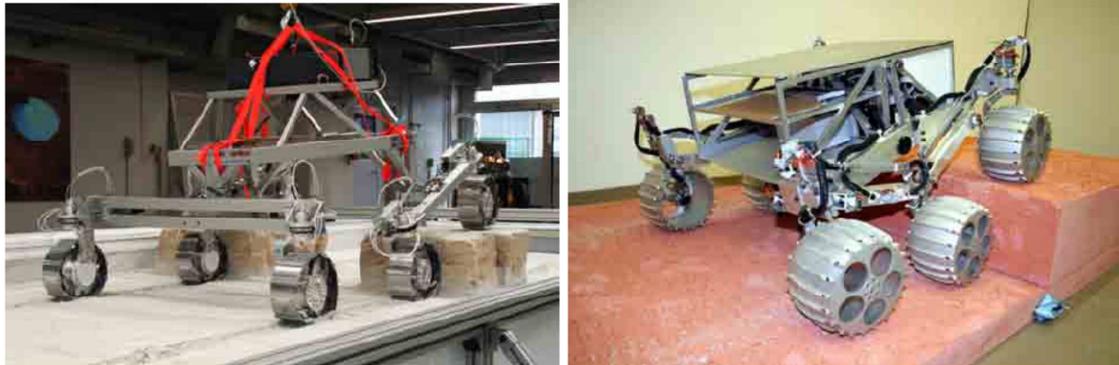


Figura 3-6: Imagens do rover usado em obstáculos
Fonte: Höpflinger et al. (2008).

Como entradas, o sistema de controle de locomoção precisa receber informação de alto nível, por exemplo, pontos de direção fornecidos em fases de projeto posteriores pelo Sistema de Navegação. O Sistema de Controle de Locomoção precisa informar posição ou velocidade para os controladores de motor para guia e torque das rodas. Devido à ausência do Sistema de Navegação, a Interface Homem Máquina (MMI) foi desenvolvida para prover ao Sistema de Controle de Locomoção com a informação necessária e para guardar os dados dos vários sensores do “rover”. O software foi implementado no CRAB2 e no ExoMars Rover.

O software de controle de locomoção permite controlar o “rover” em modos automáticos e manuais. No modo automático, o “rover” segue uma dada trajetória. Nos modos manuais, que são três, o “rover” é teleoperado diretamente pelo usuário. A Figura 3-7 mostra um diagrama dos modos e submodos de controle.

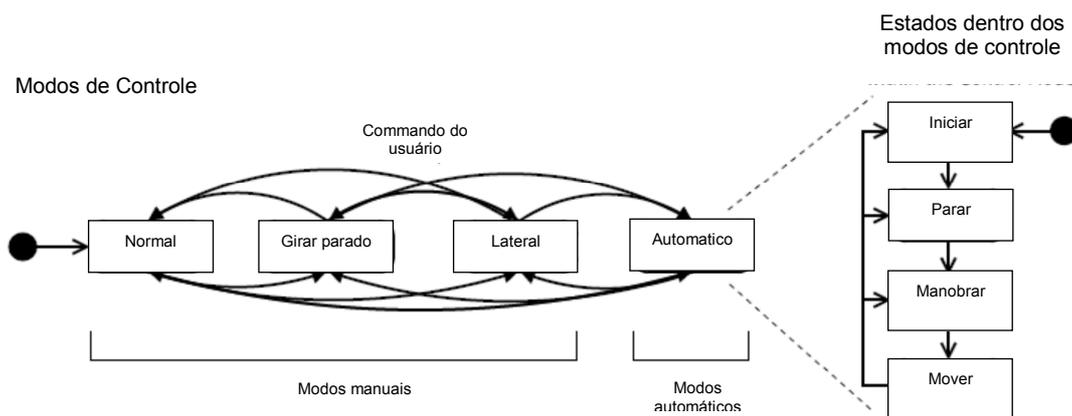


Figura 3-7: Modos e submodos de controle
 Fonte: Höpflinger et al. (2008).

3.2. Abordagens para sua Solução

As abordagens para atenuar o fenômeno do impacto (“bump”) da reconfiguração até o momento foram a mistura de sinais de saída, a mistura de sinais de entrada, o rastreo do sinal de controle, técnicas de “anti-windup”, e critérios integrais, conforme a Tabela 3-1. Deve ser notado que uma transferência sem impacto por si só elimina o risco de “windup”. Vamos detalhá-las a seguir.

3.2.1. Rastreo de Saída.

O rastreo de saída consiste em forçar o controlador final a ter um sinal de saída próximo ao do controlador inicial. As formas dos blocos que fornecem a referência e realimentação dependem da estratégia adotada. **O chaveamento é feito pelo bloco supervisorio.** Quando ele ocorre, isso faz com que a descontinuidade entre os dois controladores seja anulado ou, pelo menos, minimizado. A Figura 3-8 mostra um exemplo genérico de um controlador C2 tendo sua saída mantida em função do sinal de saída de um controlador C1. A Figura 3-9 mostra o sistema após o chaveamento, quando o bloco C2 passa a ser usado pela planta.

visto pelos atuadores como uma mudança brusca no manche, introduzindo instabilidades potencialmente catastróficas.

Como a PMM é um módulo de serviço sendo movido por rodas de reação relativamente lentas, é possível assumir que essa segunda malha é muito mais rápida que a malha original, que inclui a planta. Portanto, essa implementação é suficiente para manter ambos os controladores com saídas de valor próximo entre si.

3.2.1.1. Simplificação do Rastreo de Saída para a PMM

Como no caso da PMM lidamos com uma situação onde um controlador do tipo “bang-bang” é chaveado para um controlador do tipo PID, o sinal do controlador inicial C1 só pode ter dois valores de -10 ou +10 V, o que elimina a necessidade de um rastreo mais complexo.

Em vez de ser mantido em rastreo permanente de C1, o controlador C2 apenas recebe um sinal para inicialização, vindo do mesmo bloco supervisor que realiza o chaveamento entre modos (Figura 3-10). A Figura 3-11 mostra o sistema após o chaveamento. O sinal de inicialização dependerá apenas de o sinal de erro ser positivo ou negativo.

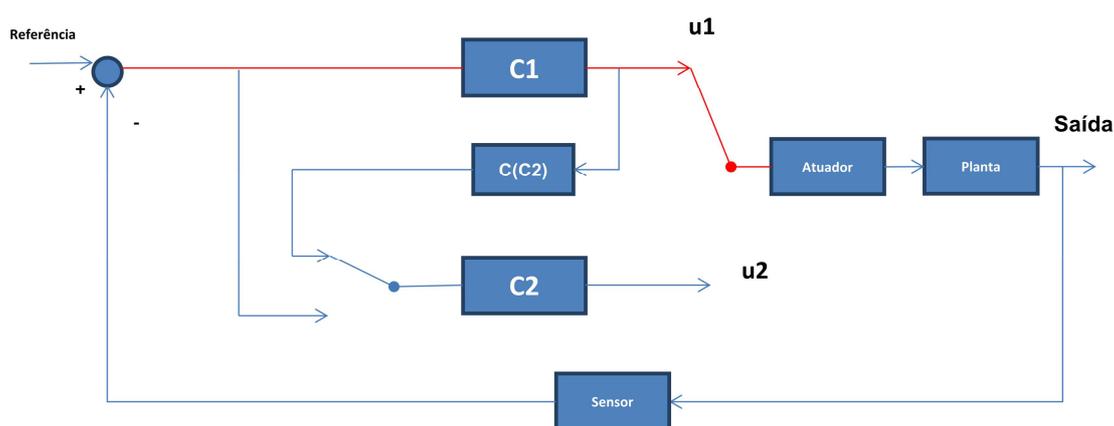


Figura 3-10: Diagrama de blocos de um sistema de controle durante a etapa de inicialização do controlador C2.

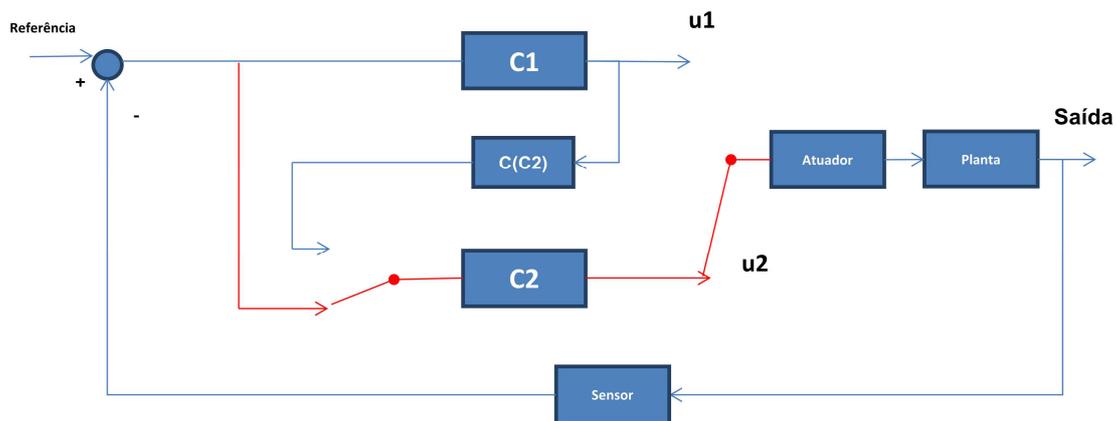


Figura 3-11: Diagrama de blocos de um sistema de controle após a etapa de chaveamento para o controlador C2.

O sinal de inicialização é uma constante, que levará a saída de C2 a um valor igual ao valor do controlador C1. Ao atingir esse valor, o chaveamento é completado, com ambos os sinais próximos entre si. Como C2 e o valor desejado são previamente conhecidos, o tempo que o sinal de C2 leva para ser inicializado não precisa ser calculado em tempo real. (Figura 3-12).

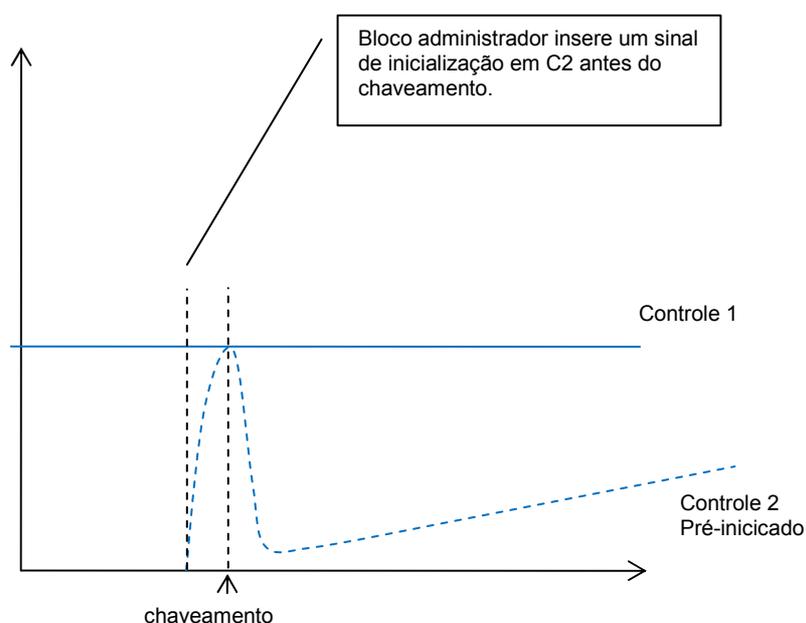


Figura 3-12: Diagrama dos sinais de controle primário (C1, contínuo) e secundário (C2, tracejado).

Assim, garantimos que as saídas terão valores não apenas próximos, mas também com derivadas próximas no momento de chaveamento (Figura 3-13). E como o controle C1 por definição não está saturando os atuadores, o controle C2 também não o fará, o que elimina a necessidade de uma estratégia “anti-windup”.

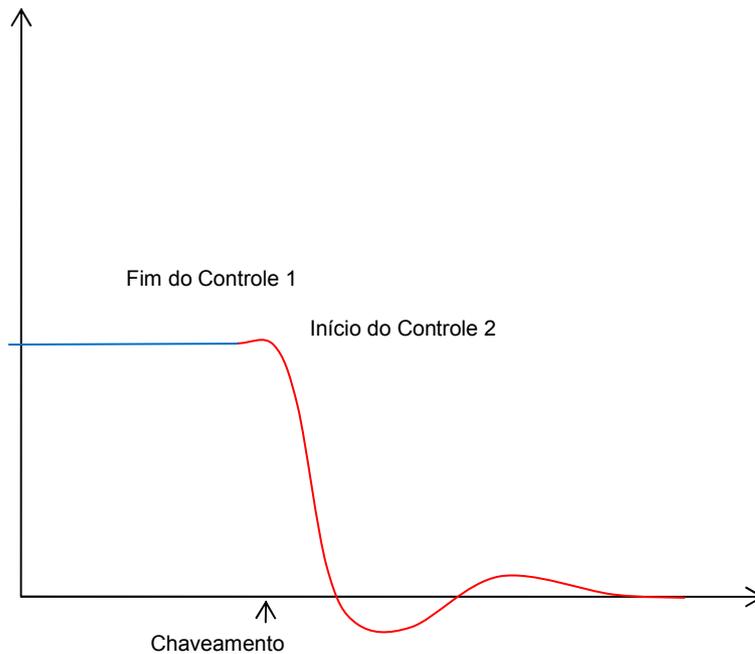


Figura 3-13: Diagrama dos sinais de controle primário (C1, azul) e secundário (C2, vermelho).

Há diferentes métodos na literatura para tratar isso, de acordo com a complexidade e requisitos de plantas não lineares com chaveamento bidirecional durante transitórios (Graebe et Ahlén 1996), até sistemas lineares em estados estacionários. (Yamé et Kinnaert 2007).

3.2.2. “Anti-windup”

O problema de “wind-up” acontece quando ocorre uma saturação que abre a malha de controle e faz a componente integral do controlador causar um sobre-sinal. Como exemplo de saturação, os ailerons de uma asa de avião podem apenas defletir em cerca de 30 graus antes de atingir limites estruturais; num

avião de caça, pode haver também um limite de velocidade de 250 graus/s nos ailerons; e para carros, há a abertura máxima para injeção de combustível. Isso pode ser muito pior para o desempenho do veículo do que uma situação onde não há esses limites (Packard 2005).

De uma forma geral e para evitar isso, a implementação consiste em medir a diferença entre o controlador u' , que tem a ação integral, e o sinal do respectivo atuador u , e então integrá-lo e subtrair esse sinal de diferença para a parte integradora desse controlador, de forma que ele cancele somente a componente integral que tenderia a crescer indefinidamente. (Figura 3-14).

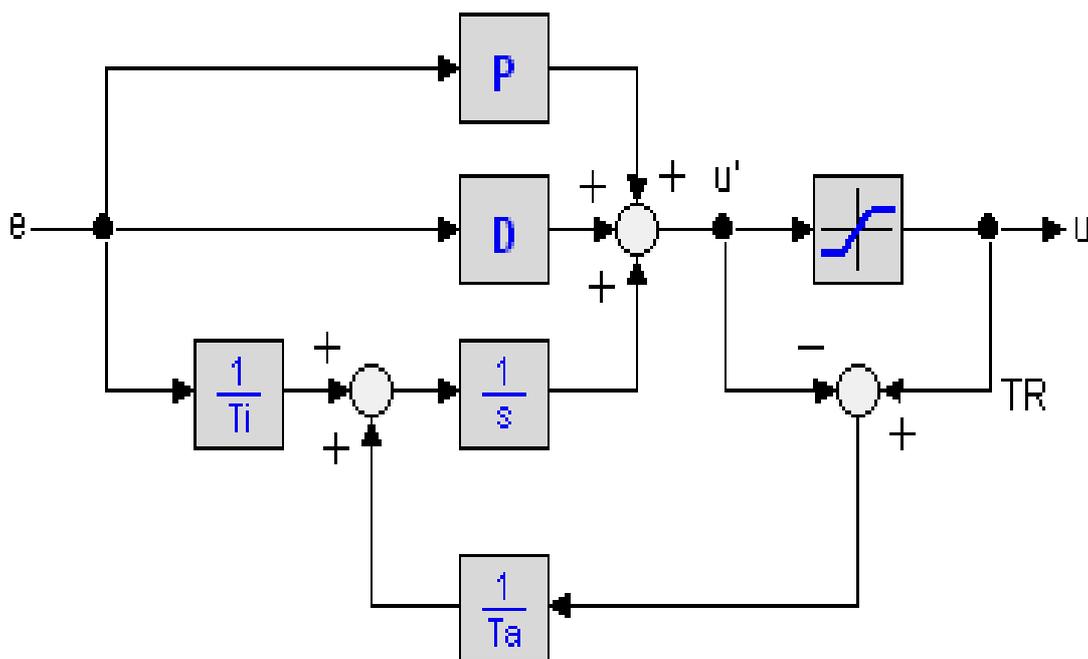


Figura 3-14: Uma de várias possíveis implementações “anti-windup” para um controlador PID.

Para prevenir que o integrador sature, a constante de tempo de rastreo escolhida deve ser pequena. No entanto, valores muito pequenos reduzem o desempenho do controlador.

3.2.3. “Cross-Fading”

O método de “cross-fading” consiste em usar ambos os sinais de controle durante uma fase de transição. Originalmente o sinal de controle é multiplexado para selecionar qual será visto pela planta (Figura 3-15).

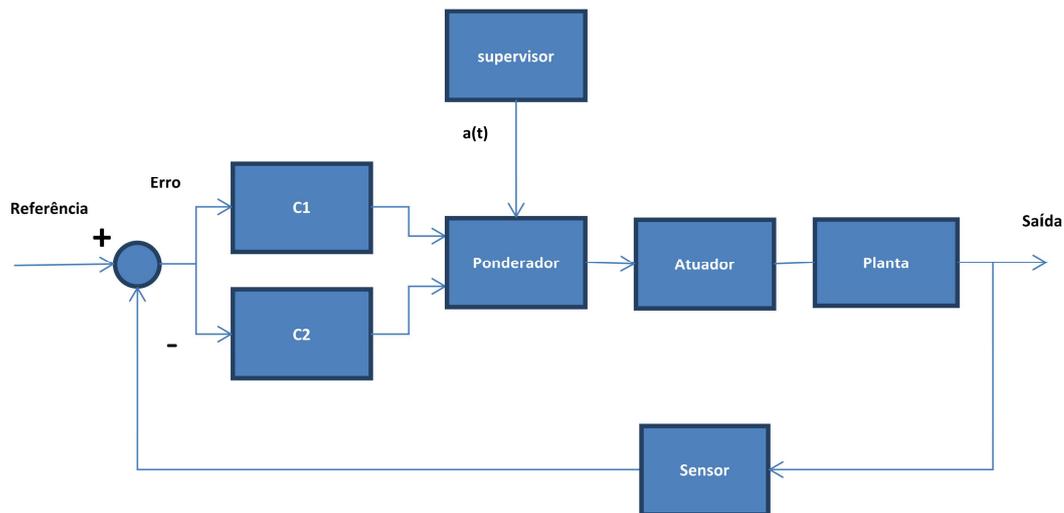


Figura 3-15: Diagrama de um sistema com “cross-fading” do sinal de controle.

O sinal resultante $u(t)$ é controlado pela função

$$u(t) = u_1(t)a(t) + u_2(t)[1 - a(t)] \quad (3.6)$$

onde $a(t)$ é uma variável que transita de 0 para 1 de uma forma suave e pré-estabelecida (Figura 3-16). A forma da função $a(t)$, assim como o tempo disponível para executar a transição, depende da planta e até mesmo do ambiente. De uma forma geral, isso só é aceitável quando a planta é muito mais lenta do que o sinal de controle.

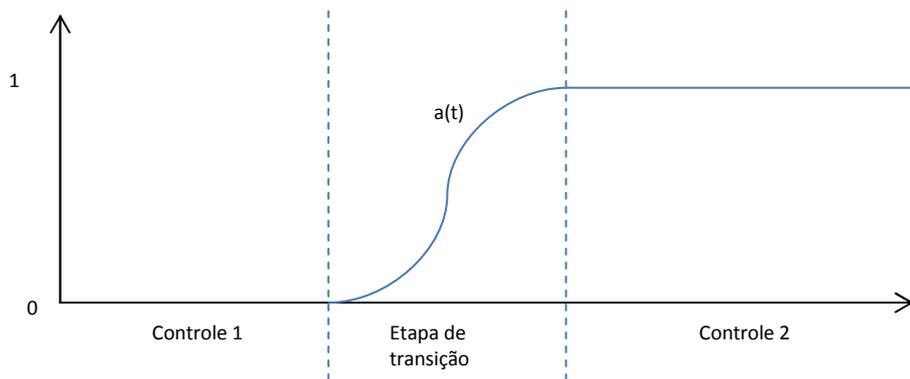


Figura 3-16: Gráfico da função de multiplexação $a(t)$ genérica ao longo do tempo.

Para evitar problemas como o “windup”, existe a alternativa de multiplexar o sinal de erro para cada controlador (Figura 3-17).

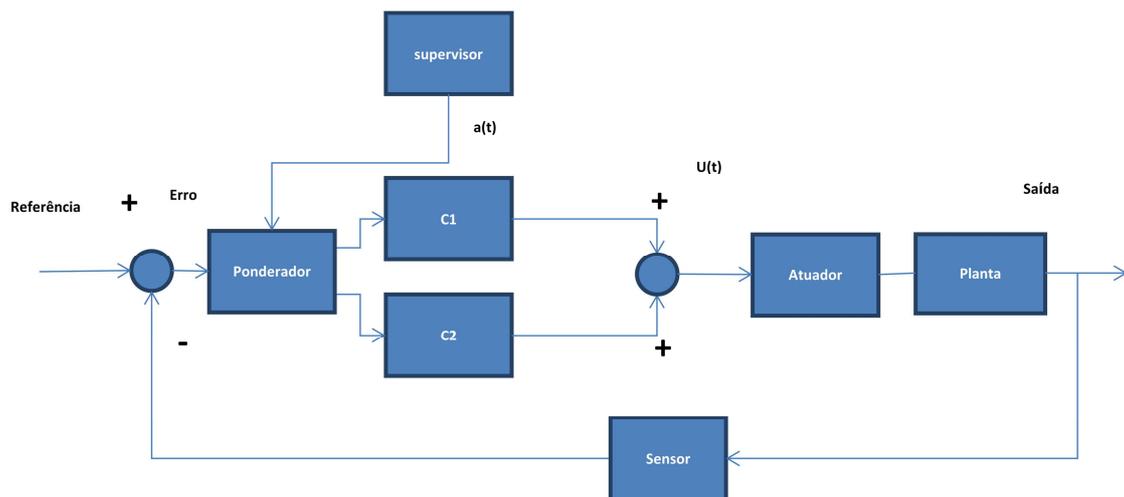


Figura 3-17: Diagrama de um sistema com “cross-fading” no sinal de erro.

Foram testadas desde uma função rampa até funções mais complexas, como a resposta de um bloco $1/s^3$ a um degrau. A função $a(t)$ foi obtida amortecendo em diferentes intensidades o degrau original que seria usado para controlar a transição num multiplexador.

3.2.4. Critérios Integrais

De acordo com (Ogata 1985), no projeto de um sistema de controle, é importante que o sistema alcance certos índices de desempenho especificados. Como sistemas de controle são dinâmicos, as especificações de desempenho podem ser dadas em termos de comportamento em resposta a transitórios para determinadas entradas, tais como degraus, rampas, etc. ou as especificações podem ser dadas em termos de um índice de desempenho.

Um índice de desempenho é um número que indica o quão bom é o desempenho de um sistema. Um sistema de controle é considerado ótimo se os valores dos parâmetros são escolhidos tais que o índice de desempenho selecionado seja máximo ou mínimo. Os valores ótimos dos parâmetros dependem diretamente do índice de desempenho selecionado.

Um índice de desempenho deve oferecer seletividade, isto é, um ajuste ótimo dos parâmetros deve claramente distinguir-se de ajustes não ótimos. Além disso, um índice de desempenho deve mostrar um único valor positivo ou zero, este sendo obtido se e somente se a medida de desvio é idêntica a zero. Para ser útil, um índice de desempenho deve ser uma função dos parâmetros do sistema e deve exibir um máximo ou mínimo. Para ser prático, tal índice deve ser fácil de computar, analiticamente ou experimentalmente.

Mostraremos em seguida critérios de erro nos quais os índices de desempenho correspondentes são integrais de uma função ou função ponderada do desvio da saída real do sistema em relação à saída desejada. Como os valores das integrais podem ser obtidos como função do parâmetro do sistema, uma vez que um índice de desempenho é especificado, um sistema ótimo pode ser projetado ajustando os parâmetros para obter o menor valor possível da integral.

Exemplos incluem a **Integral do Quadrado do Sinal de Erro (ISE)**, a **Integral do Tempo Vezes o Quadrado do Erro (ITSE)**, a **Integral do Módulo do Erro (IAE)**, e a **Integral do Módulo do Erro Vezes o Tempo (ITAE)**:

$$J_{ISE} = \int_0^{\infty} e^2(t)dt \quad (3.7)$$

$$J_{ITSE} = \int_0^{\infty} te^2(t)dt \quad (3.8)$$

$$J_{IME} = \int_0^{\infty} |e(t)|dt \quad (3.9)$$

$$J_{ITME} = \int_0^{\infty} t|e(t)|dt \quad (3.10)$$

Se um sistema de controle tem uma saída desejada $r(t)$ e uma saída real $y(t)$, o erro é definido como:

$$e(t) = r(t) - y(t) \quad (3.11)$$

Se considerarmos que o degrau de uma transição entre sinais de controle é um sinal de erro, podemos utilizar os critérios integrais para qualificar um parâmetro no sistema que produza uma trajetória de transição entre o valor inicial e final desse degrau: é estabelecida uma relação entre esse parâmetro e o resultado da integral. A definição dos critérios contém o valor do tempo final tendendo ao infinito, e isso não ocorre num sistema realista; no entanto, isso é uma aproximação válida quando a transição é muito mais rápida do que o resto do sistema.

4 ESTUDO E REDUÇÃO DOS EFEITOS DA RECONFIGURAÇÃO SOBRE OS TRANSITÓRIOS

Neste Capítulo, serão apresentadas quatro formas originais de determinar uma forma de transição entre dois modos de controle, utilizando quatro critérios integrais; e também duas formas originais de analisar a Função Transferência de um sistema durante uma transição em “cross-fading”.

Todos os critérios integrais mencionados no Capítulo 3 resultarão em zero quando a transição for o próprio degrau. Portanto, caso os parâmetros do sistema possam executar uma transição muito brusca, é importante que haja condições de contorno, como por exemplo, limitações nas derivadas ou energia associada. Uma possibilidade alternativa é buscar parâmetros que, em vez de um valor mínimo, resultem em valores a uma distância pré-estabelecida do valor mínimo.

4.1. Exemplos

Como fonte de exemplos iniciais, adotaremos o modelo da PMM para um eixo de rotação. Como descrito em (Amaral 2008), as funções de transferência completa $T(s)$, e do controlador $C(s)$, atuador $A(s)$, planta $G(s)$, e do sensor $S(s)$ são:

$$T(s) = \frac{C(s)A(s)G(s)}{1 + C(s)A(s)G(s)S(s)} \quad (4.1)$$

$$C(s) = \frac{k_p s + k_d s^2 + k_i}{s} \quad (4.2)$$

$$A(s) = \frac{k_v s}{s + \frac{1}{T_v}} \quad (4.3)$$

$$G(s) = \frac{1}{I s^2} \quad (4.4)$$

$$S(s) = 1 \quad (4.5)$$

Os valores das constantes utilizadas são:

$$T_v = 20s$$

$$K_v = 0,06Nm/V$$

$$I_x = 295,97401 \text{ kg} \cdot \text{m}^2$$

Os métodos de critérios integrais originalmente eram aplicados na análise e otimização do sinal de erro $e(t)$. Aqui, estamos interessados nos efeitos do “bump” sobre o sinal de controle $u(t)$ ou sobre a saída da planta $y(t)$, identificados na Figura 4-1. De uma forma geral, isso implica definir o problema de uma forma diferente da tradicional.

A diferença mais importante aqui é que, ao contrário do sinal de erro $e(t)$, o sinal de controle $u(t)$ e a saída da planta $y(t)$ não tenderão a zero de uma forma geral, num funcionamento normal. No cálculo dos parâmetros, é necessário subtrair o valor de chegada, a saída do controlador $u_2(t_s)$, para que o resultado realmente tenda a zero, e o critério integral possa ser aplicado.

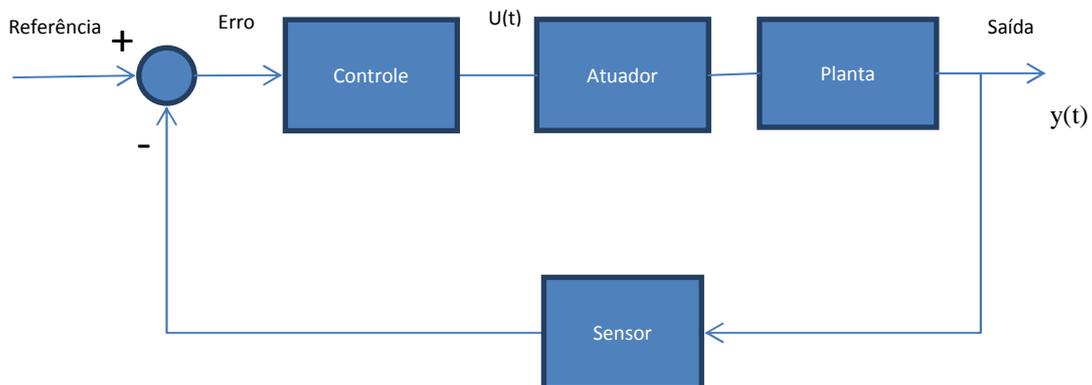


Figura 4-1: Diagrama em blocos de um sistema genérico.

4.2. Aplicação

No caso de um sistema reconfigurável, como visto na Figura 4-2, existe o chaveamento de um bloco de controle C_1 para o controle C_2 , e de uma forma geral não terão o mesmo sinal de saída no instante de chaveamento, ainda que recebam o mesmo sinal de erro.

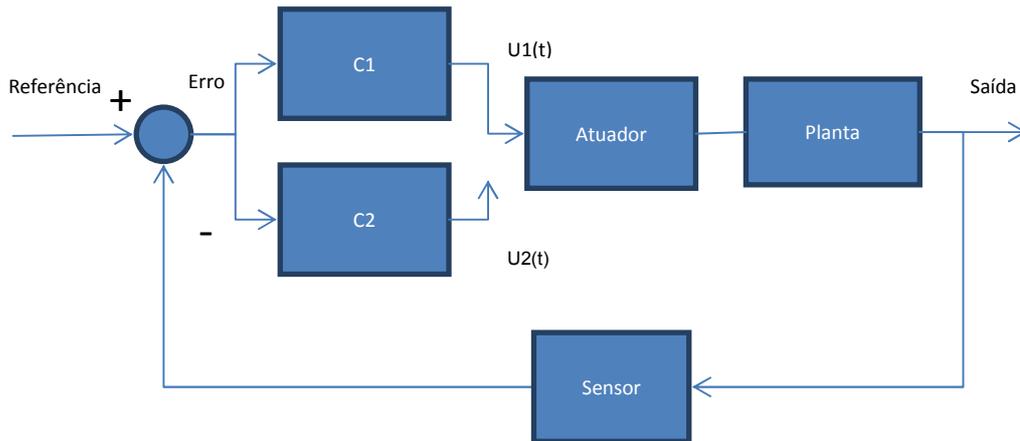


Figura 4-2: Diagrama em blocos de um sistema que chaveia entre dois controladores.

Para evitar a entrada de um degrau no bloco atuador, implementamos um modo de controle intermediário, que utiliza um novo bloco, que chamaremos de supervisor. No instante t_0 em que o chaveamento é desejado, o bloco supervisor registra o sinal $u_1(t)$ do controlador C_1 e o sinal $u_2(t)$ do controlador C_2 , que ainda não estará sendo usado. Essa diferença de valores seria a descontinuidade lançada no atuador, caso o chaveamento ocorresse sem atenuação. Conhecendo esse valor, um novo bloco de controle de $C1/2$ é introduzido no sistema, que receberá um sinal de erro baseado na referência gerada pelo supervisor, e terá parâmetros estabelecidos por um critério integral. Ao ser introduzido no sistema, a saída de $C1/2$ deverá transitar desde o valor registrado de $u_1(t_0)$ até o valor registrado de $u_2(t_0)$, da melhor forma possível segundo o critério escolhido (Figura 4-3).

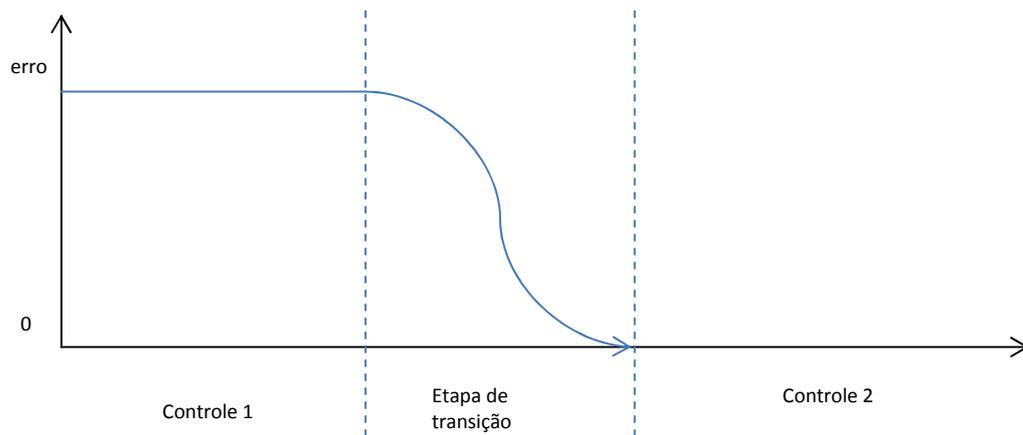


Figura 4-3: Sinal de erro reduzido segundo um critério integral durante a etapa de transição.

Como o objetivo é modificar o comportamento do sinal de controle observado pelo atuador, efetivamente ele se torna a nova saída do sistema. Para a aplicação do critério integral, convém visualizar o sistema como na Figura 4-4:

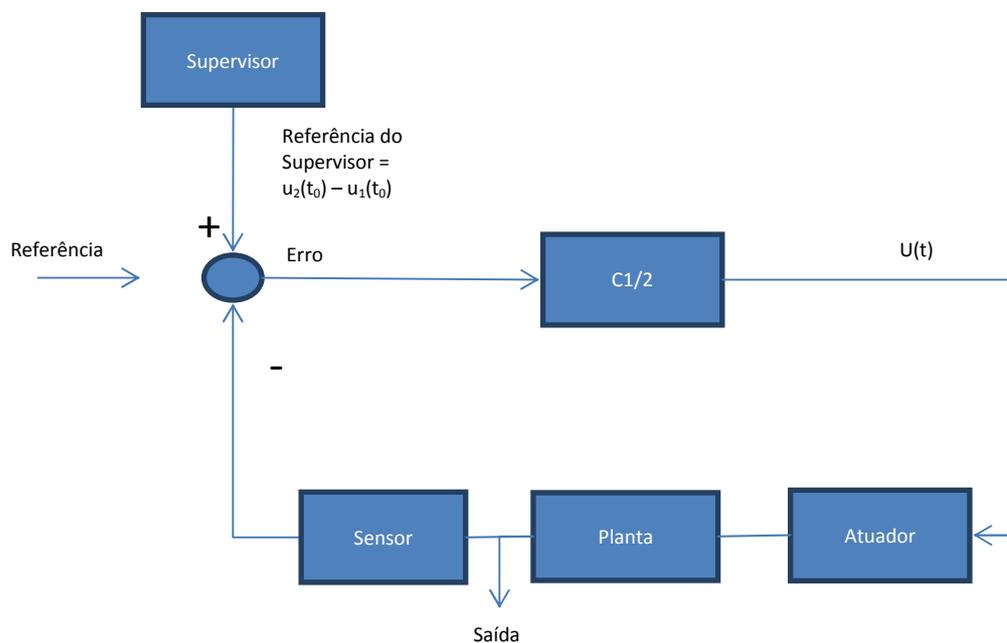


Figura 4-4: Diagrama em blocos do sistema durante a etapa de transição entre modos. A referência original deixa de ser usada, e uma nova referência do supervisor é introduzida.

Durante essa etapa, a planta não está mais seguindo a referência original, e sim sendo guiada de uma forma que a saída de C1/2 alcance o valor registrado de $u_2(t)$ da melhor forma possível.

Nos casos desta tese, o bloco de controle C1/2 foi composto por um dos blocos de controle originais, anexado a um novo bloco que modifique a propriedade do conjunto.

4.2.1. Exclusão do Atuador, Planta e Sensor

Há a possibilidade de realimentar o sinal do controlador C1/2 diretamente, durante a etapa de transição. Isso apresenta a vantagem de reduzir a ordem do problema tratado, e permite uma transição mais rápida, sem incluir a inércia da planta. Além disso, com a realimentação unitária vinda do controlador, o sinal de erro no sistema será justamente o quanto falta para o sinal de controle chegar no valor registrado de $u_2(t)$. Nesse caso particular, é possível aplicar a técnica de critérios integrais na sua forma original.

A desvantagem é manter a malha de controle aberta durante o intervalo de transição.

A Figura 4-5 mostra o sistema durante a fase imediatamente anterior à transição, quando o bloco supervisor mede as saídas dos blocos C1 e C2. A Figura 4-6 mostra o sistema durante a etapa de transição. Nela, entra em uso o controlador C1/2, que um bloco composto: há o bloco de controle inicial C_1 , que será alimentado por um bloco integrador multiplicado por um parâmetro ζ . O integrador foi escolhido pela simplicidade e por garantir que o sinal de controle só se torne estacionário quando o sinal de erro for nulo.

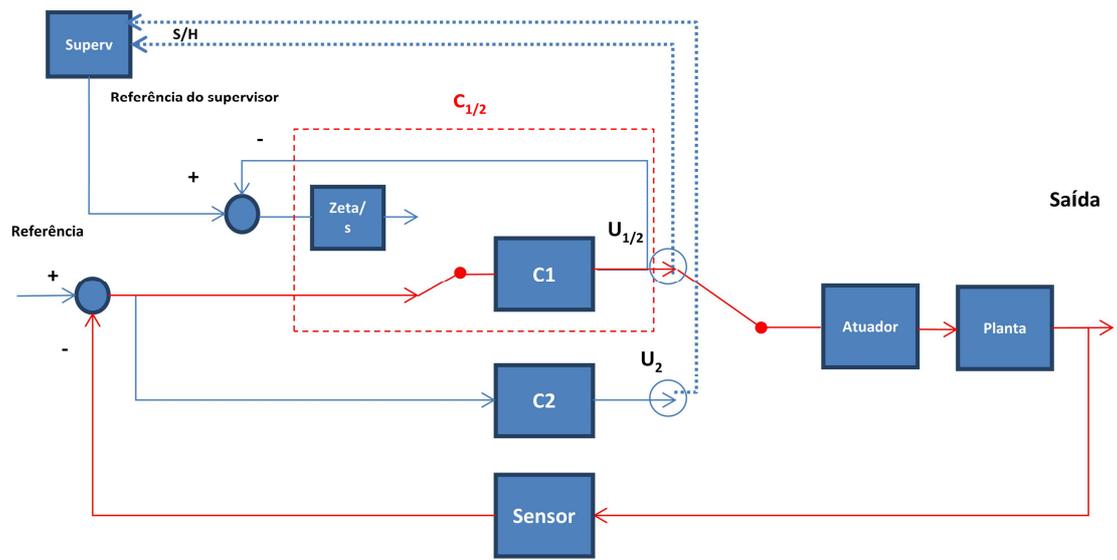


Figura 4-5: Diagrama em blocos do controle usado imediatamente antes da etapa de transição, com o bloco supervisor medindo as saídas dos blocos C1 e C1.

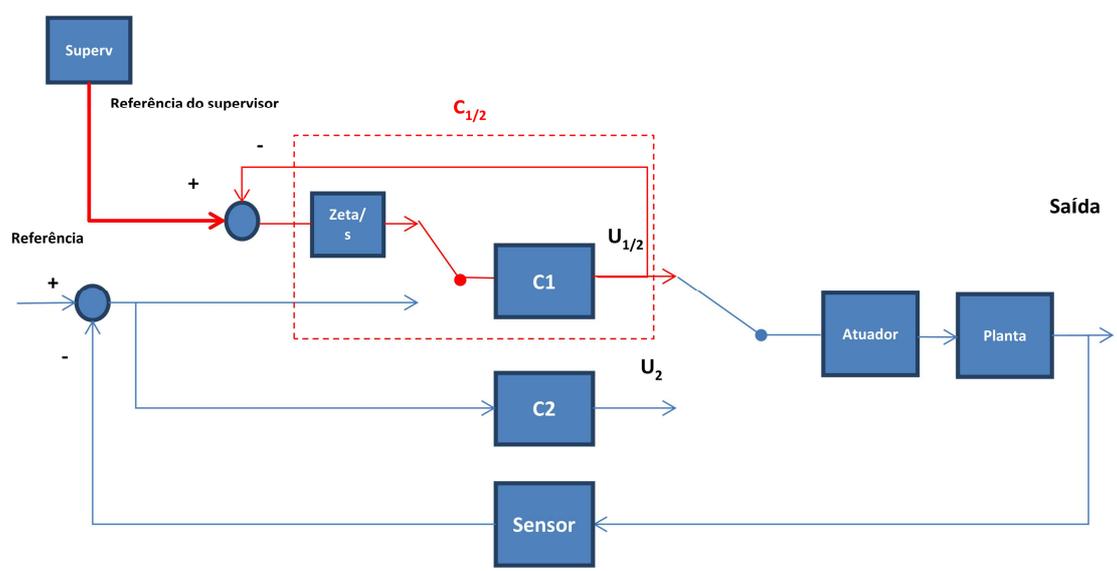


Figura 4-6: Diagrama em blocos do controle usado durante a etapa de transição, com a saída do controlador C1/2 sendo usada pela planta, mas guiada apenas pela referência do supervisor.

A referência será um degrau que faça a saída do Controlador 1 igualar-se com a saída do Controlador 2 no regime estacionário. O parâmetro ζ deve ser escolhido de forma que essa transição minimize um critério integral.

Após o sinal de controle durante a transição chegar suficientemente próximo do sinal de saída do Controlador 2, ocorre o chaveamento para este último (Figura 4-7).

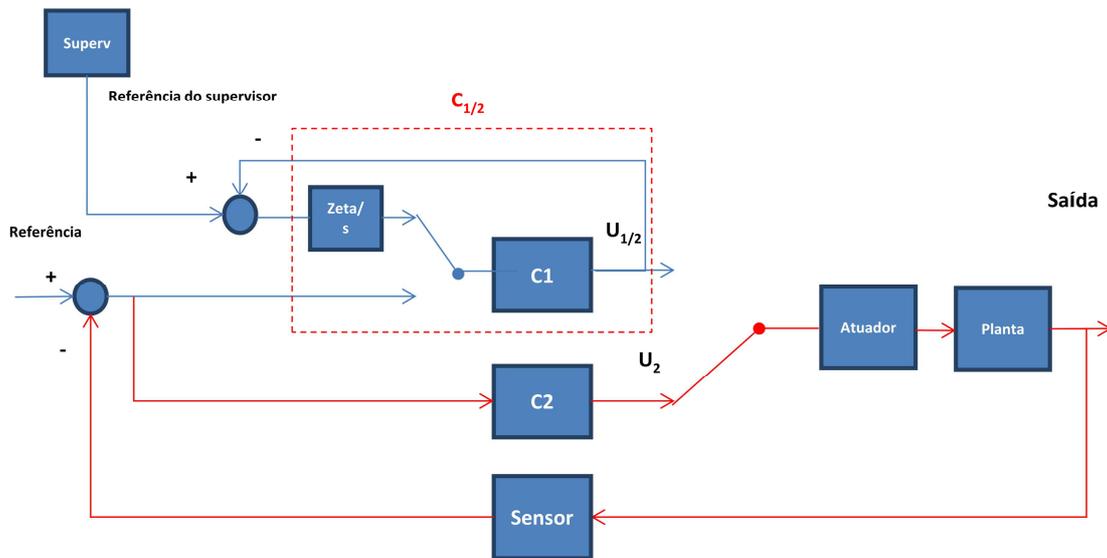


Figura 4-7: Diagrama em blocos do controle usado após a etapa de transição, com o controlador C2 sendo usado como um bloco de controle comum.

4.2.2. Critério Integral do Quadrado do Erro (ISE)

O sistema composto pelo integrador e PID tem a seguinte função de transferência $H(s)$:

$$H(s) = \frac{\zeta}{s} \left(k_p + s k_d + \frac{k_i}{s} \right) = \zeta \frac{s k_p + s^2 k_d + k_i}{s^2} \quad (4.6)$$

Como buscamos o sinal de erro $e(t)$, tratamos $H(s)$ como se fosse um bloco de realimentação, e chegamos à função transferência entre a referência $R(s)$ e o erro $E(s)$:

$$E(s) = R(s) \frac{1}{1+H(s)} \quad (4.7)$$

Em malha fechada, com realimentação unitária, temos:

$$E(s) = R(s) \frac{s^2}{s^2 + \zeta(sk_p + s^2k_d + k_i)} \quad (4.8)$$

O sinal de referência será um degrau de parâmetro D , da forma:

$$R(s) = \frac{D}{s} \quad (4.9)$$

Portanto, a resposta do erro ao degrau no domínio da frequência fica:

$$E(s) = D \frac{s}{s^2(k_d + \zeta) + s\zeta k_p + \zeta k_i} \quad (4.10)$$

Desde já fica visível que a amplitude do degrau não influencia o valor de ζ que minimizará o funcional. MOSTRE O FUNCIONAL

Devemos agora encontrar a resposta do erro no domínio do tempo. Para isso, encontraremos os polos λ_1 e λ_2 do polinômio divisor:

$$\lambda = \frac{-\zeta k_p \pm \sqrt{(\zeta k_p)^2 - 4(k_d + \zeta)\zeta k_i}}{2(k_d + \zeta)} \quad (4.11)$$

Polos reais são garantidos forçando:

$$(\zeta k_p)^2 - 4(k_d + \zeta)\zeta k_i > 0 \quad (4.12)$$

Simplificando a inequação de segundo grau resultante, temos:

$$\zeta > \frac{4k_d k_i}{k_p^2 - 4k_i} \quad (4.13)$$

Retornando à resposta do erro $E(s)$, e substituindo os polos λ_1 e λ_2 por simplicidade, aplicamos o método das frações parciais:

$$E(s) = D \left(\frac{A}{s-\lambda_1} + \frac{B}{s-\lambda_2} \right) \quad (4.14)$$

$$= D \frac{(A+B)s + (-A\lambda_2 - B\lambda_1)}{(s-\lambda_1)(s-\lambda_2)} \quad (4.15)$$

Sabemos que o coeficiente em s deve ser unitário, e o coeficiente independente de s deve ser nulo, portanto:

$$A + B = 1 \therefore B = 1 - A \quad (4.16)$$

$$-A\lambda_2 - B\lambda_1 = 0 \therefore A = -\frac{B\lambda_1}{\lambda_2} \quad (4.17)$$

Eliminando B , temos:

$$A = -\frac{B\lambda_1}{\lambda_2} = \frac{(A-1)\lambda_1}{\lambda_2} = \frac{\lambda_1}{\lambda_1 - \lambda_2} \quad (4.18)$$

E encontramos o coeficiente seguinte:

$$B = 1 - \frac{\lambda_1}{\lambda_1 - \lambda_2} = \frac{\lambda_2}{\lambda_1 - \lambda_2} \quad (4.19)$$

(Deve ser lembrado que esta forma é inaplicável quando os polos forem idênticos).

Tendo definido polos λ_1 e λ_2 , e os coeficientes A e B das frações parciais, prosseguimos com a resposta do erro ao degrau, que está na forma:

$$E(s) = D \left(\frac{A}{s-\lambda_1} + \frac{B}{s-\lambda_2} \right) \quad (4.20)$$

Aplicando a Transformada Inversa de Laplace, obtemos a resposta do erro ao degrau no domínio do tempo:

$$\mathcal{L}^{-1}\{E(s)\} = e(t) = D(Ae^{\lambda_1 t} + Be^{\lambda_2 t}) \quad (4.21)$$

Seguindo de acordo com o critério integral escolhido, elevamos a resposta do erro no tempo ao quadrado:

$$\varepsilon^2(t) = D^2(A^2 e^{2\lambda_1 t} + 2AB e^{(\lambda_1 + \lambda_2)t} + B^2 e^{2\lambda_2 t}) \quad (4.22)$$

Como o objetivo é encontrar

$$J = \int_0^\infty \varepsilon^2(t) dt \quad (4.23)$$

Podemos utilizar o Teorema da Integração Real, que diz que:

$$\int_0^\infty f(t) dt = \lim_{s \rightarrow 0} F(s) \quad (4.24)$$

Assim, basta aplicarmos a Transformada de Laplace ao quadrado da função, e anular o valor de s .

$$\mathcal{L}\{\varepsilon^2(t)\} = D^2 \left(A^2 \frac{1}{s-2\lambda_1} + 2AB \frac{1}{s-(\lambda_1+\lambda_2)} + B^2 \frac{1}{s-2\lambda_2} \right) \quad (4.25)$$

$$\lim_{s \rightarrow 0} \mathcal{L}\{\varepsilon^2(t)\} = D^2 \left(-\frac{A^2}{2\lambda_1} - \frac{2AB}{\lambda_1+\lambda_2} - \frac{B^2}{2\lambda_2} \right) \quad (4.26)$$

Substituindo os coeficientes e os polos, devemos substituí-los de volta no funcional:

$$J = D^2 \left(-\frac{A^2}{2\lambda_1} - \frac{2AB}{\lambda_1+\lambda_2} - \frac{B^2}{2\lambda_2} \right) \quad (4.27)$$

onde

$$A = \frac{\lambda_1}{\lambda_1 - \lambda_2} \quad (4.28)$$

$$B = \frac{\lambda_2}{\lambda_1 - \lambda_2} \quad (4.29)$$

e

$$\lambda_1 = \frac{-\zeta k_p + \sqrt{(\zeta k_p)^2 - 4(k_d + \zeta)\zeta k_i}}{2(k_d + \zeta)} \quad (4.30)$$

$$\lambda_2 = \frac{-\zeta k_p - \sqrt{(\zeta k_p)^2 - 4(k_d + \zeta)\zeta k_i}}{2(k_d + \zeta)} \quad (4.31)$$

Após a substituição, o funcional J do critério ISE fica:

$$J_{ISE}(\zeta) = \frac{(\zeta k_p^2 + 4k_i k_d + 4\zeta k_i)(k_d + \zeta)}{2k_p(k_p^2 - 4k_i)\zeta^2 - 8k_p k_i k_d \zeta} \quad (4.32)$$

A derivada de J em relação a ζ fica na forma:

$$\frac{d}{d\zeta} J_{ISE}(\zeta) = - \frac{k_d \left((8k_p^2 k_i + k_p^4 - 16k_i^2)\zeta^2 + (8k_p^2 k_d k_i - 32k_i^2 k_d)\zeta + (-16k_i^2 k_d^2) \right)}{2 \left((k_p^2 - 4k_i)\zeta - 4k_i k_d \right)^2 k_p \zeta^2} \quad (4.33)$$

O polinômio no dividendo é do segundo grau, e seus zeros podem ser encontrados fazendo:

$$\zeta_{critico} = \frac{(32k_i^2 k_d - 8k_p^2 k_d k_i) \pm \sqrt{(8k_p^2 k_d k_i - 32k_i^2 k_d)^2 - 4(8k_p^2 k_i + k_p^4 - 16k_i^2)(-16k_i^2 k_d^2)}}{2(8k_p^2 k_i + k_p^4 - 16k_i^2)} \quad (4.34)$$

Para que sejam reais, é preciso que:

$$(8k_p^2 k_d k_i - 32k_i^2 k_d)^2 + 4(8k_p^2 k_i + k_p^4 - 16k_i^2)(16k_i^2 k_d^2) \geq 0 \quad (4.35)$$

A partir destas informações, é possível buscar por pontos de mínimo em J para ζ positivo, ou determinar se eles não existem. Caso não existam mínimos para ζ positivo, a relação entre ζ e J ao menos permite atribuir uma qualidade ao ζ baseada na trajetória de transição que ele causará no sinal de controle. Portanto, é possível escolher um ζ que produza um J considerado aceitável.

4.2.3. Critério Integral do Tempo Vezes o Quadrado do Erro (ITSE)

Aproveitando informações obtidas na demonstração para o critério anterior, já temos calculado previamente o valor do quadrado do erro:

$$\varepsilon^2(t) = D^2(A^2 e^{2\lambda_1 t} + 2AB e^{(\lambda_1 + \lambda_2)t} + B^2 e^{2\lambda_2 t}) \quad (4.36)$$

Seguindo a definição do critério de tempo vezes o quadrado do erro (ITSE), o objetivo agora é encontrar:

$$J = \int_0^{\infty} t \varepsilon^2(t) dt \quad (4.37)$$

Podemos utilizar o Teorema da Integração Real, que diz que:

$$\int_0^{\infty} f(t) dt = \lim_{s \rightarrow 0} F(s) \quad (4.38)$$

Assim, basta aplicarmos a Transformada de Laplace no tempo vezes o quadrado da função, e fazer S tender a zero. Como foi calculado anteriormente, já sabemos que:

$$\mathcal{L}\{\varepsilon^2(t)\} = D^2 \left(A^2 \frac{1}{s-2\lambda_1} + 2AB \frac{1}{s-(\lambda_1+\lambda_2)} + B^2 \frac{1}{s-2\lambda_2} \right) \quad (4.39)$$

Aplicando as propriedades da Transformada de Laplace, sabemos também que:

$$\mathcal{L}\{t \varepsilon^2(t)\} = -\frac{d}{ds} \mathcal{L}\{\varepsilon^2(t)\} \quad (4.40)$$

Assim, calcula-se a Transformada de Laplace do tempo vezes o quadrado do erro:

$$-\frac{d}{ds}\mathcal{L}\{e^2(t)\} = D^2 \left\{ A^2 \frac{1}{(s-2\lambda_1)^2} + 2AB \frac{1}{[s-(\lambda_1+\lambda_2)]^2} + B^2 \frac{1}{(s-2\lambda_2)^2} \right\} \quad (4.41)$$

Fazendo a variável S tender a zero, temos então:

$$J = \int_0^\infty t\varepsilon^2(t)dt = \lim_{s \rightarrow 0} \mathcal{L}\{te^2(t)\} = D^2 \left[A^2 \frac{1}{4(\lambda_1)^2} + 2AB \frac{1}{(\lambda_1+\lambda_2)^2} + B^2 \frac{1}{4(\lambda_2)^2} \right] \quad (3.49)$$

Substituímos agora os valores previamente calculados $A = \frac{\lambda_1}{\lambda_1-\lambda_2}$ e $B = \frac{\lambda_2}{\lambda_1-\lambda_2}$, e

$$\lambda_1 = \frac{-\zeta k_p + \sqrt{(\zeta k_p)^2 - 4(k_d + \zeta)\zeta k_i}}{2(k_d + \zeta)} \quad (4.42)$$

$$\lambda_2 = \frac{-\zeta k_p - \sqrt{(\zeta k_p)^2 - 4(k_d + \zeta)\zeta k_i}}{2(k_d + \zeta)} \quad (4.43)$$

A resposta fica:

$$J_{ITSE}(\zeta) = \frac{[(k_p^2 + 4k_i)\zeta + 4k_i k_d](k_d + \zeta)^2}{2k_p^2 \zeta^2 [(k_p^2 - 4k_i)\zeta - 4k_i k_d]} \quad (4.44)$$

E a derivada correspondente em ζ fica:

$$\frac{d}{d\zeta} J_{ITSE}(\zeta) = \frac{k_d(k_d + \zeta)[(16k_i^2 - 4k_i k_p^2 - k_p^4)\zeta^2 + (32k_d k_i^2 - 4k_d k_i k_p^2)\zeta + 16k_d^2 k_i^2]}{k_p^2 \zeta^3 [(4k_i - k_p^2)\zeta + 4k_d k_i]^2} \quad (4.45)$$

Assim como no critério anterior, o termo D^2 foi deixado no lado direito da equação por não ser relevante na busca de zeros.

4.2.4. Critérios Integral do Módulo do Erro (IME)

Sabemos previamente que a função do erro no domínio da frequência é da forma:

$$E(s) = D \left(\frac{A}{s-\lambda_1} + \frac{B}{s-\lambda_2} \right) \quad (4.46)$$

O tratamento algébrico de funções com módulo é complexo demais para o escopo desta tese. No entanto, se garantirmos que as raízes λ_1 e λ_2 são reais (sistema superamortecido), podemos determinar seu comportamento para esse caso particular. O sinal de erro será superamortecido para quando:

$$\zeta > \frac{4k_i k_d}{k_p^2 - 4k_i} \quad (4.47)$$

Seguindo a definição do critério da Integral do Tempo Vezes o Quadrado do Erro (IME), o objetivo é encontrar:

$$J_{IME} = \int_0^{\infty} |\varepsilon(t)| dt \quad (4.48)$$

Podemos utilizar o Teorema da Integração Real, que diz que:

$$\int_0^{\infty} f(t) dt = \lim_{s \rightarrow 0} F(s) \quad (4.49)$$

Assim, fazemos a variável s tender a zero, obtendo:

$$\int_0^{\infty} f(t) dt = -D \left(\frac{A}{\lambda_1} + \frac{B}{\lambda_2} \right) \quad (4.50)$$

Já foi demonstrado que o valor de D não influencia a localização dos pontos de mínimo. Por simplicidade, era considerado $D = 1$. No entanto, este é um caso particular onde a função de módulo foi ignorada. É preciso que o sinal de erro

seja mantido positivo para que o resultado coincida com o da definição. Assim, de acordo com a Equação 4-9, faremos com que $D = -1$.

Substituindo os valores A e B,

$$A = \frac{\lambda_1}{\lambda_1 - \lambda_2} \quad (4.51)$$

$$B = \frac{\lambda_2}{\lambda_1 - \lambda_2} \quad (4.52)$$

$$\lambda_1 = \frac{-\zeta k_p + \sqrt{(\zeta k_p)^2 - 4(k_d + \zeta)\zeta k_i}}{2(k_d + \zeta)} \quad (4.53)$$

$$\lambda_2 = \frac{-\zeta k_p - \sqrt{(\zeta k_p)^2 - 4(k_d + \zeta)\zeta k_i}}{2(k_d + \zeta)} \quad (4.54)$$

Temos o resultado:

$$J_{IME}(\zeta) = \frac{2}{(k_d + \zeta)\sqrt{k_p^2 \zeta^2 - 4k_i \zeta(k_d + \zeta)}} \quad (4.55)$$

para $\zeta > \frac{4k_i k_d}{k_p^2 - 4k_i}$.

A derivada correspondente fica:

$$\frac{d}{d\zeta} J_{IME}(\zeta) = \frac{2[(8k_i - 2k_p^2)\zeta^2 + (10k_i k_d - k_d k_p^2)\zeta + 2k_i k_d^2]}{(k_d + \zeta)^2 (k_p^2 \zeta^2 - k_i \zeta(4k_d + 4\zeta))^{\frac{3}{2}}} \quad (4.56)$$

para $\zeta > \frac{4k_i k_d}{k_p^2 - 4k_i}$.

4.2.5. Critérios Integral do Tempo Vezes o Módulo do Erro (ITME)

Como também possui a função módulo em sua definição, o critério ITME será definido apenas para quando $\zeta > \frac{4k_i k_d}{k_p^2 - 4k_i}$, onde $|\varepsilon(t)| = \varepsilon(t)$.

Partimos da função do erro no domínio da frequência:

$$E(s) = D \left(\frac{A}{s-\lambda_1} + \frac{B}{s-\lambda_2} \right) \quad (4.57)$$

Aplicando as propriedades da Transformada de Laplace, sabemos também que:

$$\mathcal{L}\{t\varepsilon(t)\} = -\frac{d}{ds}\mathcal{L}\{\varepsilon(t)\} \quad (3.58)$$

Derivamos $E(s)$ em s para encontrar:

$$\frac{d}{ds}\mathcal{L}\{\varepsilon(t)\} = -D \left(\frac{A}{(s-\lambda_1)^2} + \frac{B}{(s-\lambda_2)^2} \right) \quad (4.59)$$

Utilizando o Teorema da Integração Real, temos que:

$$\int_0^{\infty} t\varepsilon(t)dt = \lim_{s \rightarrow 0} D \left[\frac{A}{(s-\lambda_1)^2} + \frac{B}{(s-\lambda_2)^2} \right] \quad (4.60)$$

Portanto, o valor de J encontrado é:

$$J_{ITME}(\zeta) = D \left(\frac{A}{\lambda_1^2} + \frac{B}{\lambda_2^2} \right) \quad (4.61)$$

Assim como já foi mencionado na sessão do critério (IME), para que o sinal de erro seja mantido positivo, o sinal D do degrau será considerado negativo $D = -1$.

Após a substituição de A , B , λ_1 e λ_2 , temos:

$$J_{ITME}(\zeta) = \frac{4k_p}{4k_i(k_d + \zeta)^2 \sqrt{(k_p^2 - k_i^2)\zeta^2 - 4k_d k_i \zeta}} \quad (4.62)$$

para $\zeta > \frac{4k_i k_d}{k_p^2 - 4k_i}$

E a derivada em ζ correspondente é

$$\frac{d}{d\zeta} J_{ITME}(\zeta) = \frac{k_p [(16k_i - 4k_p^2)\zeta^2 + (18k_i k_d - k_d k_p^2)\zeta + 2k_i k_d^2]}{k_i(k_d + \zeta)^4 [(k_p^2 - 4k_i)\zeta^2 - 4k_d k_i \zeta]^{\frac{3}{2}}} \quad (4.63)$$

para $\zeta > \frac{4k_i k_d}{k_p^2 - 4k_i}$

4.2.6. Estabilidade Geral Durante Transições.

Para analisar a estabilidade geral do sistema durante uma transição, assumimos que foi usada a técnica de “cross-fading”, já descrita no Capítulo 3 e mostrada no modelo da Figura 4-8.

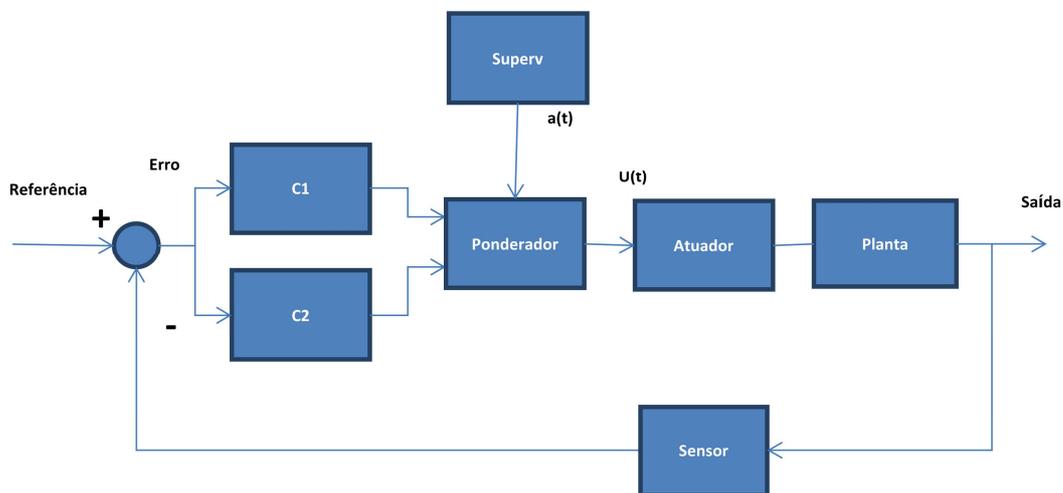


Figura 4-8: Modelo de um sistema que pode realizar uma transição em “cross-fading” entre os controladores C1 e C2.

Sejam $c_1(t)$ e $c_2(t)$ as respostas ao impulso unitário dos controladores inicial e final, respectivamente. O objetivo é encontrar a resposta ao impulso do sistema composto por esses dois blocos chaveados dessa forma, e assim determinar o impacto que isso tem nos polos e zeros do sistema completo em malha fechada.

Uma forma óbvia de modelar o “cross-fading” de duração Δt seria com uma função composta por rampas e degraus (onde $h(t)$ representa a função degrau):

$$a(t) = h(t) \frac{t}{\Delta t} - h(t - \Delta t) \frac{t}{\Delta t} \quad (4.64)$$

Esta função produz a forma vista na Figura 4-9.

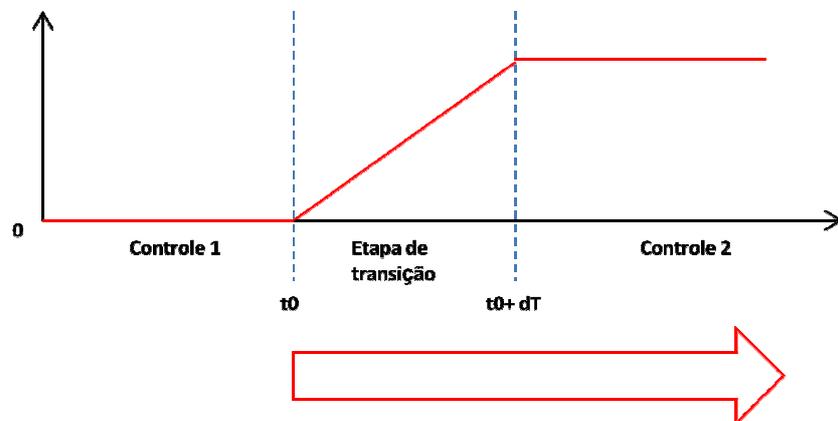


Figura 4-9: Gráfico da função $a(t)$ para uma forma de “cross-fading” linear.

Para obter a Função Transferência do bloco de controle composto por “cross-fading” com esta função, é preciso fazer a **convolução no domínio da frequência** das Funções Transferência de $c_1(s)$ e $c_2(s)$ com as Transformadas de Laplace de $[1 - a(t)]$ e $a(t)$, respectivamente.

$$C_{composto}(s) = C_1(s) * \mathcal{L}\{1 - a(t)\} + C_2(s) * \mathcal{L}\{a(t)\} \quad (4.65)$$

No entanto, o segundo termo da soma não pode ser resolvido facilmente, sem antes especificar a função C_2 . Como alternativa, apresentamos abaixo duas formas originais de modelar o “cross-fading”, de forma que uma expressão geral para o bloco de controle equivalente possa ser obtida.

4.2.6.1. “Cross-Fading” Linear Ilimitado.

Aqui é usada uma forma de simplificação. O sinal $a(t)$ que rege o “cross-fading” é uma função de primeiro grau, que descreve uma linha reta que vai de 0 a 1 num intervalo de tempo Δt (Figura 4-10).

$$a(t) = \frac{t}{\Delta t} \quad (4.66)$$

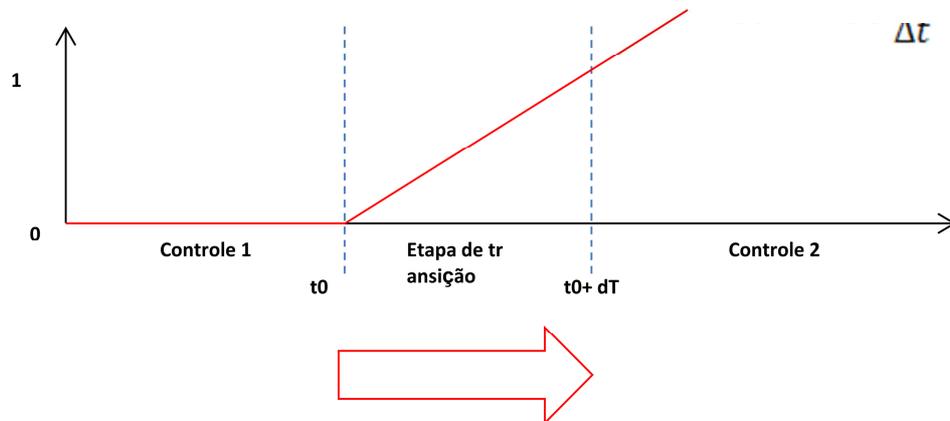


Figura 4-10: Função $a(t)$ para o “cross-fading” linear ilimitado, onde não há a modelagem do fim da transição.

Sejam $c_1(t)$ e $c_2(t)$ as respostas ao impulso dos Controladores C1 e C2, respectivamente. Se considerarmos ambos os blocos de controle sob

chaveamento em “cross-fading” como um só bloco, sua resposta ao impulso será dada por:

$$c(t) = c_1(t) \left[h(t) \left(1 - \frac{t}{k} \right) \right] + c_2(t) \left[h(t) \frac{t}{k} \right] \quad (4.67)$$

Usamos as propriedades conhecidas da Transformada de Laplace para encontrar a expressão desse bloco completo no domínio da frequência:

$$\mathcal{L}\{h(t)f(t)\} = F(s) \quad (4.78)$$

$$\mathcal{L}\left\{\frac{1}{k}h(t)tf(t)\right\} = \frac{1}{k} \frac{d}{ds} F(s) \quad (4.79)$$

A expressão completa fica na forma:

$$C(s) = C_1(s) - \frac{1}{k} \frac{d}{ds} C_1(s) + \frac{1}{k} \frac{d}{ds} C_2(s) \quad (4.80)$$

Essa função que rege o “cross-fading” não está modelada para considerar o final da transição. Isso exigiria resolver Transformadas de Laplace com a função degrau sob um deslocamento de tempo diferente do resto da função. Não foi possível encontrar formas de resolver isso sem antes especificar a Função Transferência dos blocos de controle.

Não incluir o término da etapa de transição no modelo faz com que ambos os sinais de controle continuem sendo multiplicados indefinidamente por sinais de módulo crescente. De uma forma geral, um sistema composto dessa forma será instável.

No entanto, como a descrição desse sistema é usada por um intervalo limitado e conhecido de tempo, Δt segundos, pode-se usar isso para desconsiderar os polos positivos de crescimento lento, baseando-se num nível de tolerância arbitrário. Pode-se estabelecer o critério de que nenhum modo instável com polo de parte real positiva λ cresça mais do que x% durante o intervalo de tempo de transição, que ocorre quando $t/k = 1$.

$$e^{\lambda k} < 1 + \frac{x}{100} \quad (4.81)$$

Resolvendo a inequação para λ , temos:

$$\lambda < \frac{1}{k} [\ln(100 + x) - \ln 100] \quad (4.82)$$

Por exemplo, se uma transição dura 0,1 segundo, e a exigência é que nenhum modo instável cresça mais do que 10%, é preciso considerar os polos de parte real maior do que 0,9531.

4.2.6.2. “Cross-Fading” em Decaimento Exponencial

O método anterior utilizou uma aproximação no modelo; o método apresentado aqui é matematicamente preciso. Assume-se que a função $a(t)$ que rege o “cross-fading” é similar a um decaimento exponencial sob uma constante de tempo k (Equação 4.83). Sem perda de generalidade, assumimos que a etapa de transição começou no tempo $t = 0$.

$$a(t) = 1 - e^{-tk} \quad (4.83)$$

Rigorosamente, um decaimento exponencial nunca se anula por completo. No entanto, consideraremos que a transição estará terminada após cinco constantes de tempo k (Figura 4-11).

$$\Delta t = 5k \quad (4.84)$$

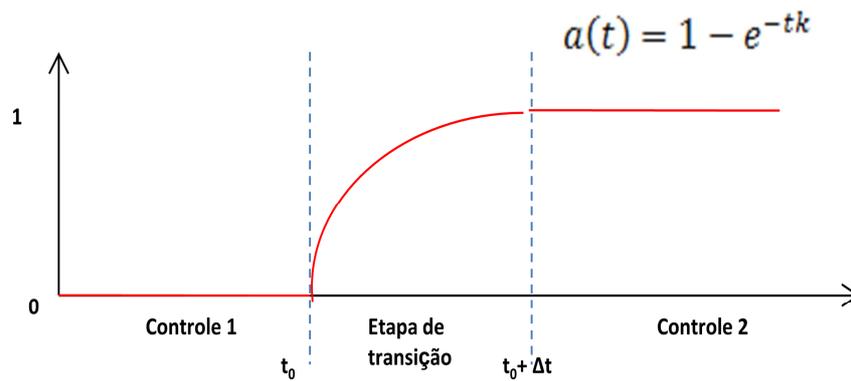


Figura 4-11: Função $a(t)$ para o “cross-fading” em decaimento exponencial.

Seja $c(t)$ a resposta ao impulso vista por um bloco equivalente aos dois controladores chaveados por “cross-fading”. Ele ficará sob a forma:

$$c(t) = c_1(t)e^{-tk} + c_2(t)(1 - e^{-tk}) \quad (4.85)$$

Aproveitando as propriedades tabeladas da Transformada de Laplace, podemos encontrar a resposta ao impulso do controlador composto no domínio da frequência, $C(s)$:

$$\mathcal{L}\{c_1(t)e^{-tk}\} = C_1(s + k) \quad (4.86)$$

$$\mathcal{L}\{c_2(t)(1 - e^{-tk})\} = C_2(s) - C_2(s + k) \quad (4.87)$$

Portanto, o bloco de controle composto em função da constante de tempo k fica na forma:

$$C(s) = C_1(s + k) + C_2(s) - C_2(s + k) \quad (4.88)$$

Se as Funções Transferência dos blocos controladores são conhecidas, torna-se trivial compor a Função Transferência do controlador composto. A função exponencial pode não ser a forma mais conveniente de executar um “cross-

fading”; no entanto, se ela for usada, o resultado da análise não dependerá de aproximações.

Se os polos e zeros da análise indicarem um sistema estável, pode-se assumir que o sistema continuará estável para uma função de “cross-fading” $a(t)$ menos brusca do que a do decaimento exponencial.

4.2.6.3. Aplicação ao Caso da PMM

Aqui aplicaremos uma transição entre dois PIDs, cuja forma geral é:

$$C_n(s) = k_{pn} + \frac{k_{in}}{s} + k_{dn}s \quad (4.89)$$

O controlador final ($n = 2$) é idêntico ao calculado para o modelo da PMM ($k_p = 40$, $k_i = 1$, $k_d = 400$), e o controlador inicial ($n = 1$) é o mesmo controle com a componente integral cinco vezes maior ($k_p = 40$, $k_i = 5$, $k_d = 400$). As constantes dos controladores PIDs são adimensionais.

Considerando o modelo de “cross-fading” linear, temos:

$$C_{lin}(s) = C_1(s) - \frac{1}{k} \frac{d}{ds} C_1(s) + \frac{1}{k} \frac{d}{ds} C_2(s) \quad (4.90)$$

Realizando a substituição e simplificando, o bloco de controle equivalente fica:

$$C_{lin}(s) = k_{p1} + \frac{k_{i1}}{s} + k_{d1}s + \frac{1}{k} \left(\frac{k_{i1}}{s^2} - k_{d1} \right) + \frac{1}{k} \left(k_{d2} - \frac{k_{i2}}{s^2} \right) \quad (4.91)$$

Da mesma forma, considerando o modelo do “cross-fading” em decaimento exponencial, temos:

$$C_{exp}(s) = C_1(s+k) + C_2(s) - C_2(s+k) \quad (4.92)$$

Realizando a substituição e simplificando, o bloco de controle equivalente fica:

$$C_{exp}(s) = k_{p1} + \frac{k_{i1} - k_{i2}}{s+k} + k_{d1}s + (k_{d1} - k_{d2})k + \frac{k_{i2}}{s} \quad (4.93)$$

4.2.6.4. Transição entre dois PIDs na Planta da PMM.

Considerando o modelo da PMM movida pelas rodas de reação, temos o sistema abaixo, onde $T(s)$ é a Função Transferência, $C_1(s)$ é o controlador PID inicial ($n = 1$), $C_2(s)$ é o controlador PID final ($n = 2$), $A(s)$ é o atuador (roda de reação), $G(s)$ é a planta da PMM e $S(s)$ é o sensor:

$$T(s) = \frac{C(s)A(s)G(s)}{1+C(s)A(s)G(s)} \quad (4.94)$$

$$C_1(s) = k_{p1} + \frac{k_{i1}}{s} + k_{d1}s \quad (4.95)$$

$$C_2(s) = k_{p2} + \frac{k_{i2}}{s} + k_{d2}s \quad (4.96)$$

$$A(s) = \frac{k_v s}{s + \frac{1}{T_v}} \quad (4.97)$$

$$G(s) = \frac{1}{I s^2} \quad (4.98)$$

$$S(s) = 1 \quad (4.99)$$

Onde,

$$T_v = 20 \text{ s}$$

$$K_v = 0,06 \text{ Nm/V}$$

$$I_x = 295,97401 \text{ kg} \cdot \text{m}^2$$

Para determinar o comportamento do sistema durante o “cross-fading”, foi usado um controlador equivalente, composto a partir dos controladores C1 e C2. Segundo o modelo de “cross-fading” linear, ele é dado pela Equação 4.91; segundo o modelo de “cross-fading” exponencial, ele é dado pela Equação 4.93.

A Função Transferência no modelo linear fica na forma:

$$T(s) = \frac{\left[k_{p1} + \frac{k_{i1}}{s} + k_{d1}s + \frac{1}{k} \left(\frac{k_{i1}}{s^2} - k_{d1} \right) + \frac{1}{k} \left(k_{d2} - \frac{k_{i2}}{s^2} \right) \right] \frac{k_v s}{s + \frac{1}{T_v}} \frac{1}{IS^2}}{1 + \left[k_{p1} + \frac{k_{i1}}{s} + k_{d1}s + \frac{1}{k} \left(\frac{k_{i1}}{s^2} - k_{d1} \right) + \frac{1}{k} \left(k_{d2} - \frac{k_{i2}}{s^2} \right) \right] \frac{k_v s}{s + \frac{1}{T_v}} \frac{1}{IS^2}} \quad (4.100)$$

A Função Transferência no modelo exponencial fica na forma:

$$T(s) = \frac{\left[k_{p1} + \frac{k_{i1} - k_{i2}}{s+k} + k_{d1}s + (k_{d1} - k_{d2})k + \frac{k_{i2}}{s} \right] \frac{k_v s}{s + \frac{1}{T_v}} \frac{1}{IS^2}}{1 + \left[k_{p1} + \frac{k_{i1} - k_{i2}}{s+k} + k_{d1}s + (k_{d1} - k_{d2})k + \frac{k_{i2}}{s} \right] \frac{k_v s}{s + \frac{1}{T_v}} \frac{1}{IS^2}} \quad (4.101)$$

Os resultados desta análise são mostrados no Capítulo 6.

5 SIMULAÇÕES E ANÁLISES DA REDUÇÃO DOS EFEITOS DA RECONFIGURAÇÃO SOBRE OS TRANSITÓRIOS

Este Capítulo descreve as simulações e análises da redução dos efeitos da reconfiguração sobre os transitórios, realizadas para as configurações das Figuras 3-8 a 4-7, já apresentadas.

5.1. Uso de rastreo de saída para a transição de submodos dentro do modo de navegação fina da PMM.

Foi utilizada a forma de rastreo simplificada para a PMM, descrita na Sessão 3.2.1.1. Exemplos são mostrados em pares com e sem uma implementação para redução de impacto, para comparação. Há um chaveamento entre os controles inicial e final aos 30 segundos.

As Figuras 5-1 e 5-2 mostram o sinal de controle como ele é visto pela planta, com e sem a abordagem de rastreo de saída, respectivamente. Essa abordagem consiste em enviar um sinal para forçar o controlador C2 a aproximar-se da saída do controlador C1. A Figura 5-2 mostra a variação total entre as saídas, de 10 a -10 Volts, provável de introduzir problemas de “windup”. A versão com rastreo de saída mostra uma transição muito mais suave, dentro do envelope de 10 V.

As Figuras 5-3 e 5-4 mostram o módulo da velocidade angular, respectivas às Figuras 5-1 e 5-2. A implementação do rastreo de saída para redução do impacto foi capaz de eliminar as oscilações. A transição resulta apenas em uma redução na desaceleração aos 30 segundos.

As Figuras 5-5 e 5-6 mostram o erro de apontamento respectivos às Figuras 5-1 e 5-2. O impacto não é prontamente visível, mas causa um sobressinal maior. A implementação para redução de impacto obtém um apontamento mais rápido.

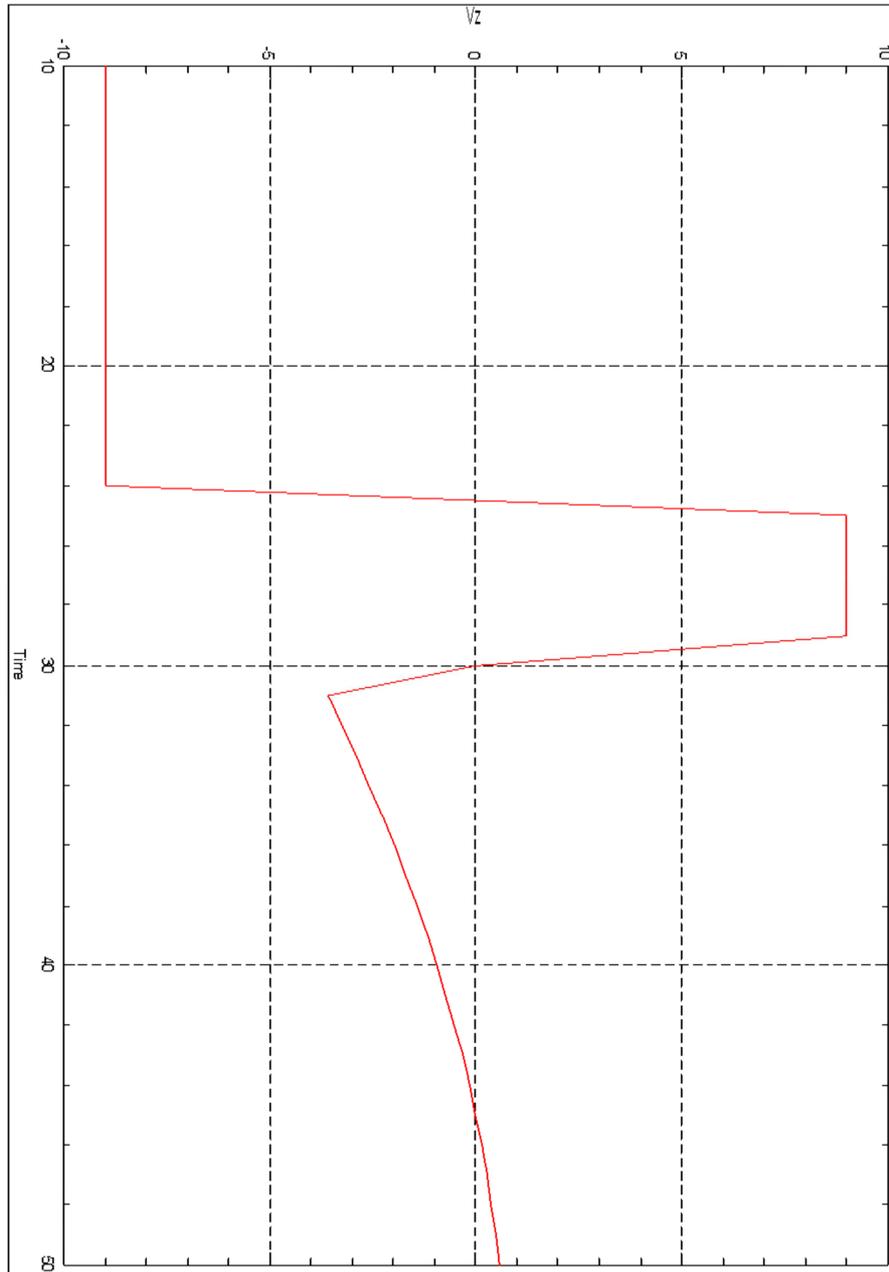


Figura 5-1: Sinal de Controle com rastreo de saída.

Figura 5-2: Sinal de Controle sem rastreo de saída.

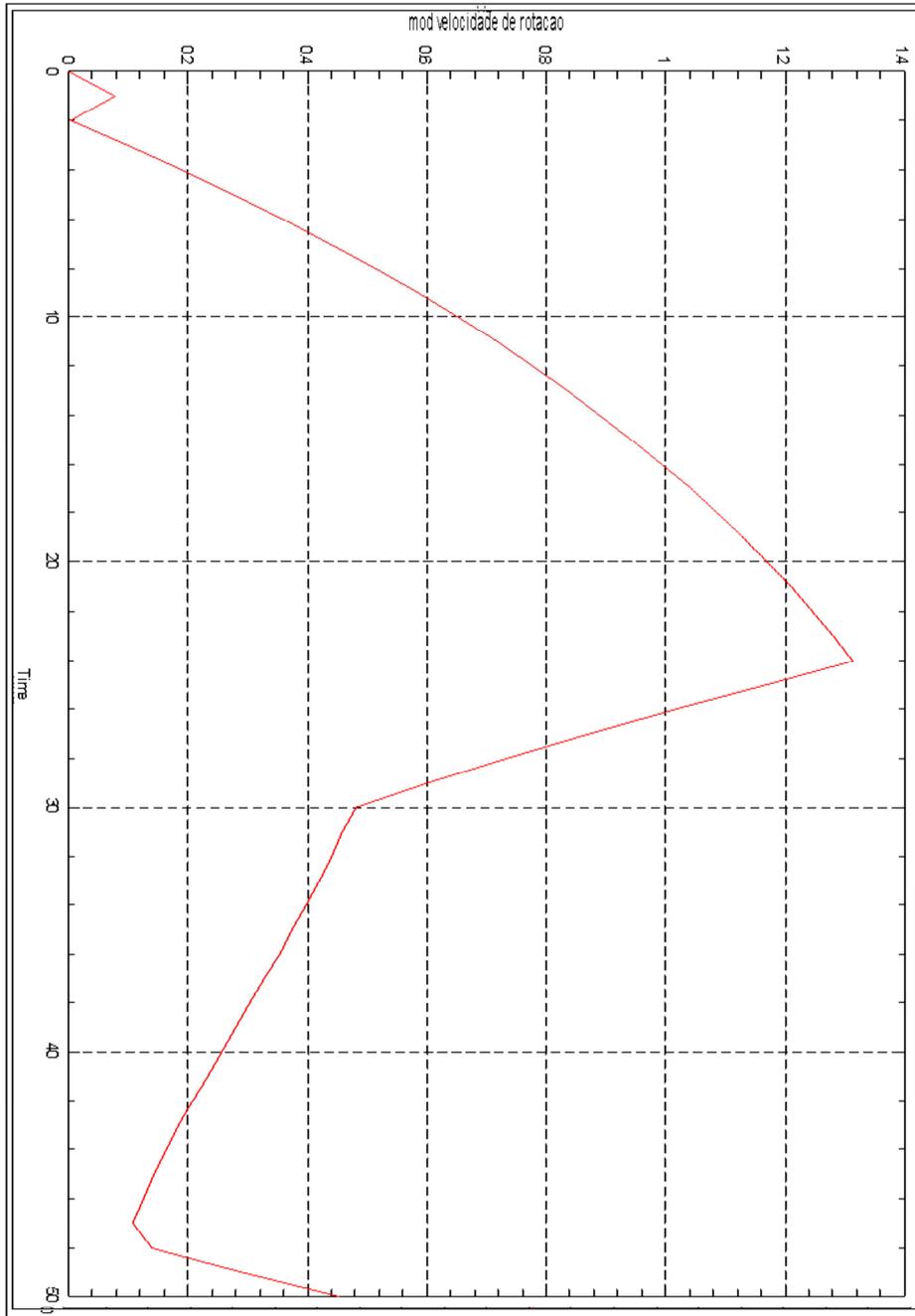


Figura 5-3: Velocidade Angular com rastreo de saída.

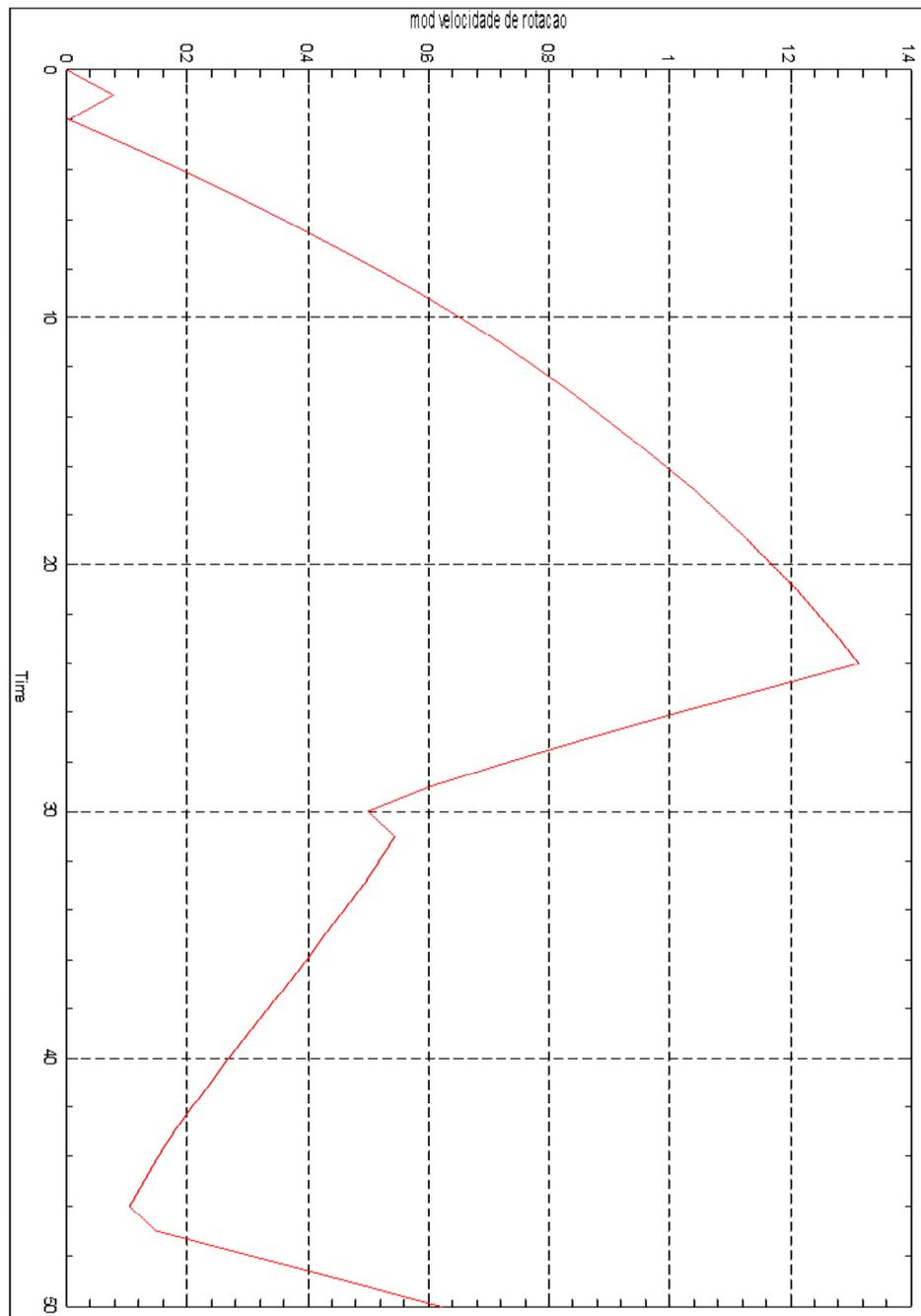


Figura 5-4: Velocidade Angular sem rastreo de saída.

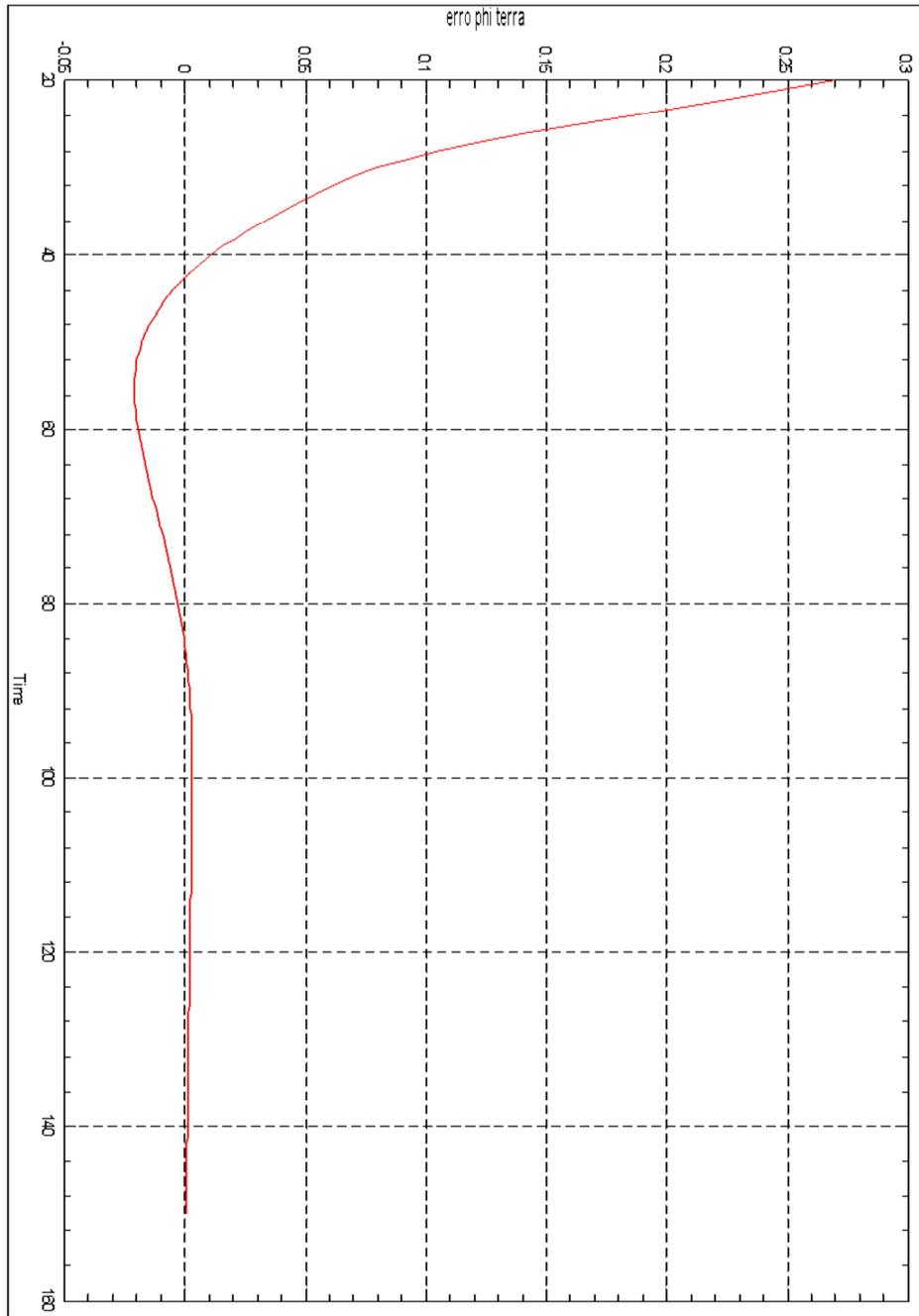


Figura 5-5: Sinal de erro com rastreo de saída

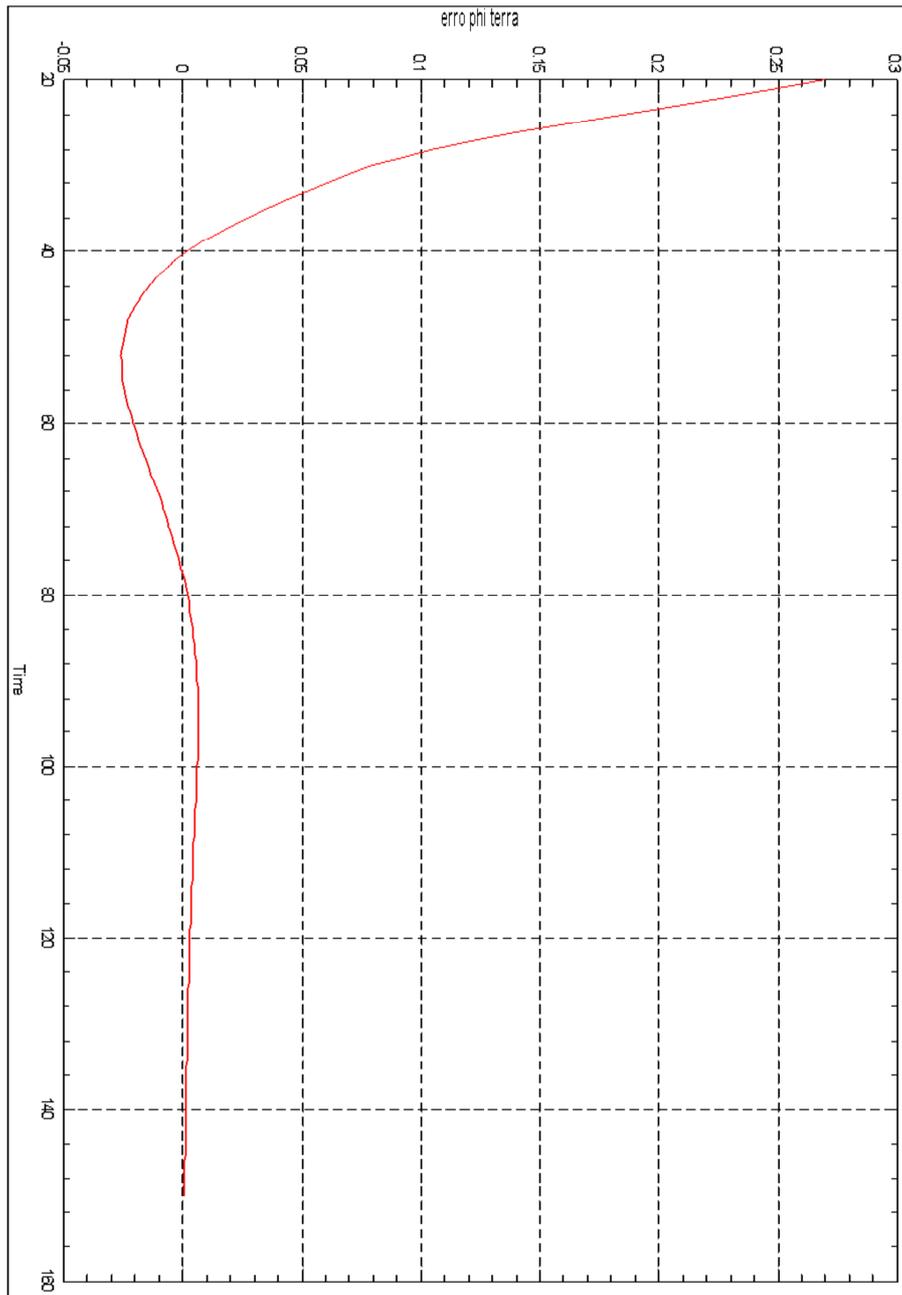


Figura 5-6: Sinal de erro sem rastreo de saída.

5.2. Combinações de “cross-fading” e “anti-windup” no Modo de Navegação Fina.

As implementações de “anti-windup” e “cross-fading” foram de acordo com o descrito nas Sessões 3.2.2 e 3.2.3, respectivamente. As Figuras 5-7 e 5-8 mostram que há saturação do sinal de controle durante o instante de chaveamento, em rotações de 180 e 30 graus, respectivamente, no eixo “y”, quando não há estratégia aplicada (sem “cross-fading” nem “anti-windup”). É possível executar a manobra e chavear para o Modo Nominal em menos de 100 e 80 segundos, respectivamente.

A Figura 5-9 mostra três rotações a partir de uma atitude inicial de (30, 30, 30) graus, sem estratégias aplicadas (nem “anti-windup” nem “cross-fading”). A PMM leva menos de 200 segundos para entrar no Modo Nominal.

A Figura 5-10 mostra que aplicar a estratégia de “anti-windup” em uma situação de rotações com ângulos pequenos em vários eixos não evita a saturação do sinal de controle, mas melhora o tempo de execução das três rotações e a transição para o próximo modo.

As Figuras 5-11 e 5-12 mostram que o uso de “cross-fading” evita a saturação do sinal de controle.

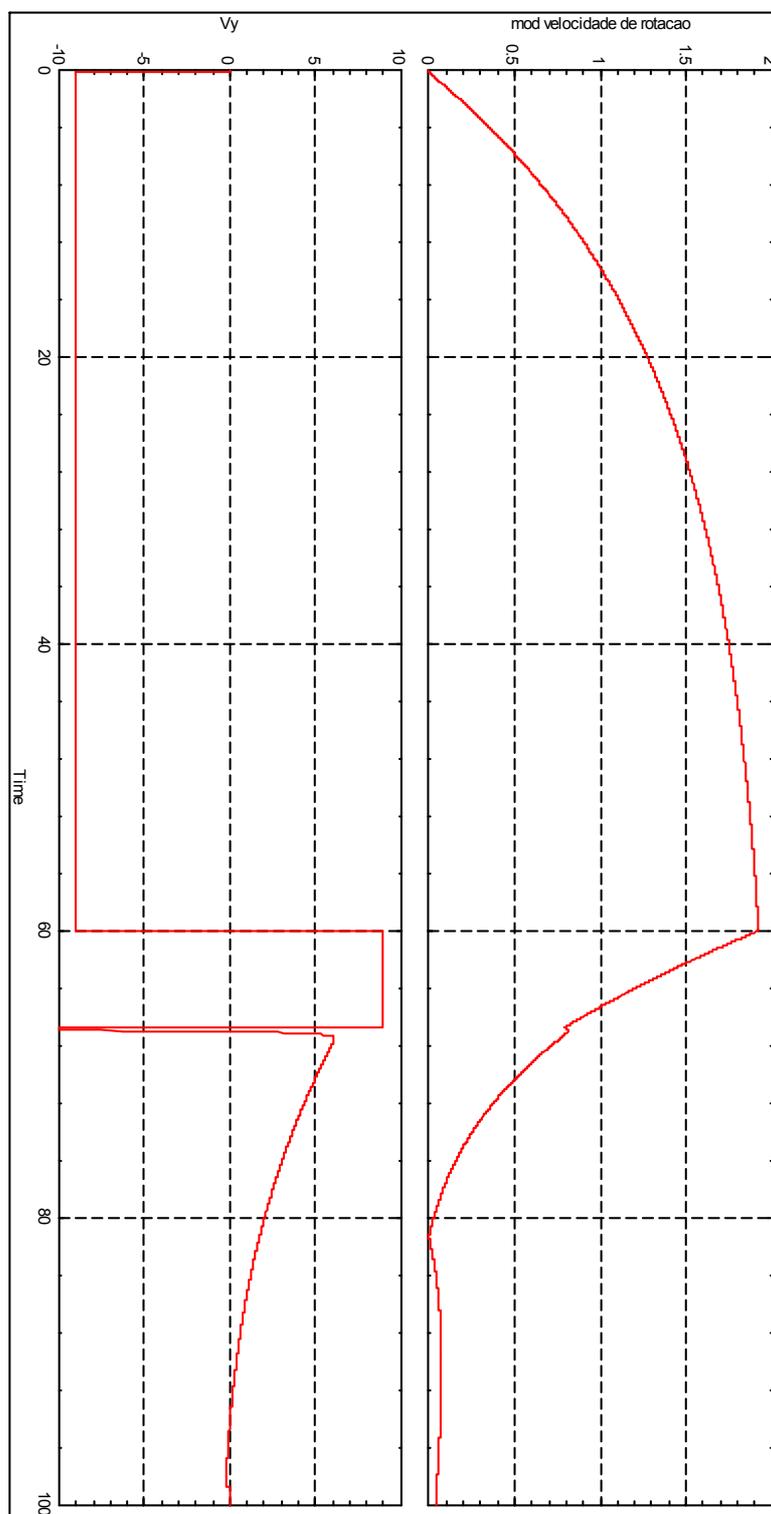


Figura 5-7: Módulo da velocidade angular (acima) e sinal de controle (abaixo) sem “cross-fading” nem “anti-windup”.

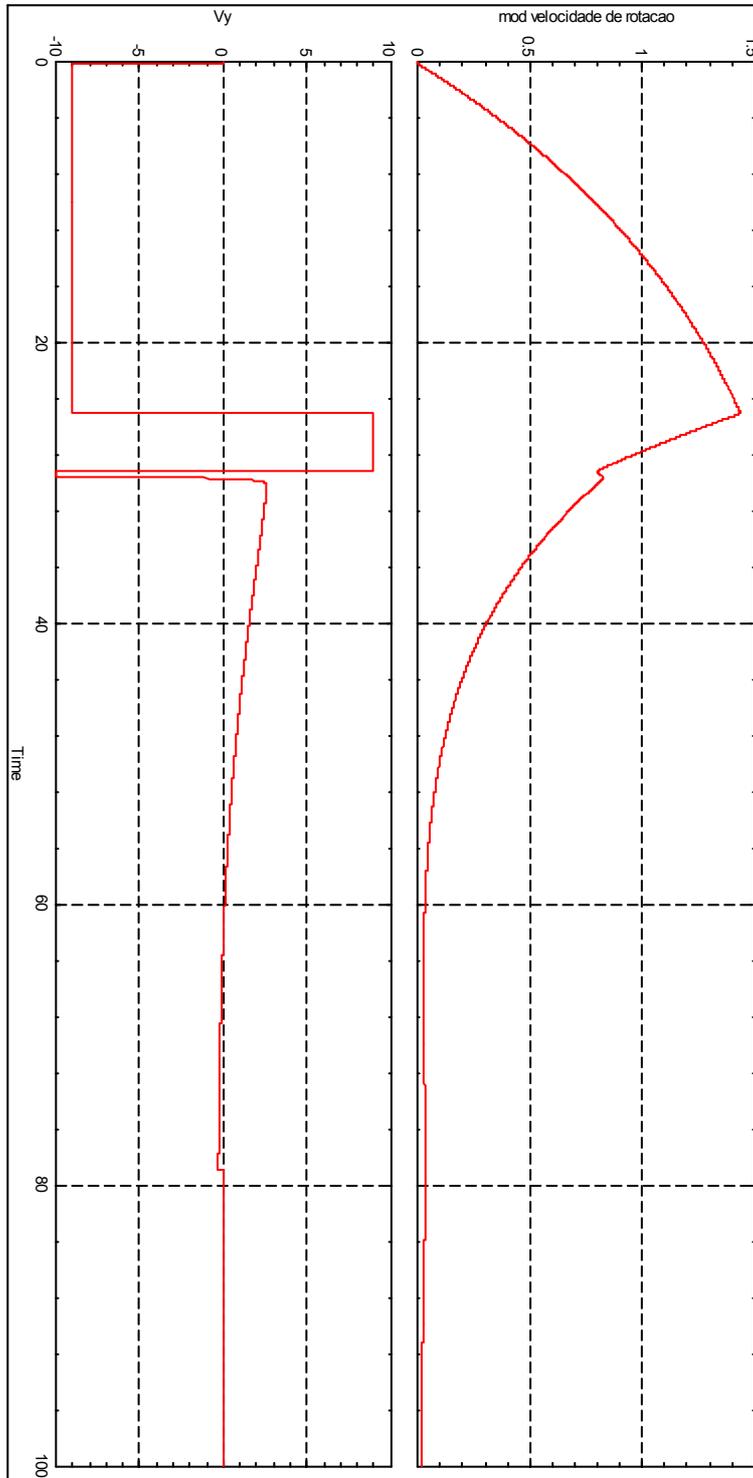


Figura 5-8: Módulo da velocidade angular (acima) e sinal de controle (abaixo) sem “cross-fading” nem “anti-windup”.

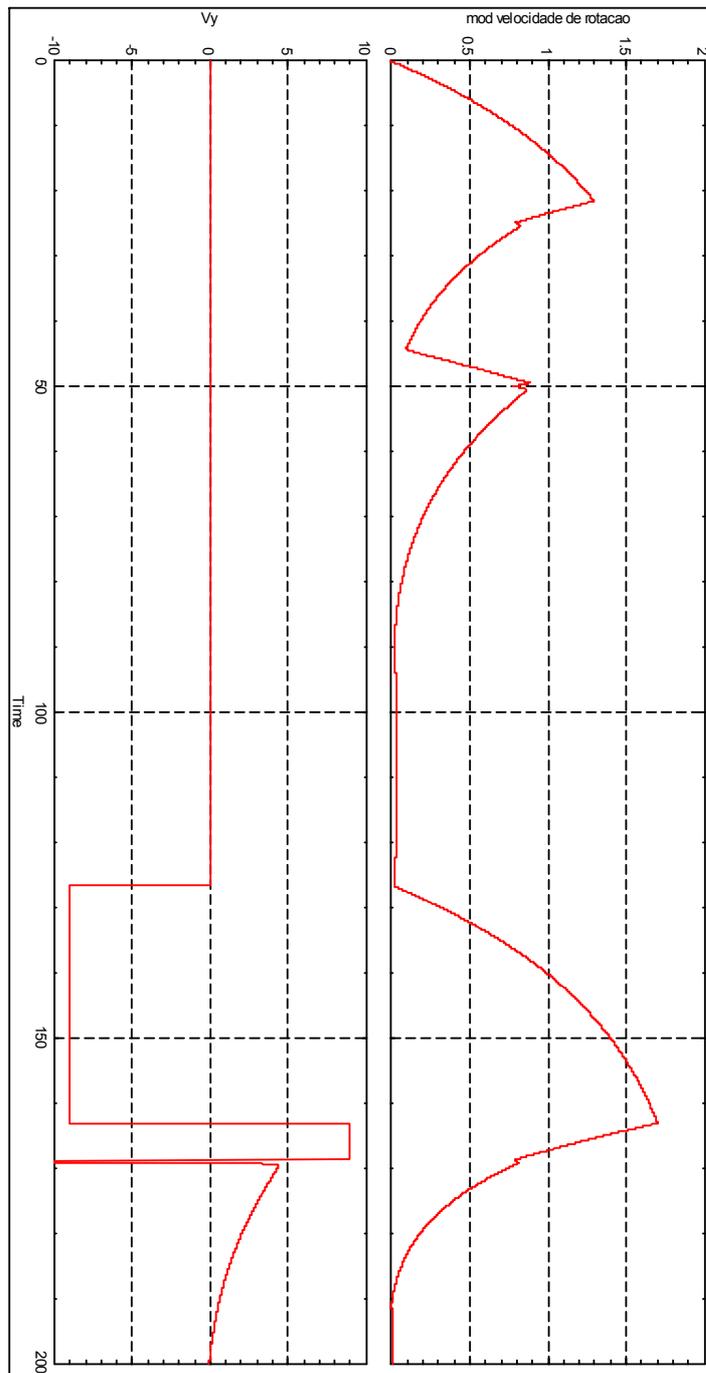


Figura 5-9: Módulo da velocidade angular (acima) e sinal de controle no eixo y (abaixo) sem “cross-fading” nem “anti-windup”.

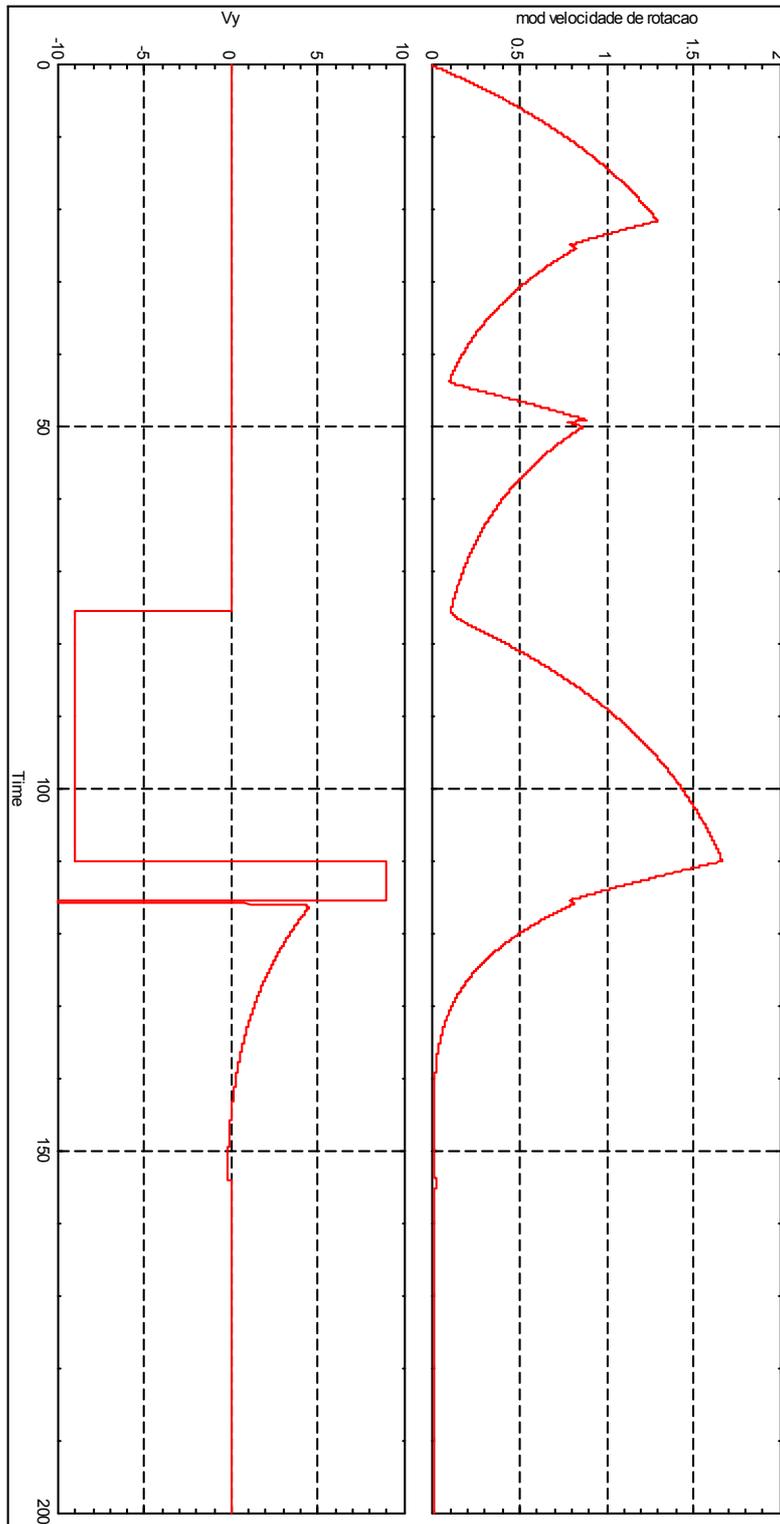


Figura 5-10: Módulo da velocidade angular total (acima) e sinal de controle no eixo y (abaixo) com “anti-windup”.

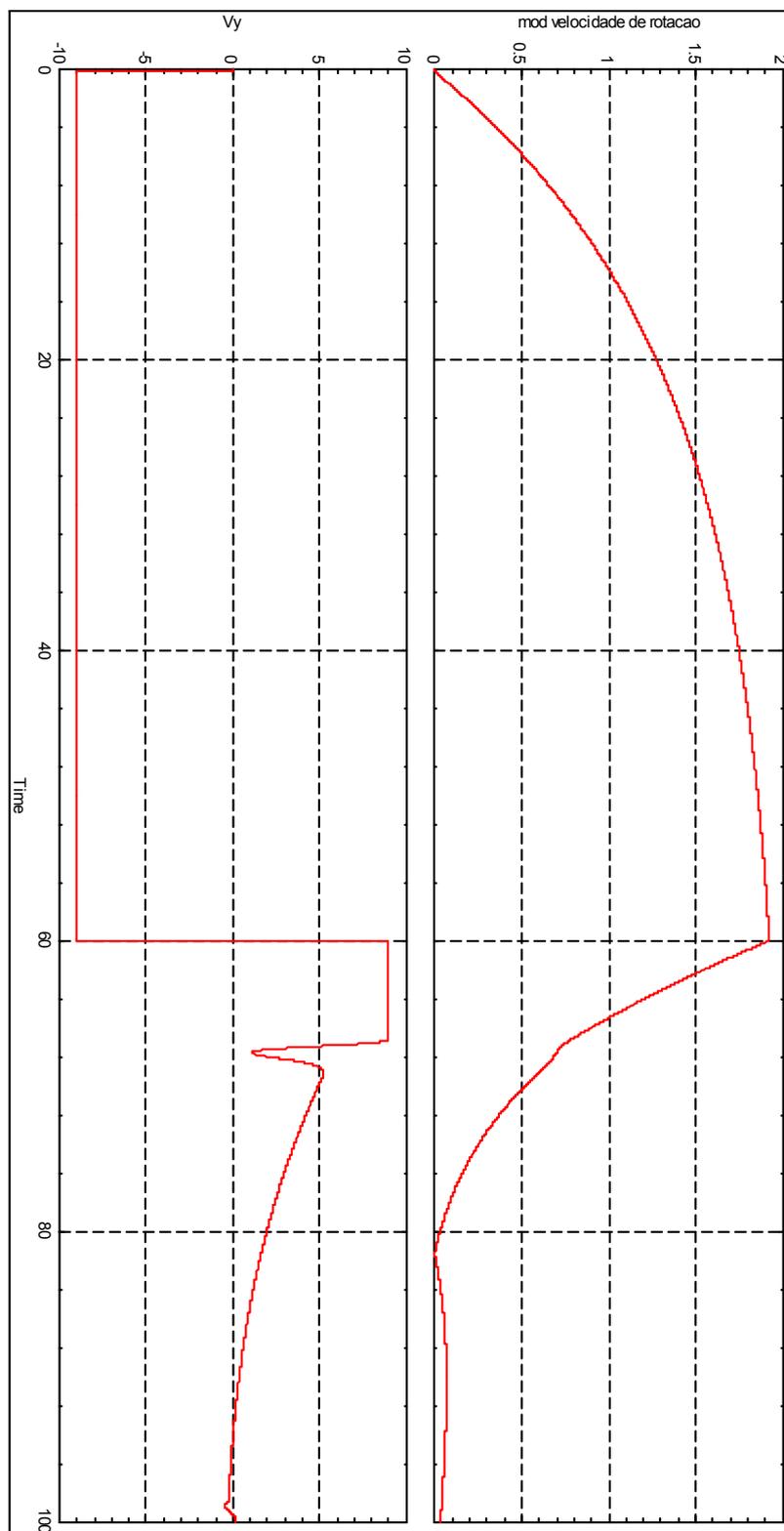


Figura 5-11: Módulo da velocidade angular (acima) e o sinal de controle (abaixo) com “cross-fading”.

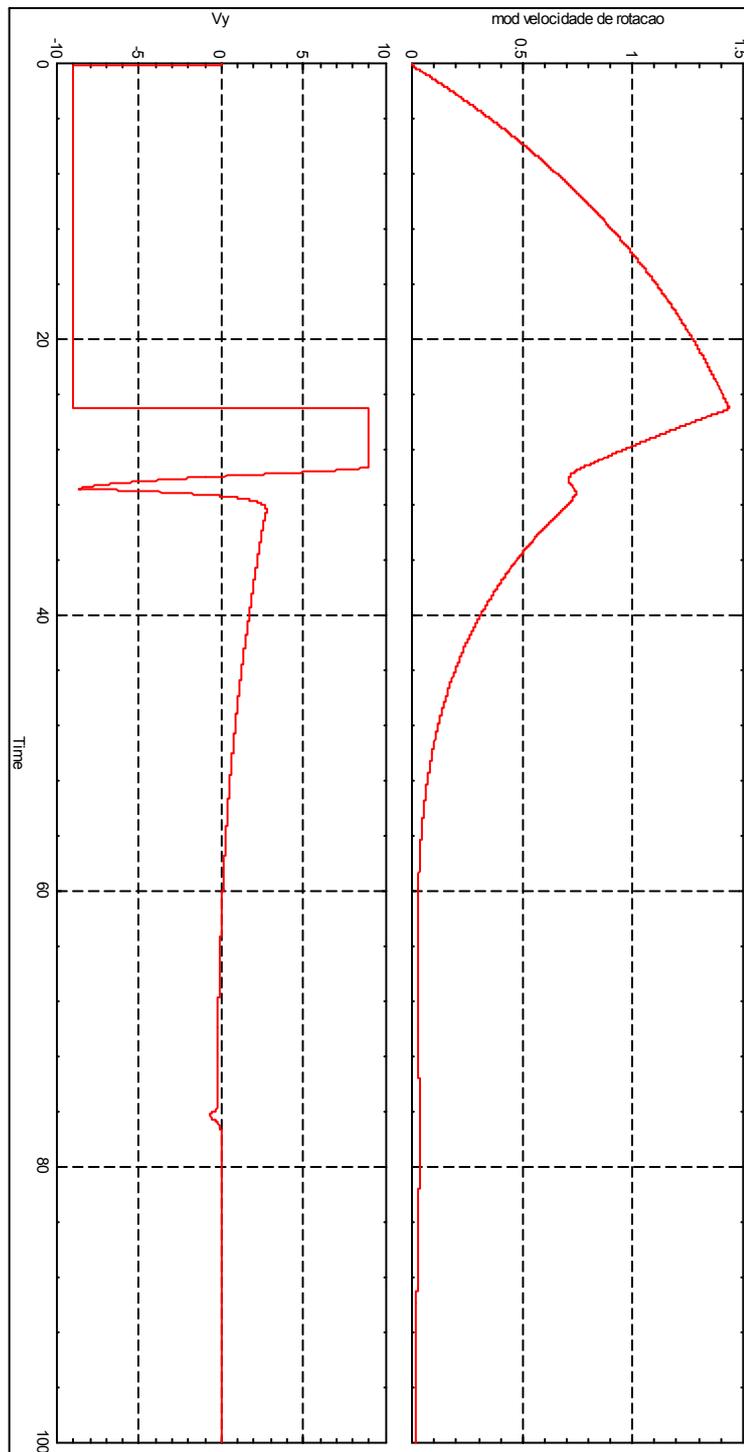


Figura 5-12: Módulo da velocidade angular (acima) e o sinal de controle (abaixo) com “cross-fading”.

5.3. Par de Motores de Corrente Contínua

A configuração do sistema foi a exibida na Figura 4-4, da Sessão 3.1.2. Introduzimos um pulso como uma perturbação no sinal do atuador de um dos motores. Seu valor foi 1, e começou em 10 segundos e terminou aos 40 segundos.

A Figura 5-13 mostra a diferença de posição dos motores, com apenas o PLL ativo, sem implementação do “cross-fading”, mostrando fortes sobressinais.

A Figura 5-14 mostra a diferença de posição dos motores, com um modo incluindo apenas um PID, e outro incluindo um PLL. Há um sobressinal maior e a diferença não é completamente anulada até que a perturbação seja desativada. Há também um rápido chaveamento entre modos, indesejado, após o sobressinal.

A Figura 5-15 mostra a diferença de posição entre os motores, o sinal de chaveamento original, e o sinal de chaveamento suavizado, entre um modo com um PID e um modo com PID mais PLL combinados. O sobressinal é menor, e o sinal oscila em torno de zero, mas com variações muito menores.

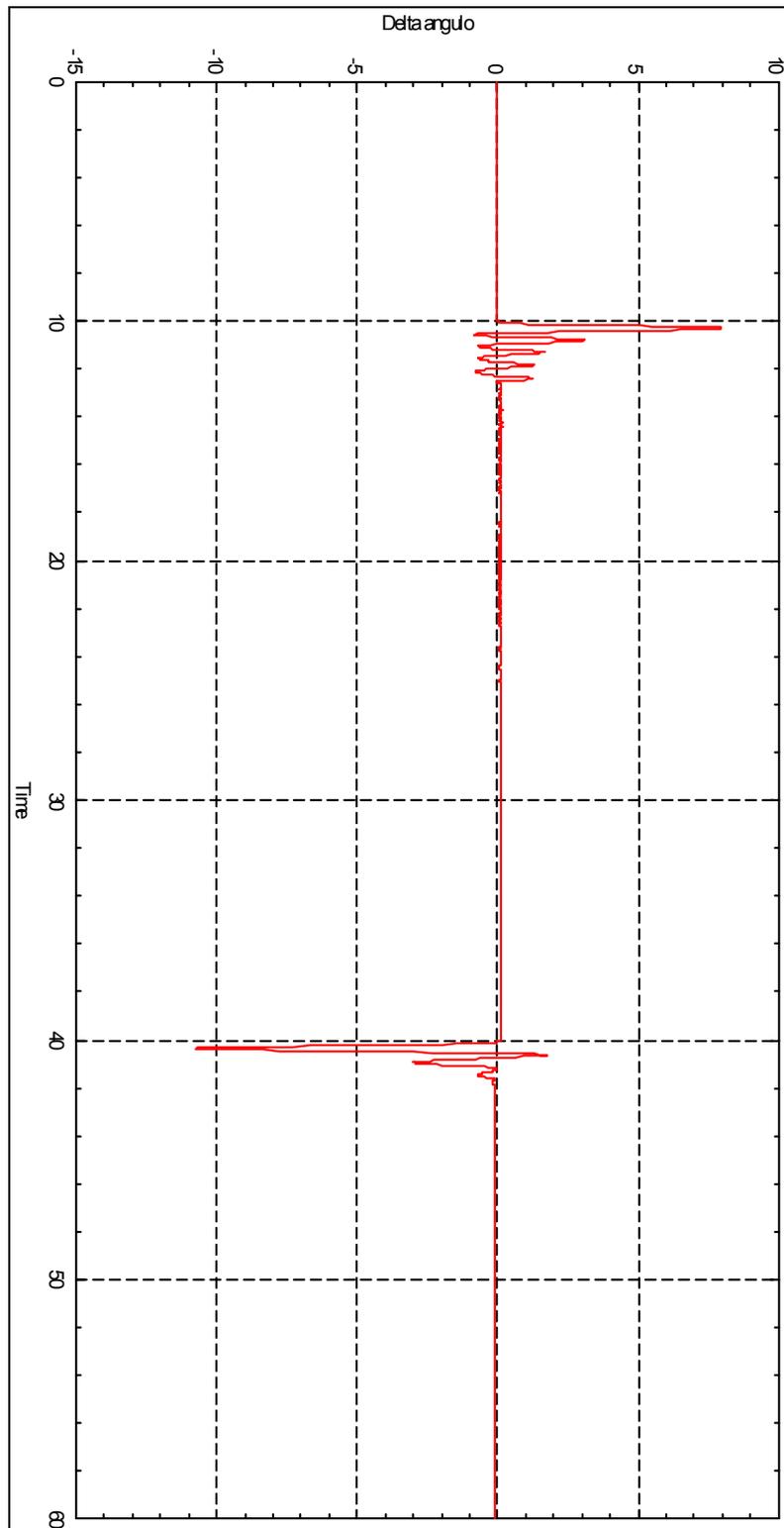


Figura 5-13: Diferença de posição entre os motores com 1 PLL mas sem “cross-fading”, em graus.

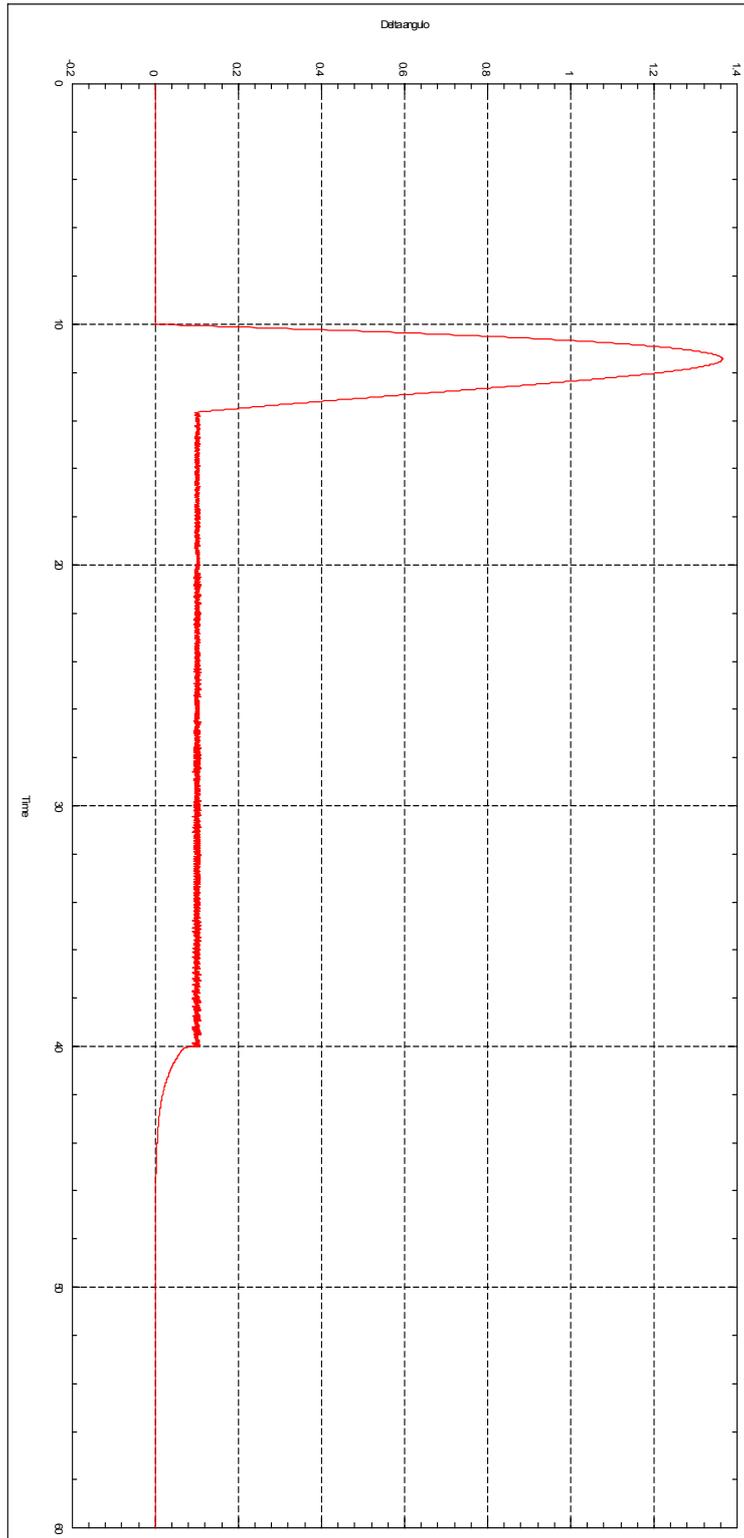


Figura 5-14: Mostra a diferença de posição entre os motores com 1 PLL e 1 PID, mas sem “cross-fading”.

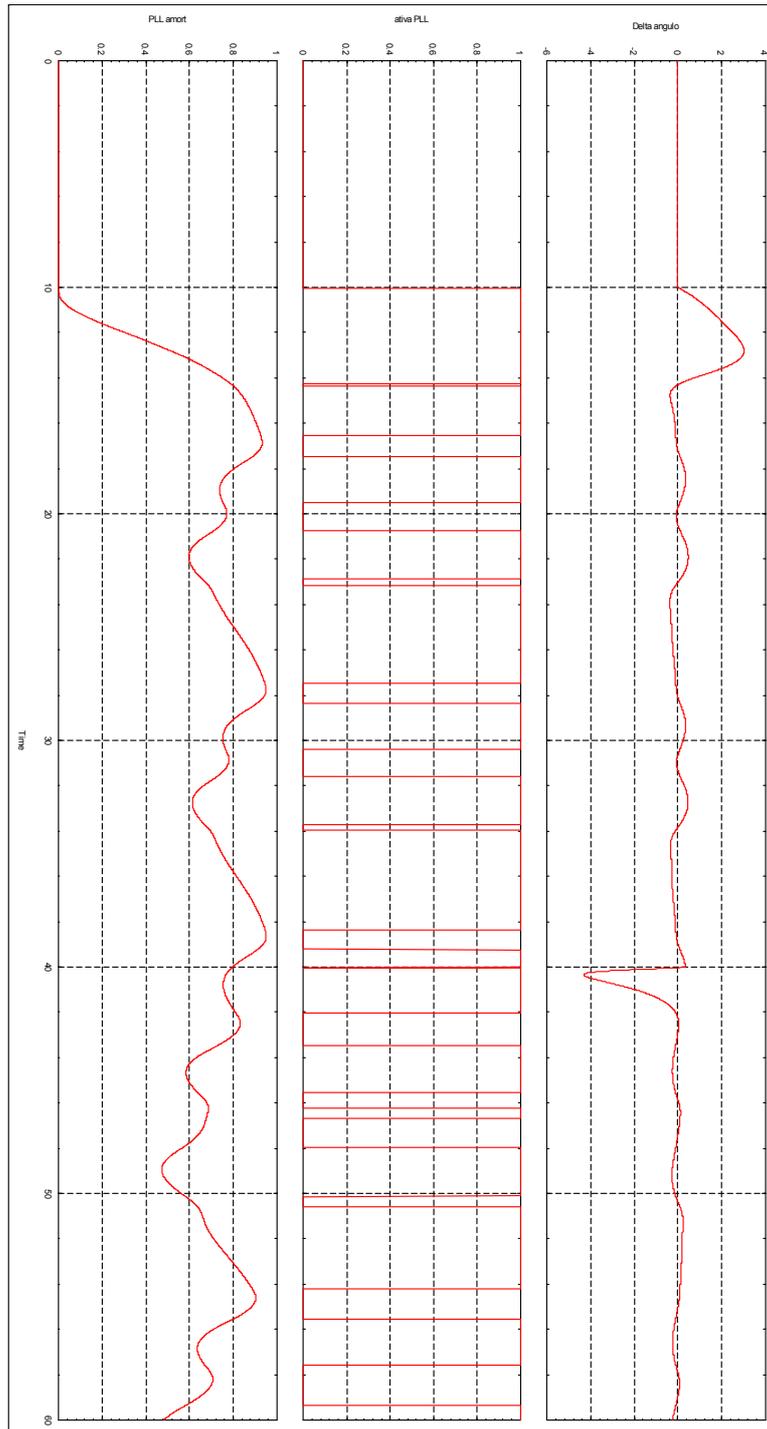


Figura 5-15: Mostra a diferença de posição entre os motores (acima), o sinal de chaveamento original (meio), e o sinal de chaveamento suavizado (abaixo) com 1 PID e 1 PLL+1 PID, mas sem “cross-fading”.

5.4. Critérios integrais

A implementação dos critérios integrais para o chaveamento entre um controlador proporcional e um controlador PID são exemplificadas a seguir. Para destacar o que foi obtido, antes mostraremos exemplos sem atenuação, e com atenuação por parâmetros arbitrários.

O modelo adotado foi uma versão da PMM simplificada para um eixo. A Figura 5-16 mostra o sinal de controle visto pela planta durante vários chaveamentos entre os dois tipos de controlador, um proporcional e um PID, sem nenhuma estratégia de atenuação de impacto.

A Figura 5-17 mostra o sinal de controle visto pela planta durante vários chaveamentos entre os dois tipos de controlador. Foi utilizado um sistema de controle PID + Integrador não ótimo durante o chaveamento; foram usados parâmetros que facilitassem a visualização de múltiplas transições na escala apresentada. É possível observar que essa escolha suavizou a transição, que não é mais composta por degraus abruptos.

A Figura 5-18 mostra um exemplo, não ótimo, do que ocorre caso o chaveamento seja lento em comparação com a planta. O sinal do controlador C2 distanciou-se demais do valor registrado durante a transição. Caso o problema ocorra até mesmo com uma trajetória ótima, isso poderá ser compensado adicionando uma nova etapa de transição.

Para os modelos da PMM e dos dois motores de indução, a curva dos funcionais estabelecidos não apresentou um ponto de mínimo. Portanto, para comparação, escolheremos o ζ tal que faça o J correspondente ter o valor de $10e-5$.

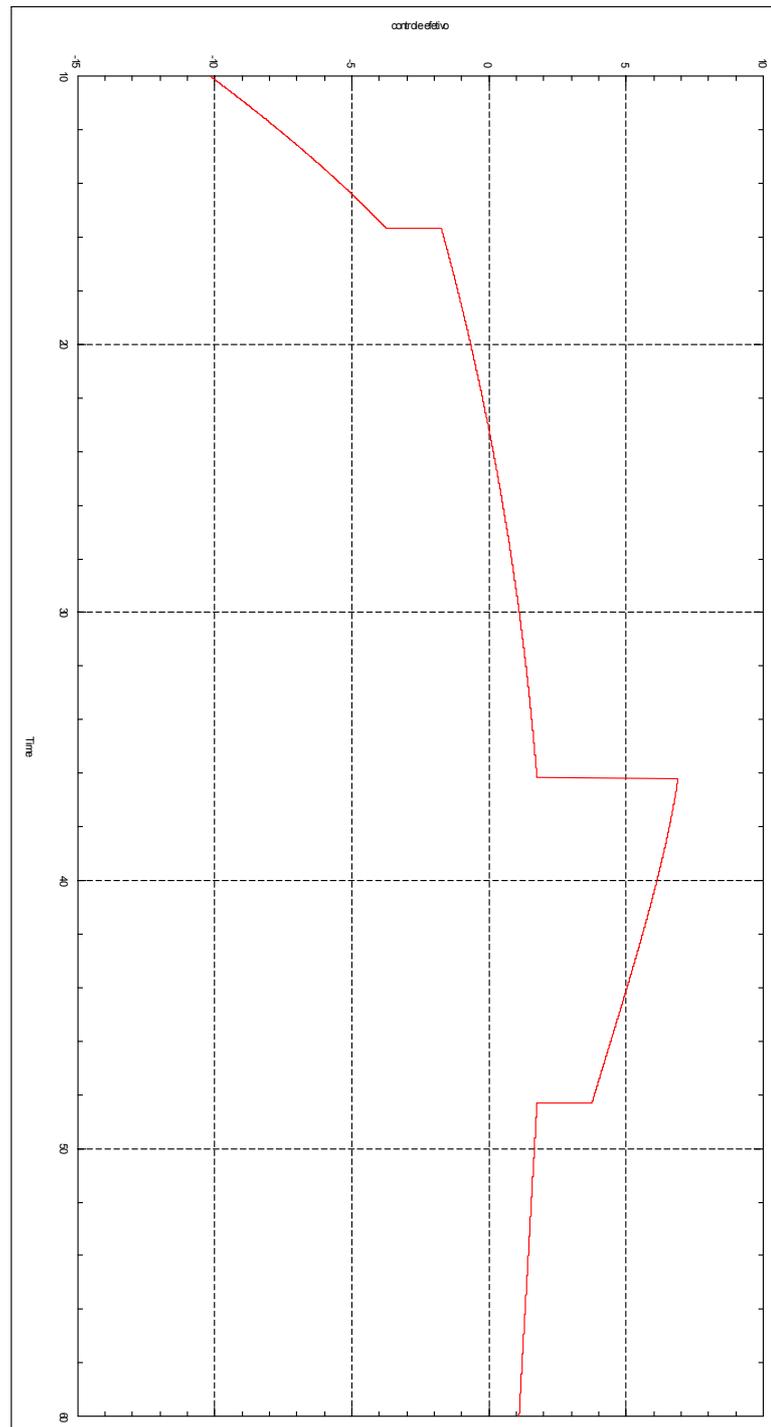


Figura 5-16: Sinal de controle ao longo de vários chaveamentos entre modos de controle sem nenhuma estratégia de atenuação de impacto.

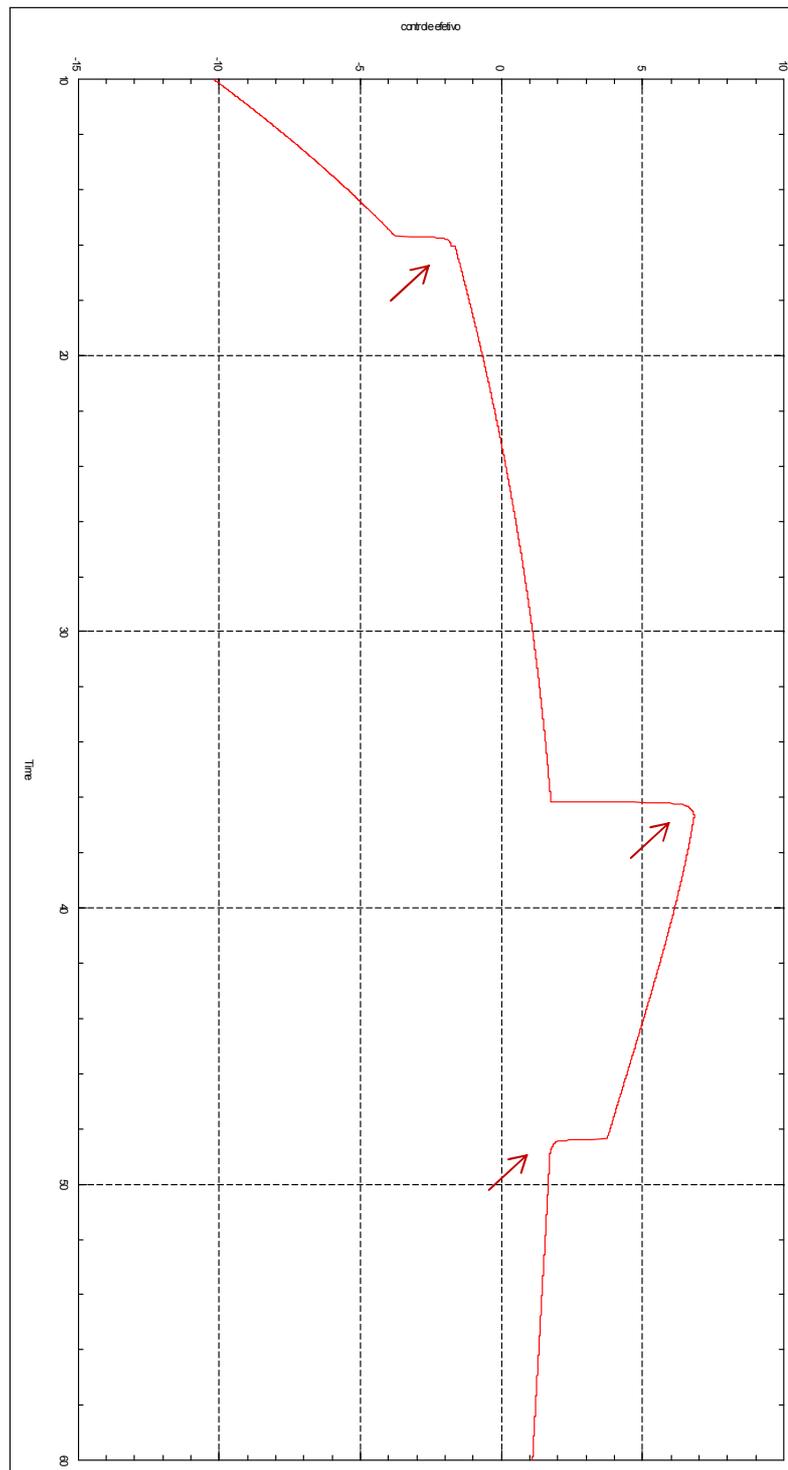


Figura 5-17: Sinal de controle ao longo de vários chaveamentos entre modos de controle com controle PID + Integrador intermediário não ótimo.

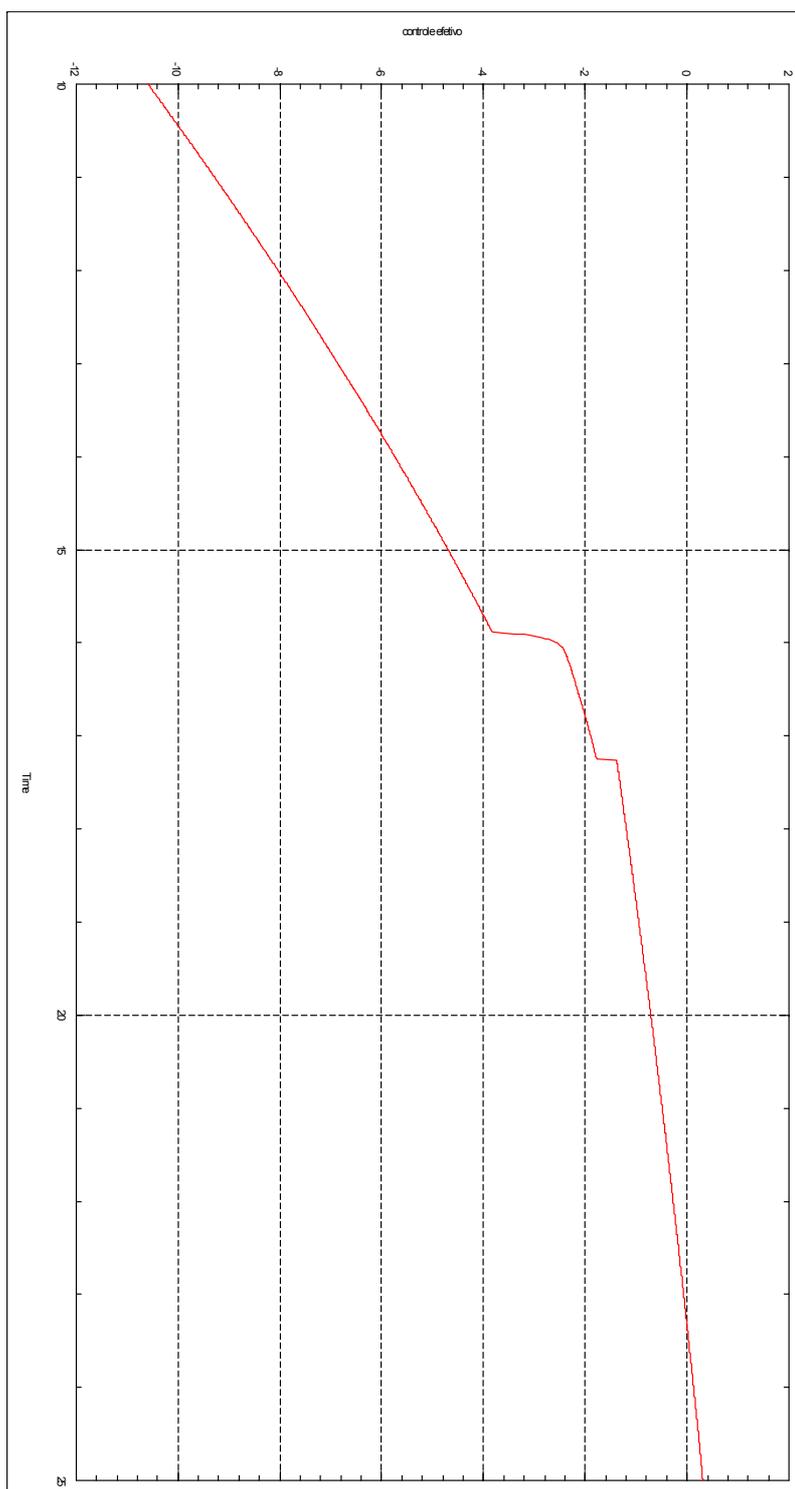


Figura 5-18: Sinal de controle ao longo de vários chaveamentos entre modos de controle com controle PID + Integrador intermediário não ótimo, lento em comparação com o segundo controlador.

5.4.1. Critério Integral do Quadrado do Erro (ISE).

A Figura 5-19 mostra o comportamento do funcional J ao longo de ζ , para os valores do PID aplicados no exemplo da PMM ($k_d = 40,593$; $k_p = 454,11$; $k_i = 1$). A relação entre o funcional J e a variável ζ (zeta)

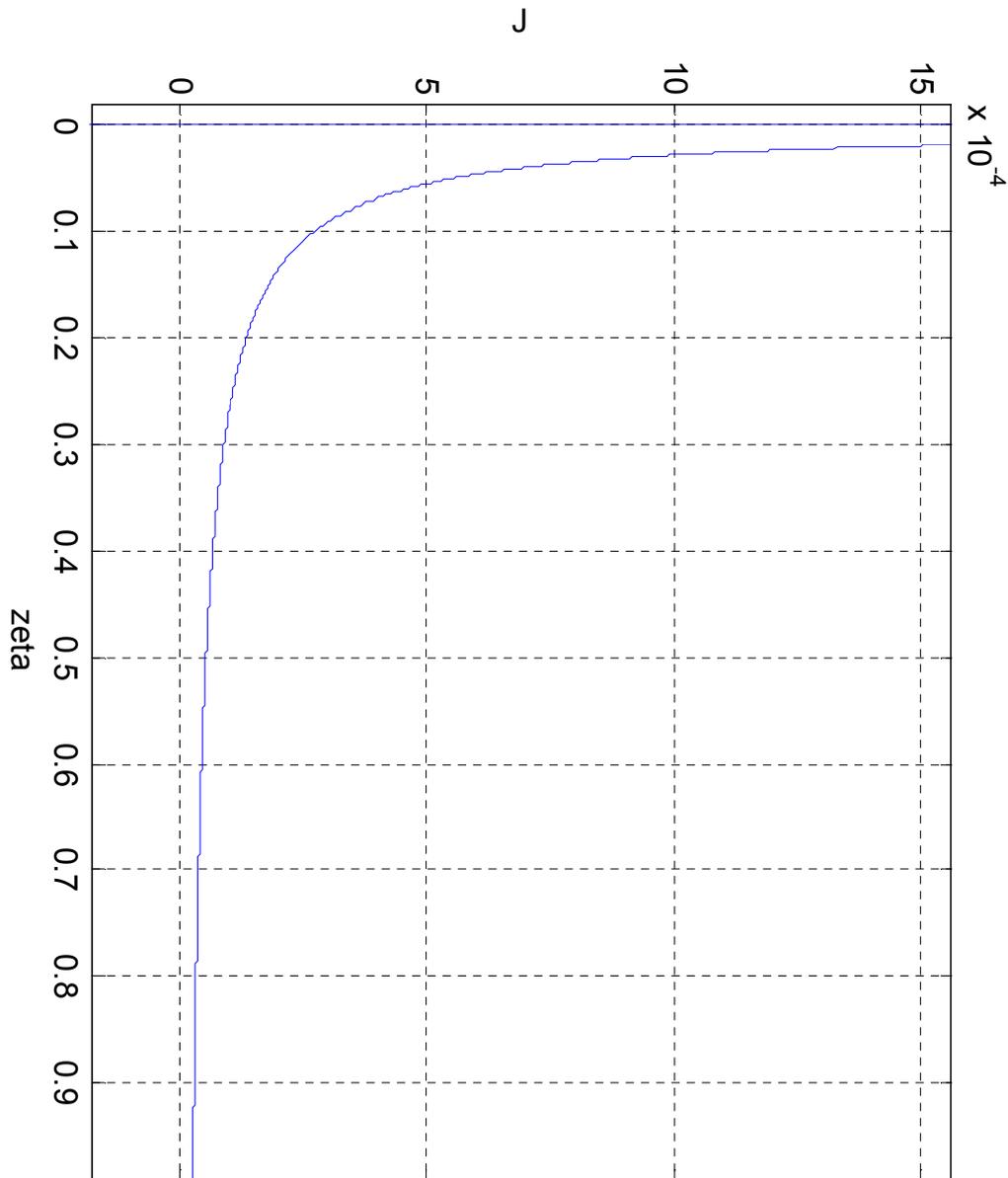


Figura 5-19: Funcional J (ISE) ao longo de ζ .

A Figura 5-20 mostra o comportamento do funcional J ao longo de ζ , para os valores de PID usados no exemplo dos dois motores de indução ($k_d = 0,04$; $k_p = 3$; $k_i = 1,15$).

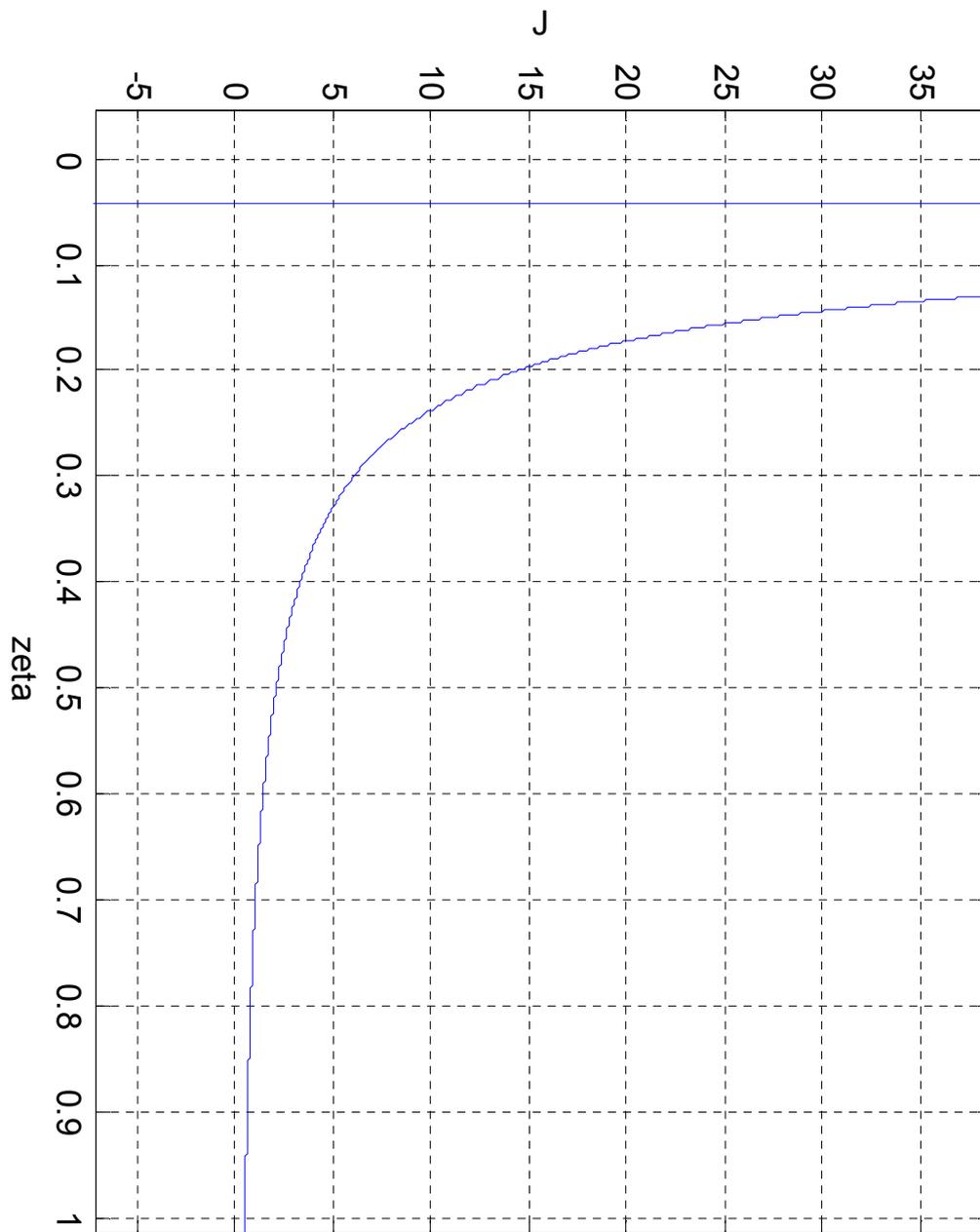


Figura 5-20: Funcional J (ISE) ao longo de ζ .

5.4.2. Critério Integral do Tempo vezes o Quadrado do Erro (ITSE)

A Figura 5-21 mostra o comportamento funcional J ao longo de ζ , para os valores do PID aplicados no exemplo da PMM ($k_d = 40,593$; $k_p = 454,11$; $k_i = 1$).

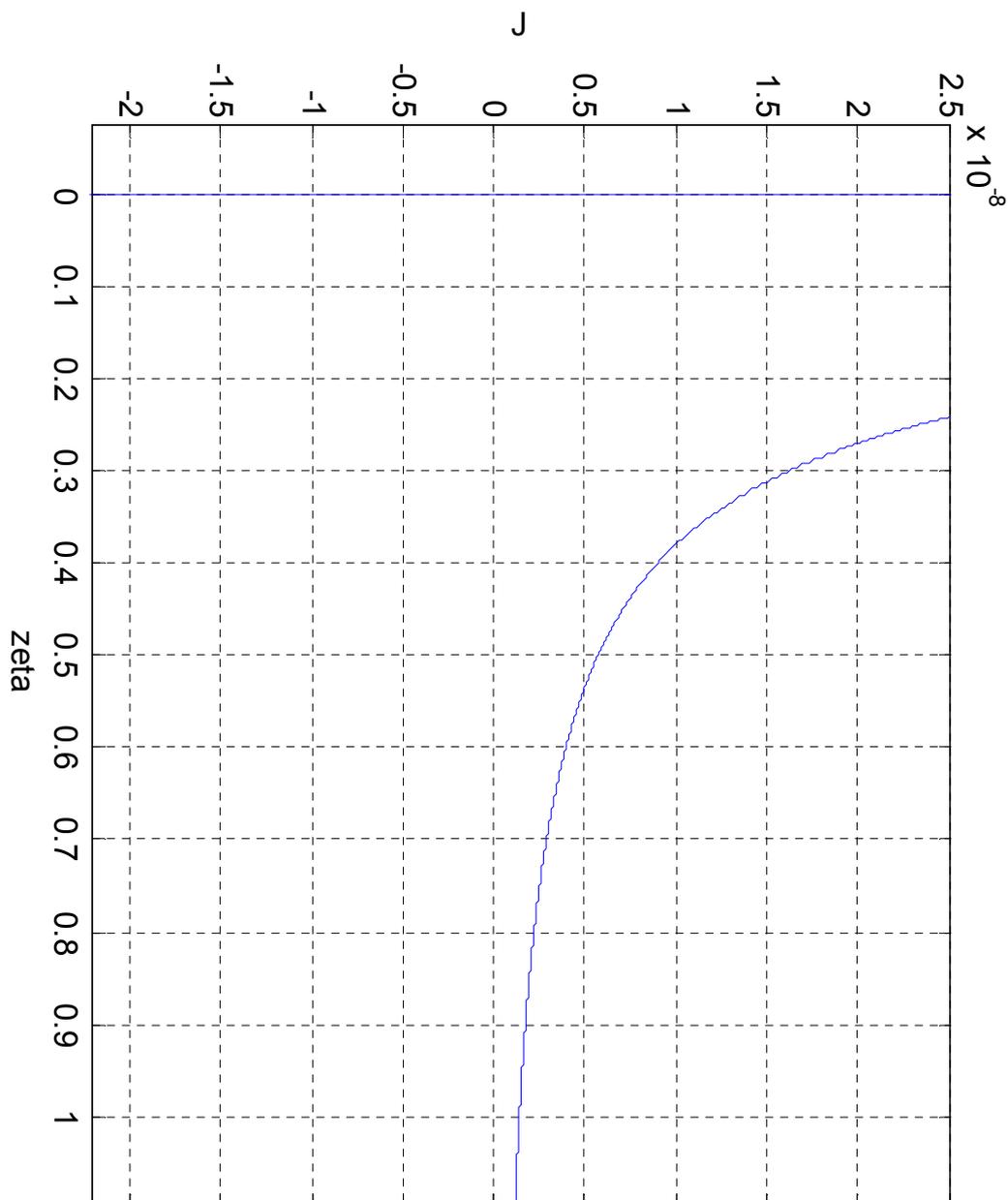


Figura 5-21: Funcional J (ITSE) ao longo de ζ .

A Figura 5-22 mostra o comportamento do funcional J ao longo de ζ , para os valores de PID aplicados no exemplo dos dois motores de indução ($k_d = 0,04$; $k_p = 3$; $k_i = 1,15$).

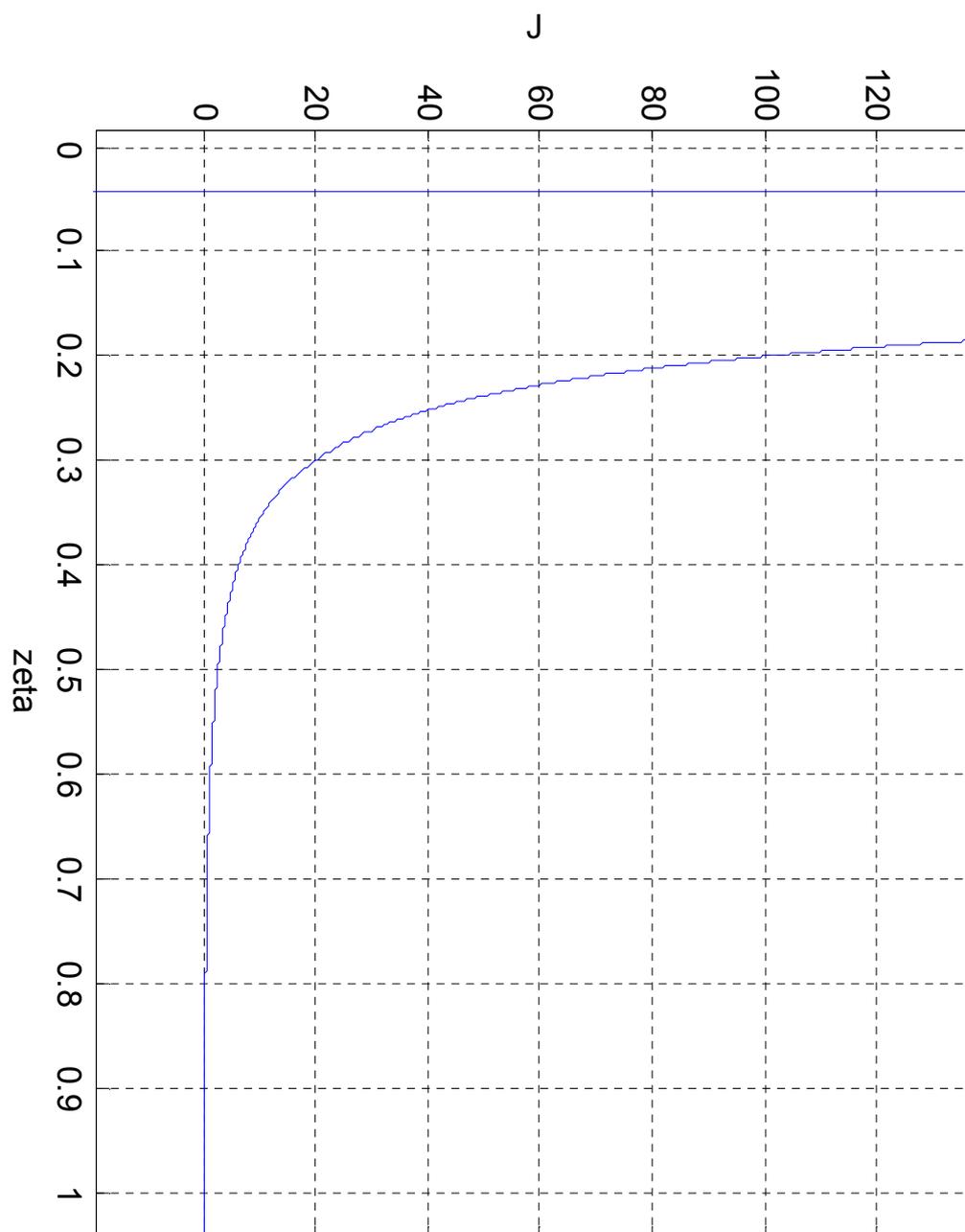


Figura 5-22: Funcional J (ITSE) ao longo de ζ .

5.4.3. Critério Integral do Módulo do Erro (IME)

A Figura 5-23 mostra o comportamento do funcional J ao longo de ζ , para os valores do PID usados no exemplo da PMM, válido para $\zeta > 7,874e-4$.

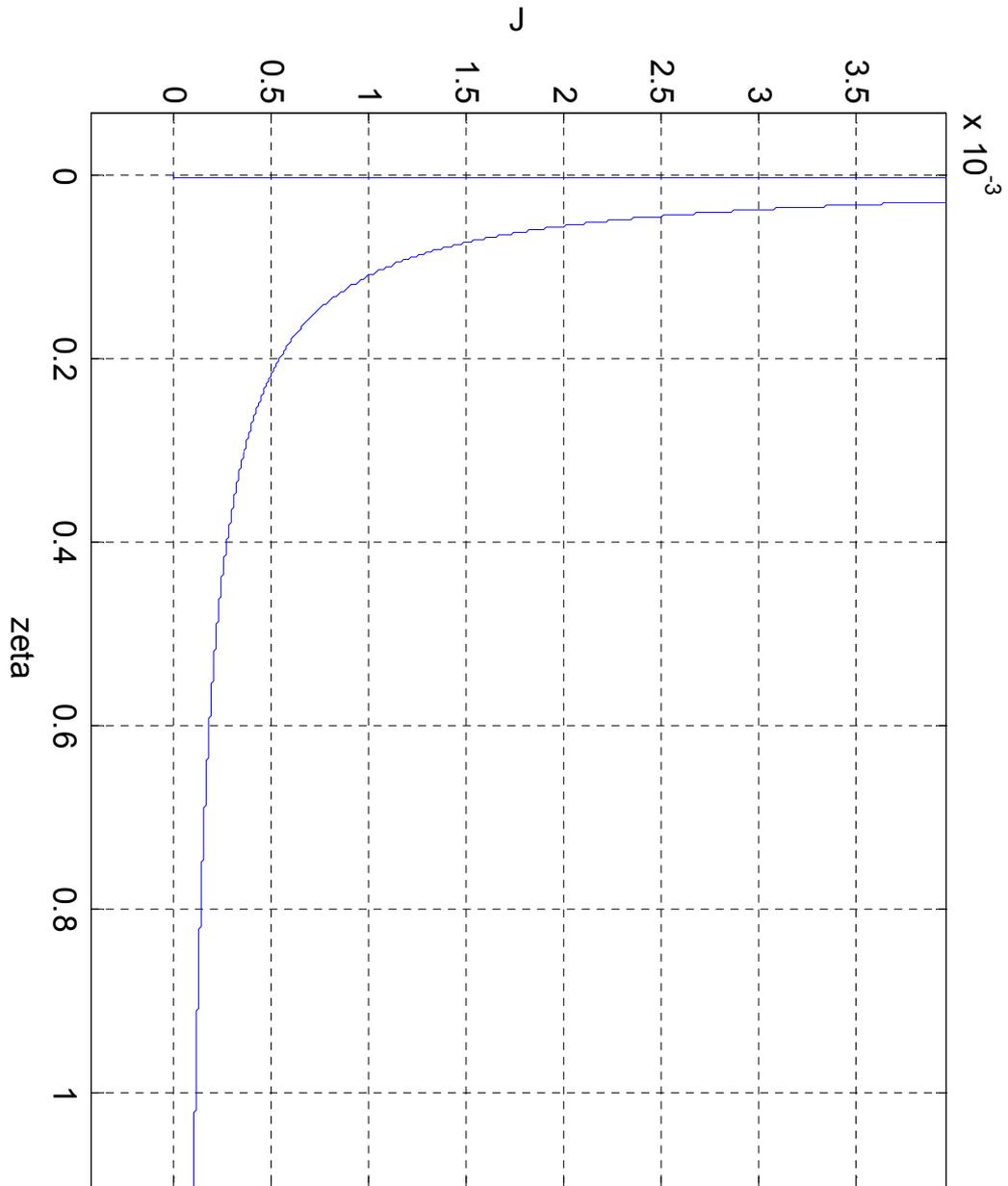


Figura 5-23: Funcional J (IME) ao longo de ζ .

A Figura 5-24 mostra o comportamento do funcional J ao longo de ζ , com os dados do PID dos motores de indução, válido para $\zeta > 0,0418$.

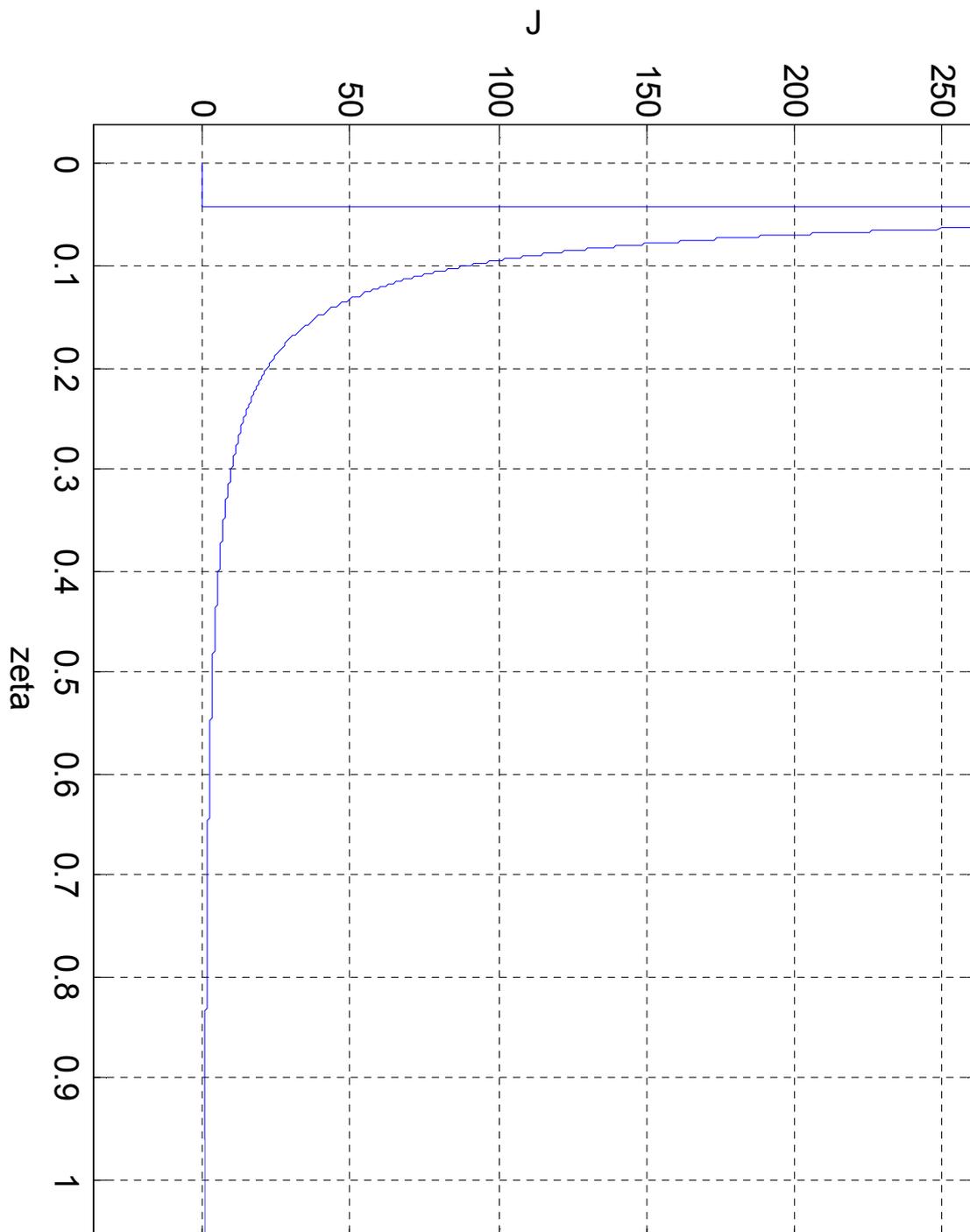


Figura 5-24: Funcional J (IME) ao longo de ζ .

5.4.4. Critério Integral do Tempo vezes o Módulo do Erro (ITME)

A Figura 5-25 mostra o gráfico de J ao longo de ζ , para os valores de PID adotados no caso da PMM, válido para $\zeta > 7,874e-4$.

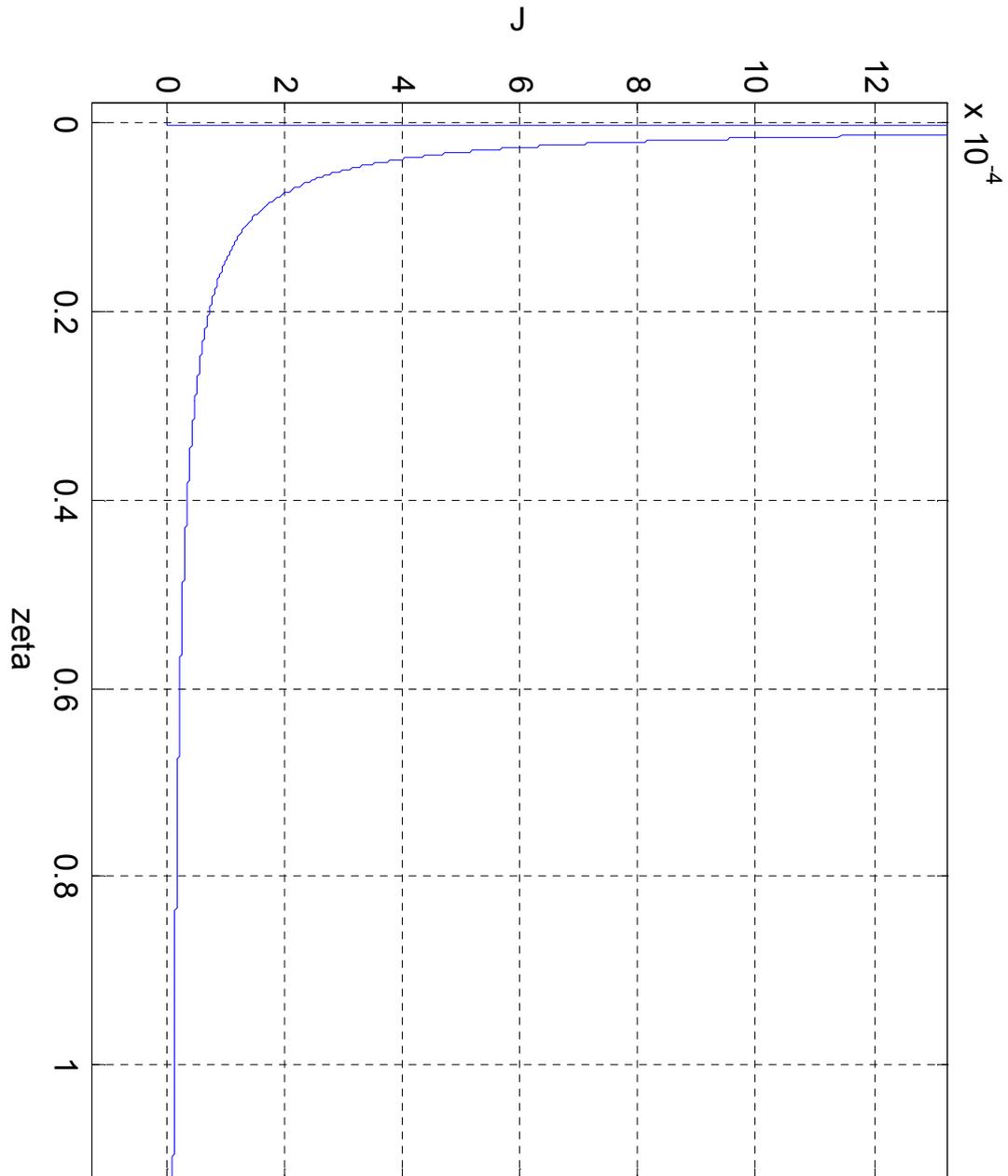


Figura 5-25: Funcional J (ITME) ao longo de ζ

A Figura 5-26 mostra o gráfico de J com os dados do PID para os motores de indução, válido para $\zeta > 0,0418$.

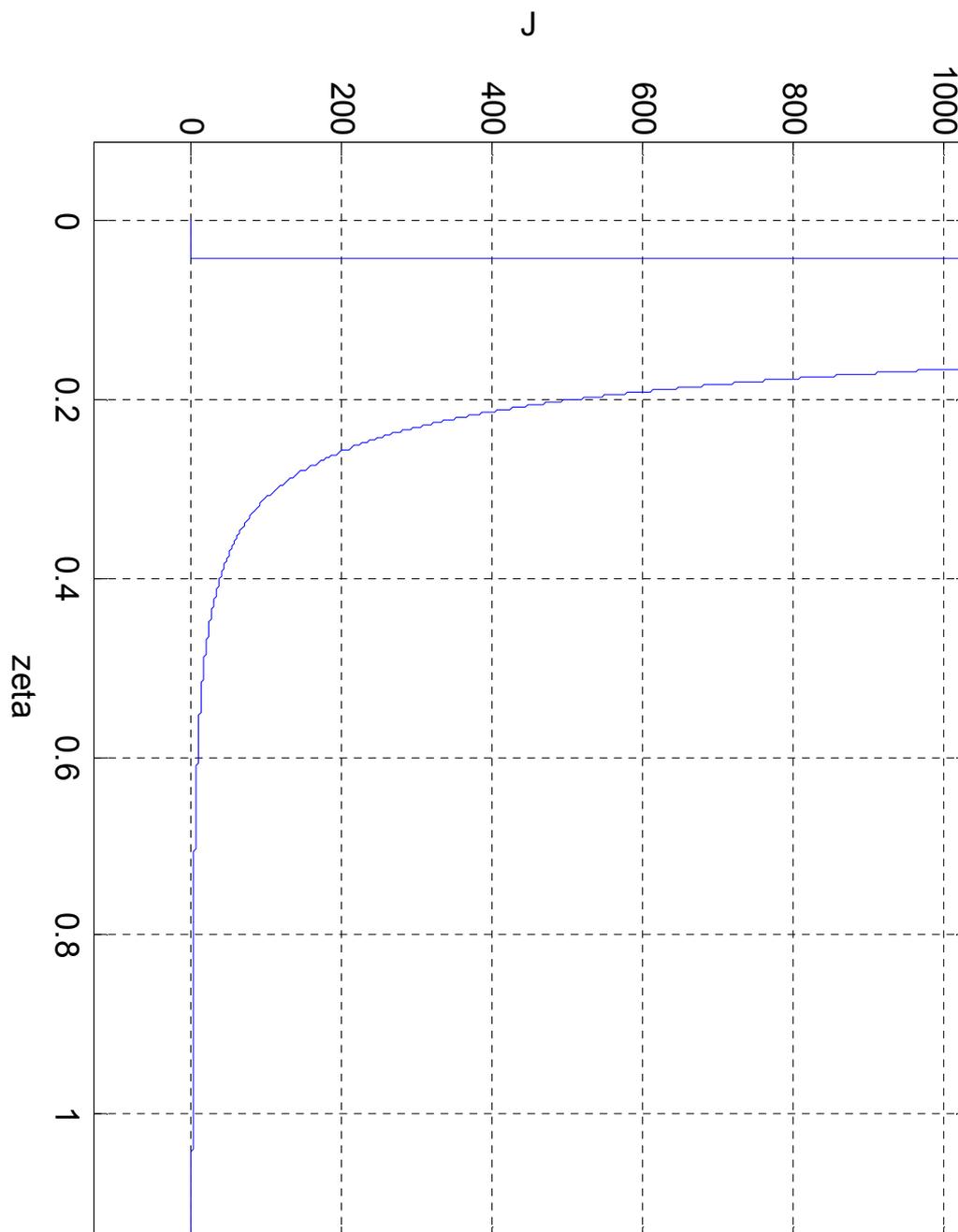


Figura 5-26: Funcional J (ITME) ao longo de ζ .

5.4.5. Comparação dos Critérios no Modelo da PMM

Arbitrando o valor de $J = 10e-5$, comparamos os quatro critérios ISE, ITSE, IME e ITME aplicando seus respectivos parâmetros ζ em situações idênticas. Os valores encontrados foram: ISE = 2,55, ITSE = 0,013, IME = 8,85, ITME = 1,35.

Observando os resultados, constatou-se que a multiplicação pelo tempo e o quadrado do erro têm correlação com parâmetros menores e transições mais lentas. Os critérios baseados no quadrado do erro são mais dependentes dos sobressinais, e os critérios baseados na multiplicação pelo tempo são mais dependentes do tempo de acomodação. Por outro lado, enquanto o módulo e o tempo são menores do que a unidade, eles tendem a ser menos relevantes para o critério.

Os parâmetros ζ variaram pelo menos duas ordens de magnitude de um critério para outro; portanto, a escolha dependerá das necessidades particulares da planta. De todos, o critério ITSE resultou numa transição muito mais lenta do que as outras três.

As Figuras 5-27, 5-28, 5-29 e 5-30 mostram o sinal de controle visto pela planta durante uma transição entre dois modos de controle, no modelo de 1 eixo da PMM, em diferentes escalas de tempo.

A Figura 5-31 mostra o sinal de controle efetivo visto pela planta, utilizando o critério ITSE, que foi o mais lento de todos. A transição foi completada em 0,06 s.

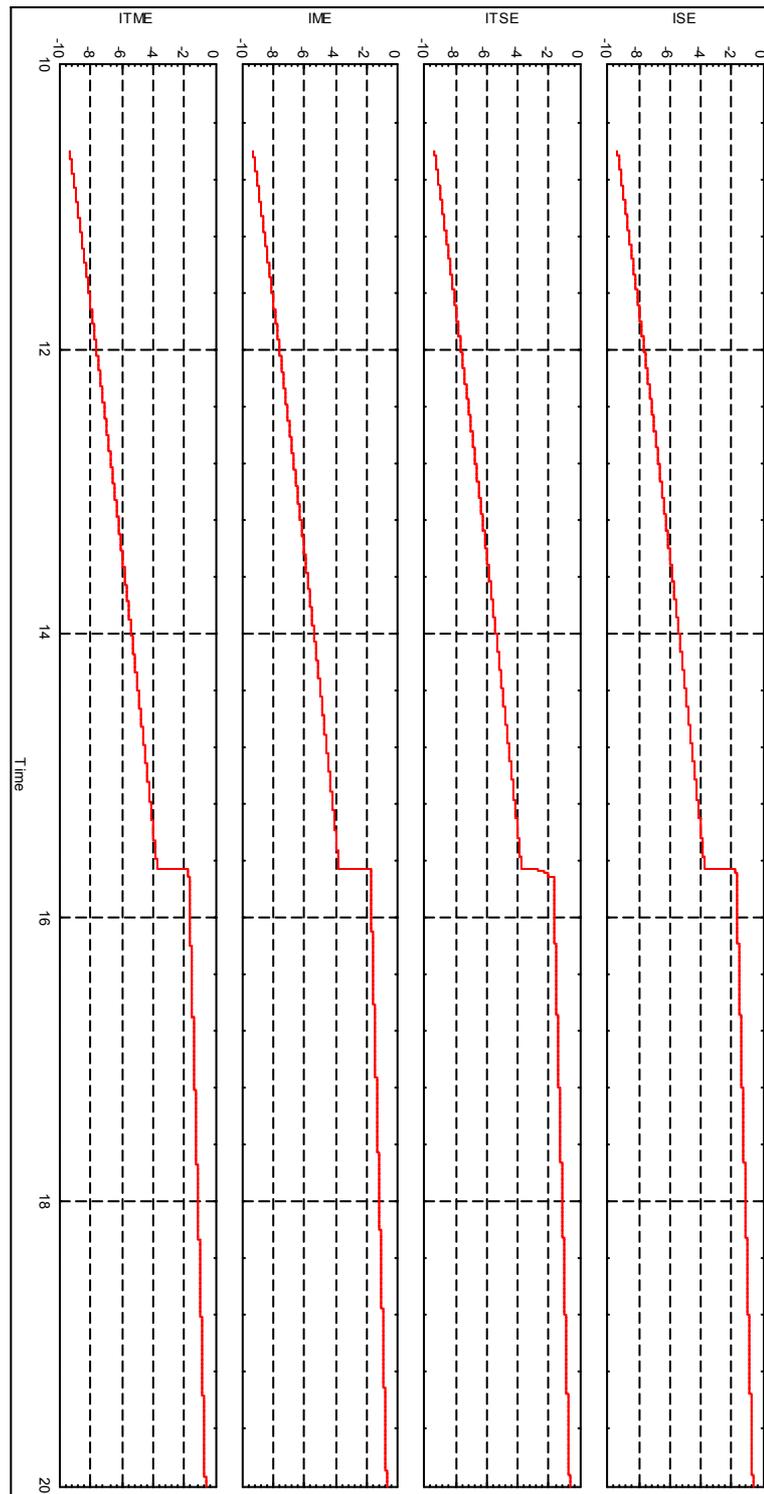


Figura 5-27: comparação entre as transições no modelo de 1 eixo da PMM para os quatro critérios integrais para $J = 10e-5$.

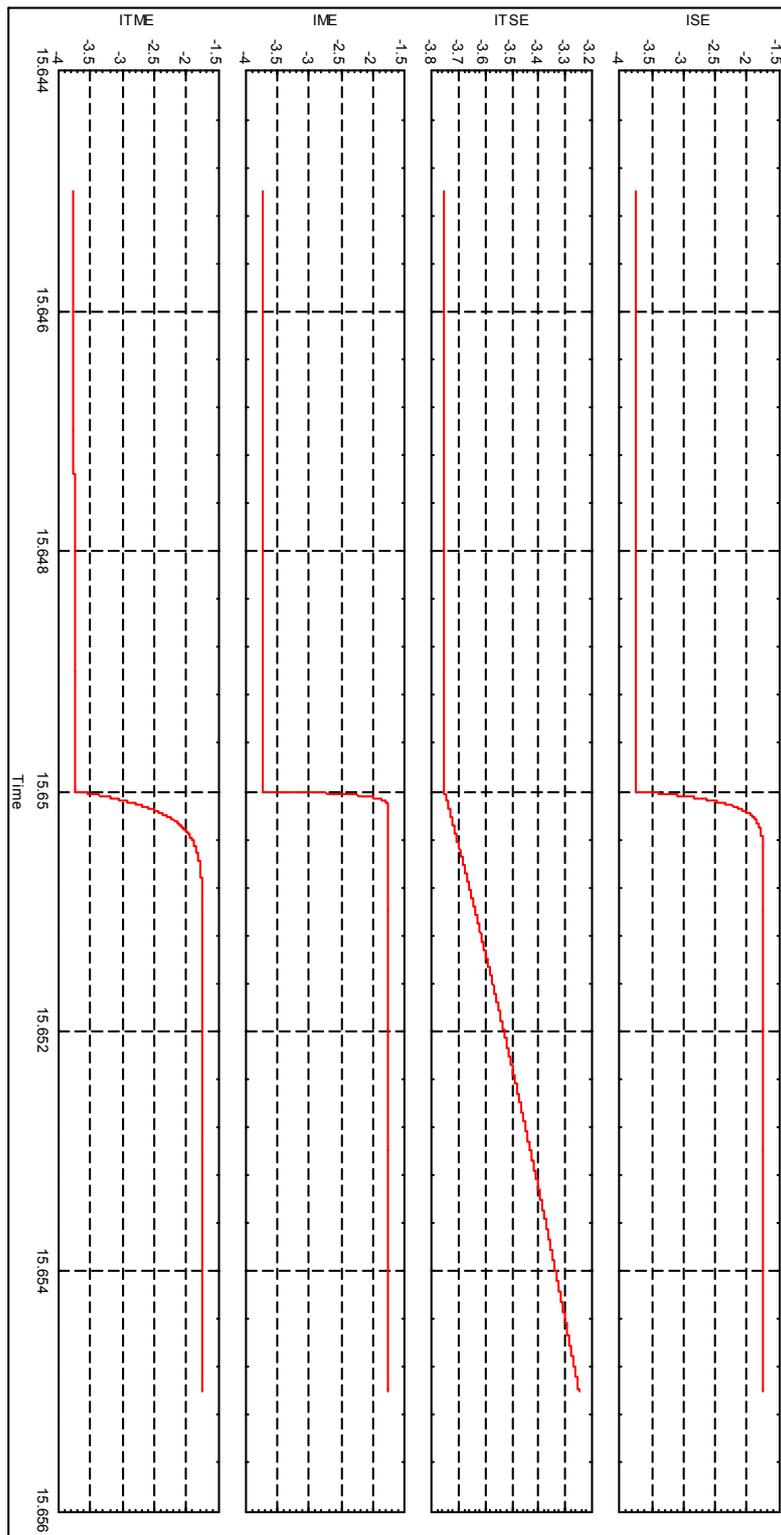


Figura 5-28: comparação entre as transições no modelo de 1 eixo da PMM para os quatro critérios integrais para $J = 10e-5$.

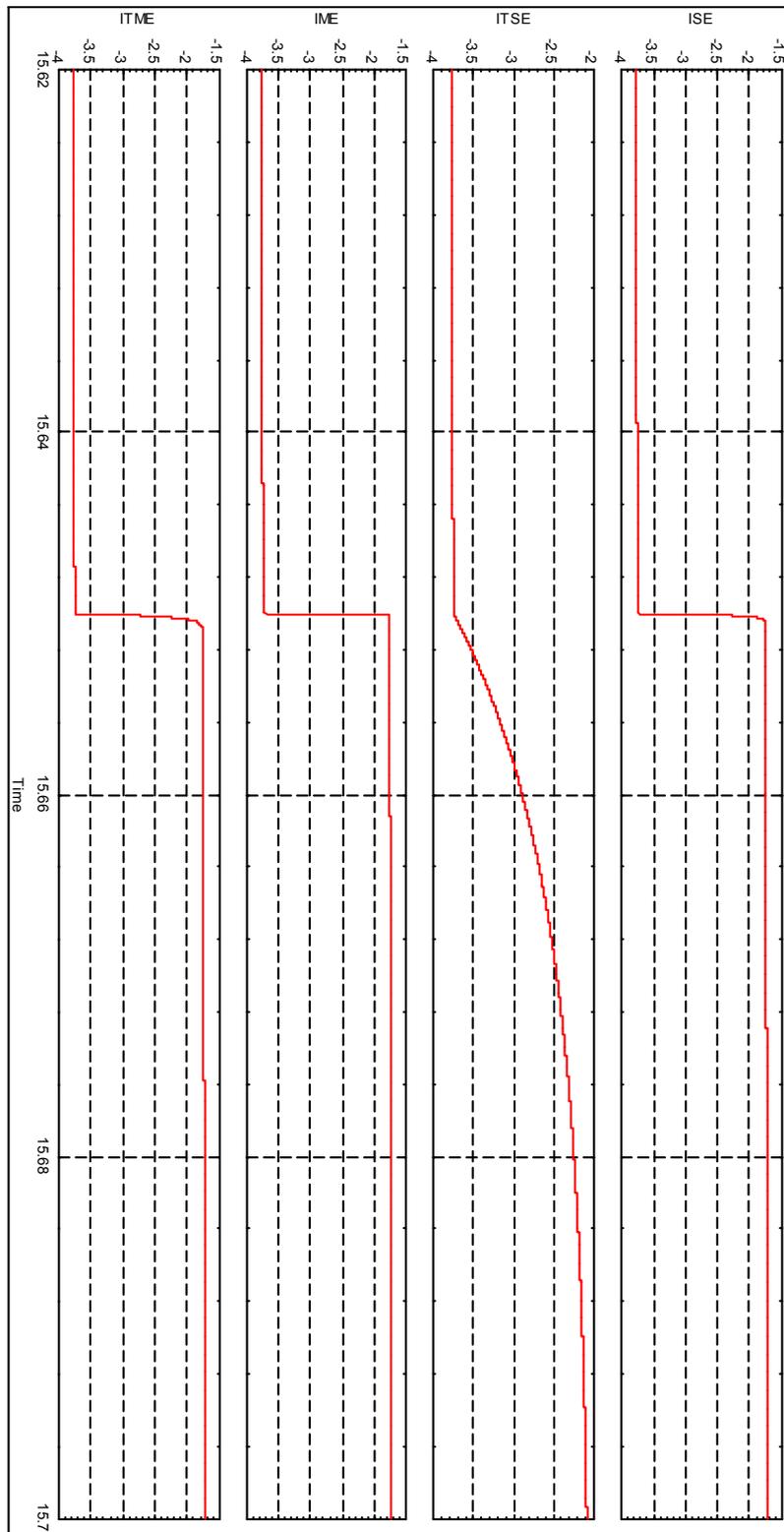


Figura 5-29: comparação entre as transições no modelo de 1 eixo da PMM para os quatro critérios integrais para $J = 10e-5$.

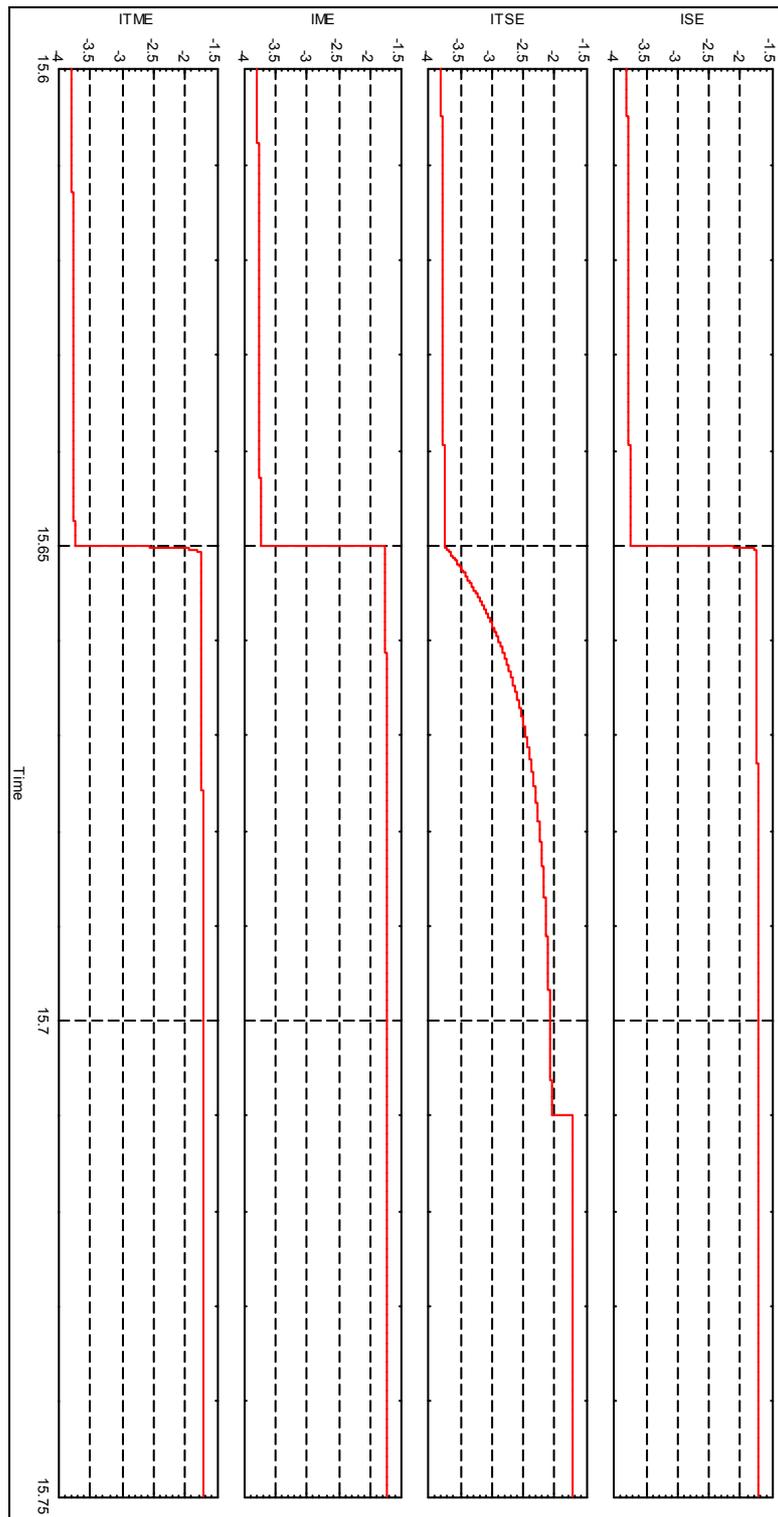


Figura 5-30: comparação entre as transições no modelo de 1 eixo da PMM para os quatro critérios integrais para $J = 10e-5$.

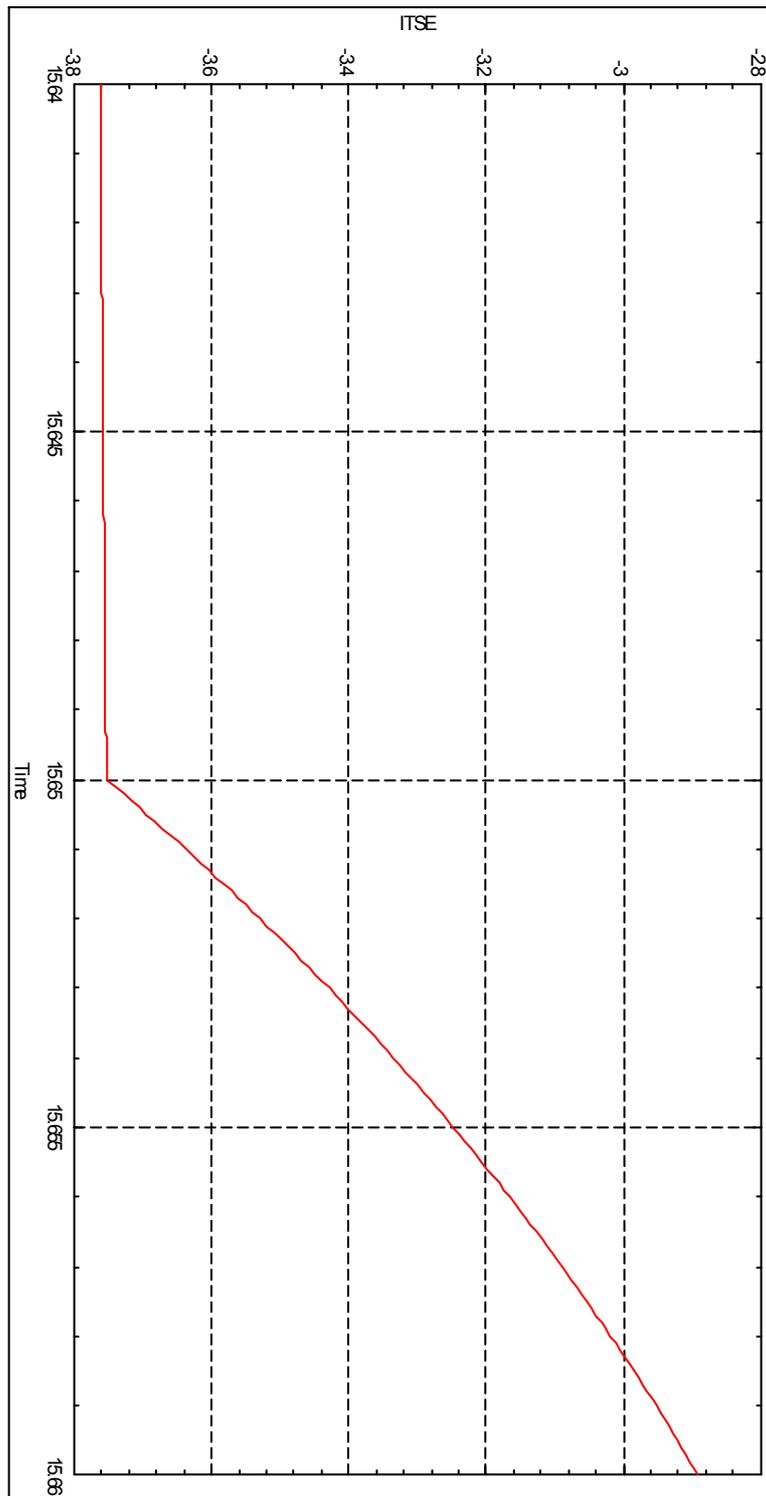


Figura 5-31: Sinal de controle visto pela planta durante uma transição de modos (ITSE).

A Figura 5-32 mostra o sinal de controle efetivo visto pela planta, no caso da PMM, utilizando o critério ITSE. Foi usada uma escala menor que na figura anterior. É possível ver que ainda há um impacto considerável, ao passar do controle transitório para o controle final. O problema não se deve à dinâmica rápida da planta, e sim à velocidade de transição obtida pelo critério. Aqui, o chaveamento do modo de transição para o controlador final foi ativado porque o módulo da primeira derivada do sinal se tornou baixo demais. Há a possibilidade de adiar esse chaveamento até que os dois sinais se aproximem. No entanto, a transição será muito mais demorada.

A Figura 5-33 mostra o sinal de controle efetivo visto pela planta, no caso da PMM, utilizando o critério ITME. A aproximação do sinal de transição para o sinal do controlador final é suave, e não há impacto no chaveamento entre eles. A transição foi obtida em 0,001 s.

A Figura 5-34 mostra o sinal de controle efetivo visto pela planta, no caso da PMM, utilizando o critério ISE. O resultado também é uma aproximação suave de um sinal para o outro. A transição foi obtida em 0,0005 s.

A Figura 5-35 mostra o sinal de controle efetivo visto pela planta, no caso da PMM, utilizando o critério IME. Esta foi a transição mais rápida, e foi obtida em 0,00025 s.

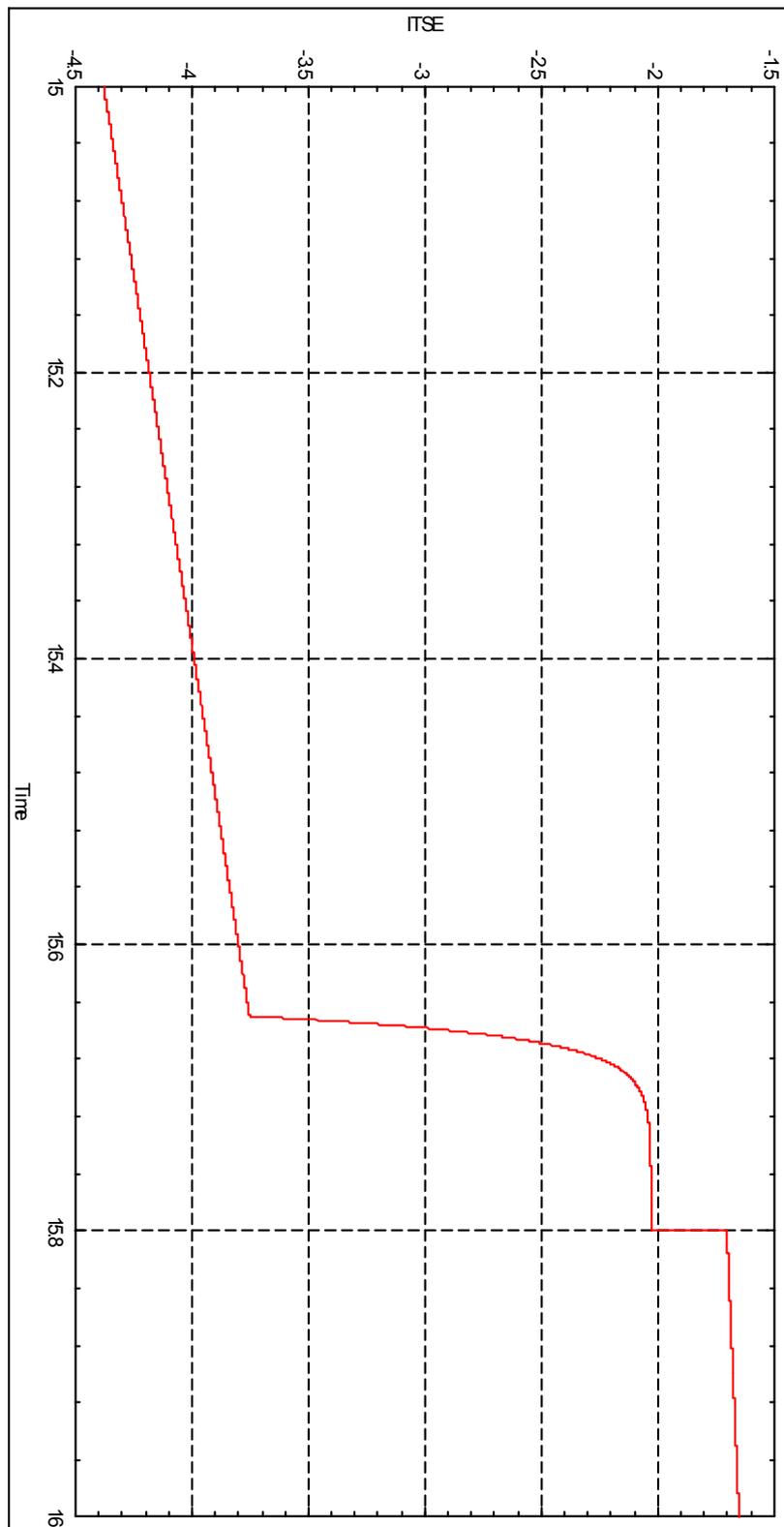


Figura 5-32: Sinal de controle visto pela planta durante uma transição de modos (ITSE).

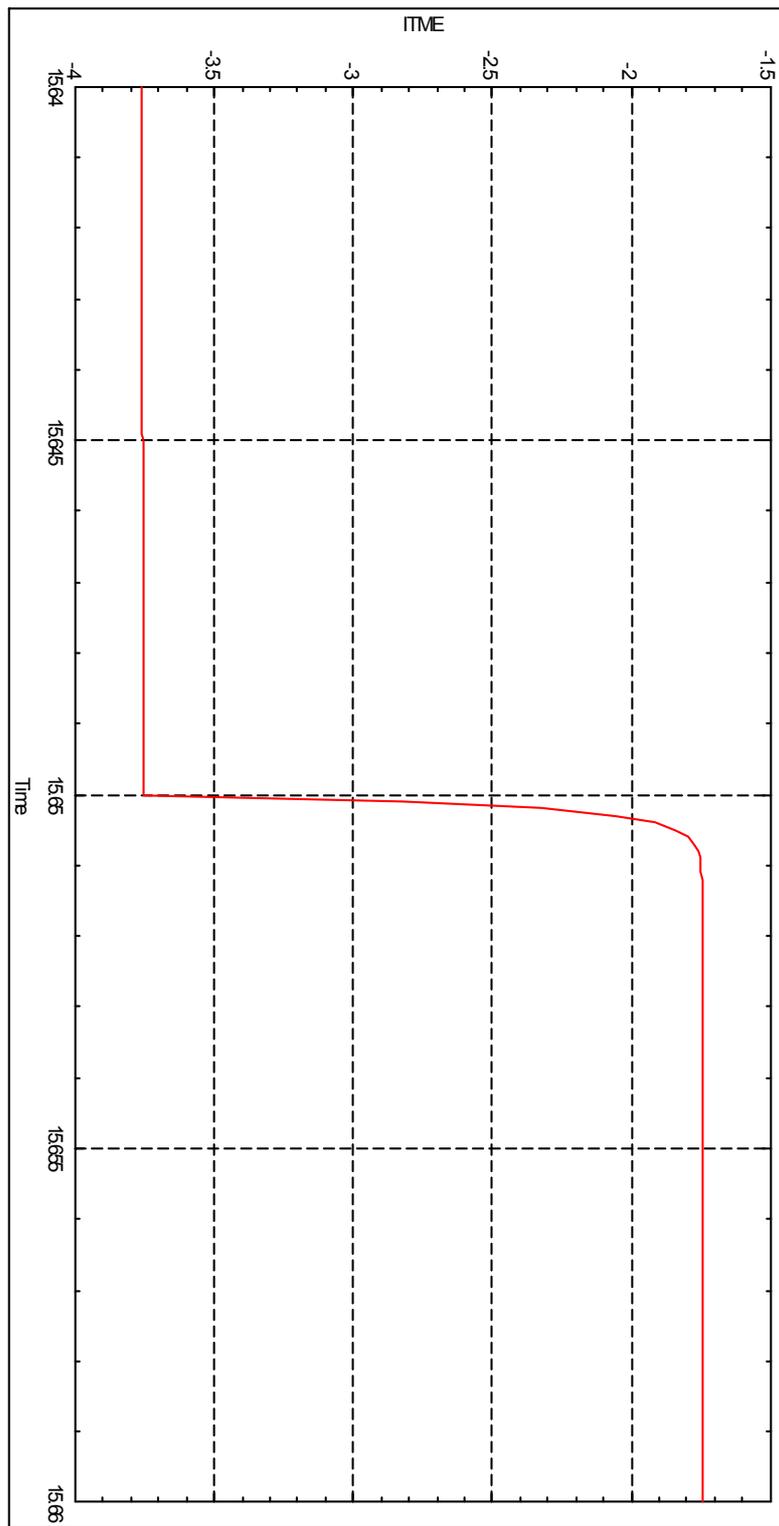


Figura 5-33: Sinal de controle visto pela planta durante uma transição de modos (ITME).

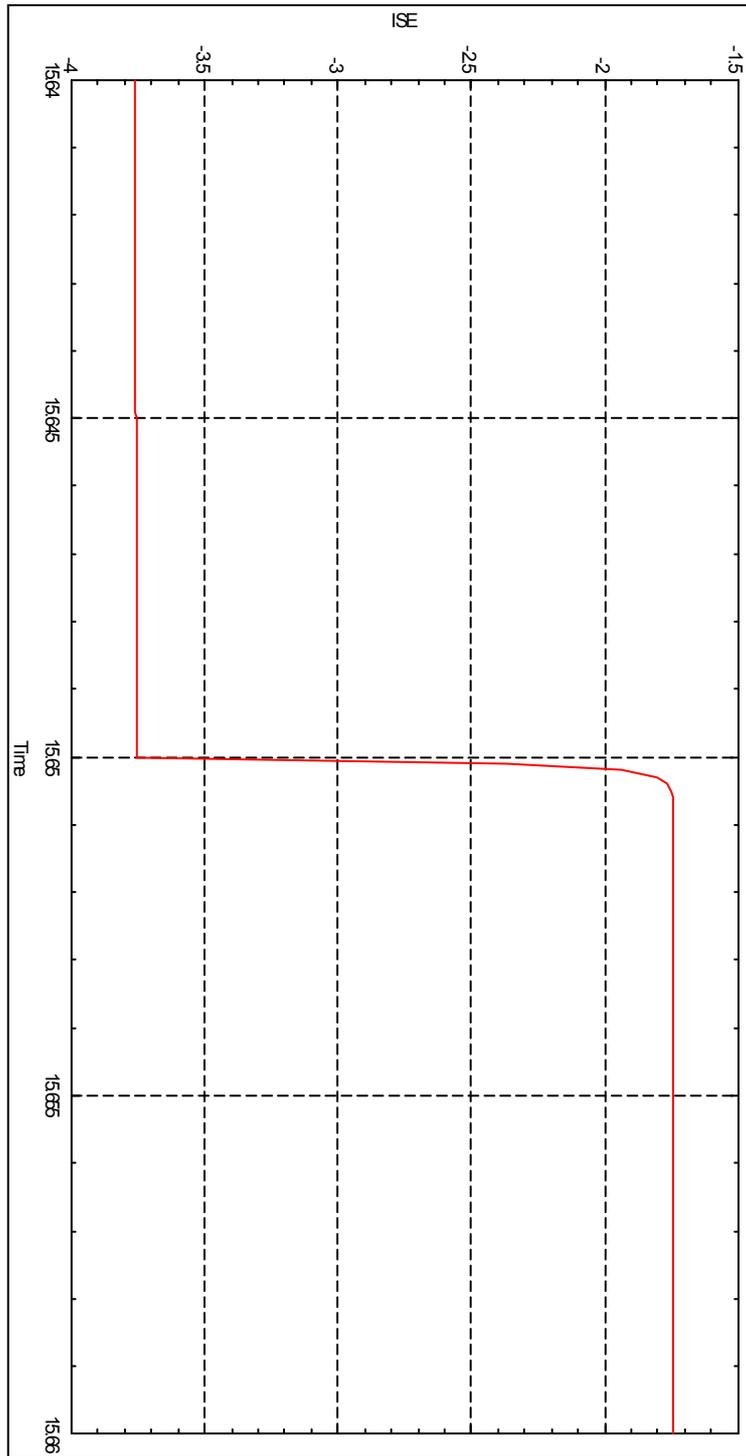


Figura 5-34: Sinal de controle visto pela planta durante uma transição de modos (ISE).

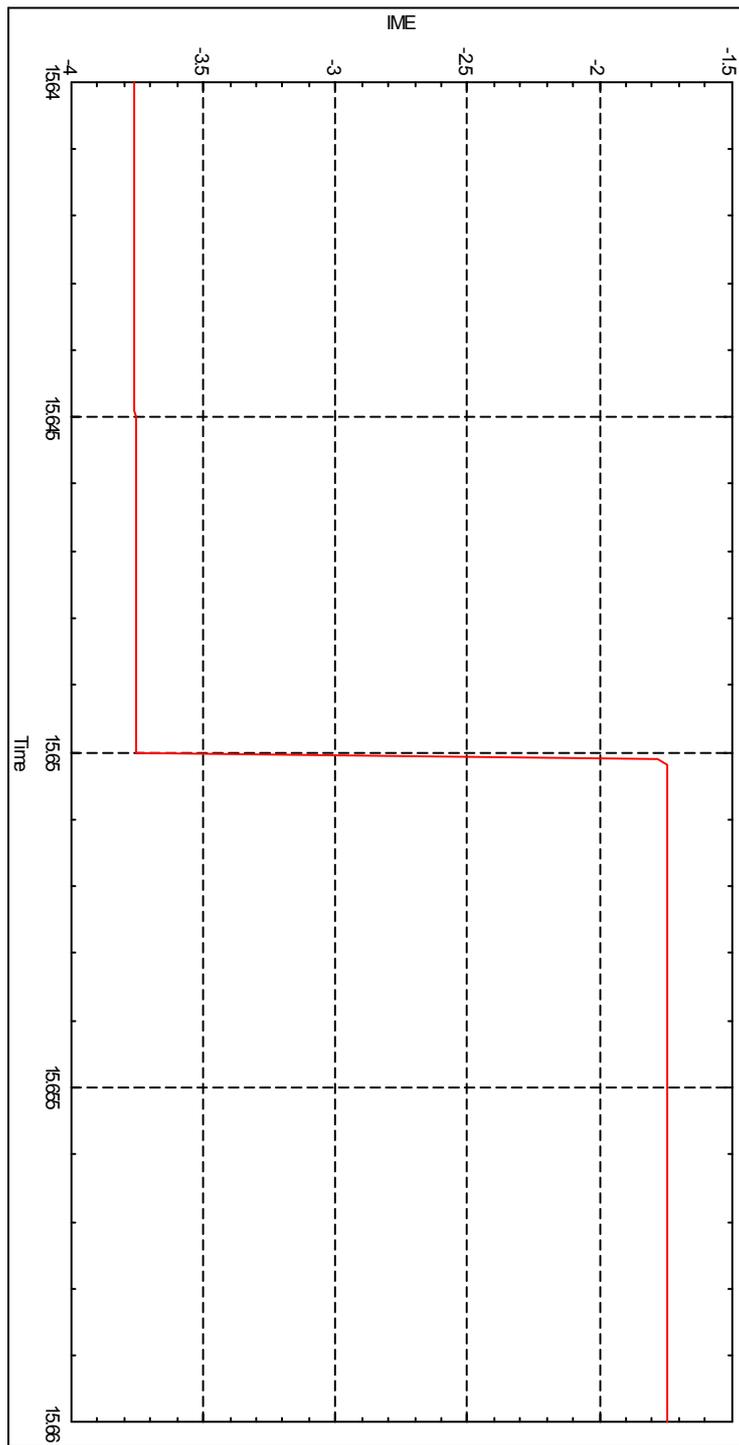


Figura 5-35: Sinal de controle visto pela planta durante uma transição de modos (IME).

5.4.6. Comparação dos Critérios no Modelo dos Motores de Indução.

A Figura 5-36 mostra o sinal de controle durante uma transição de modos, usando o critério ISE para o caso dos motores de indução. Foi arbitrado o valor de $10e-5$ para o J . O valor do menor ζ que satisfaz esta condição foi 227.

A Figura 5-37 mostra o sinal de controle durante uma transição de modos, usando o critério ITSE para o caso dos motores de indução. O valor do menor ζ que satisfaz esta condição foi 11,45. A transição não foi rápida o bastante para evitar um pequeno impacto na transição do controle transitório para o controlador final. Assim como no caso da PMM, este foi o critério que levou mais tempo para executar a transição.

A Figura 5-38 mostra o sinal de controle durante uma transição de modos, usando o critério IME para o caso dos motores de indução. O valor do menor ζ que satisfaz esta condição foi 308,5.

A Figura 5-39 mostra o sinal de controle durante uma transição de modos, usando o critério ITME para o caso dos motores de indução. O valor do menor ζ que satisfaz esta condição foi 18,75. Como este também foi um caso de transição relativamente lenta (0,07 s), o sinal de controle final já não estava mais no local registrado pelo bloco administrador no momento da transição, e houve um pequeno impacto.

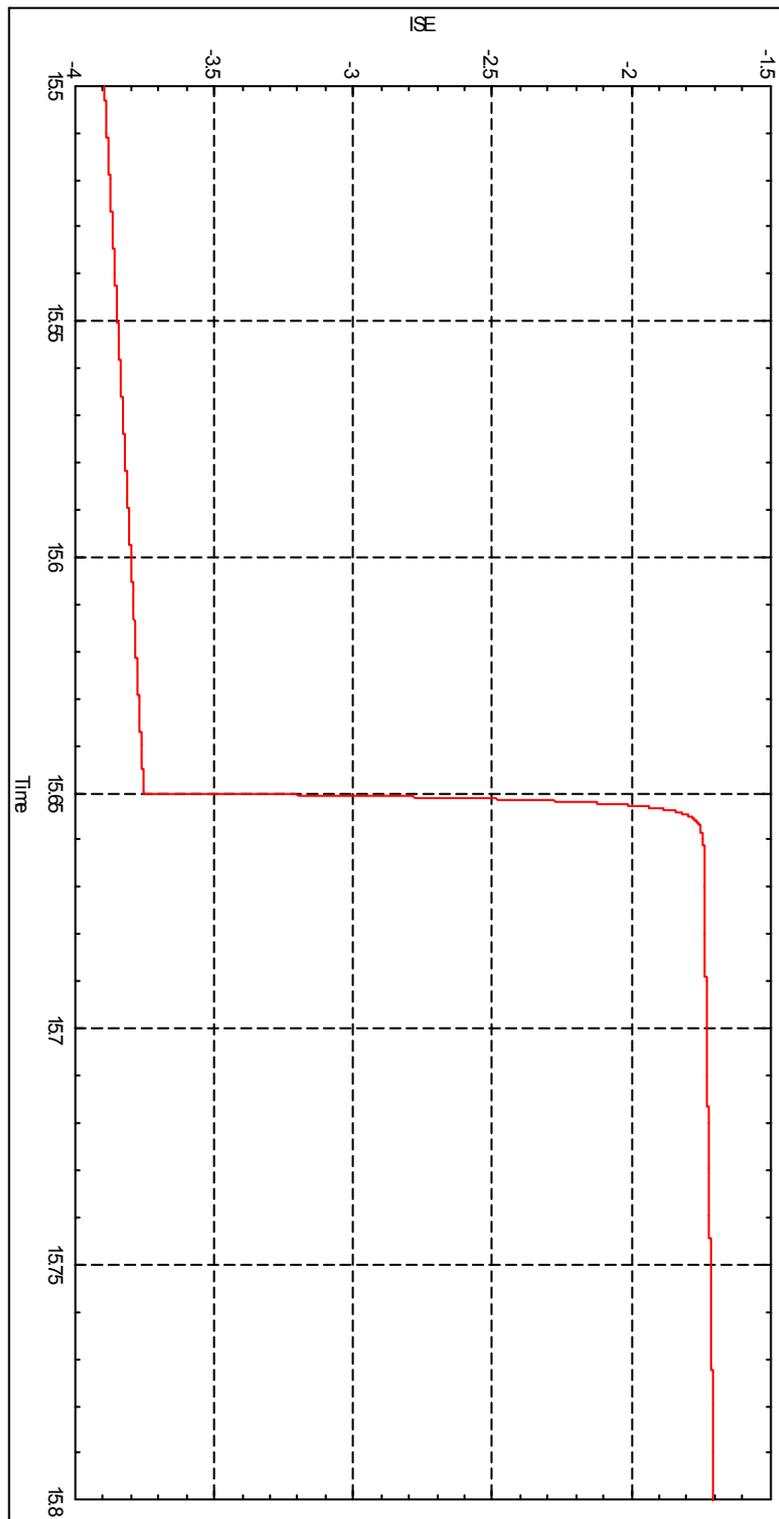


Figura 5-36: Sinal de controle visto pela planta durante uma transição de modos (ISE).

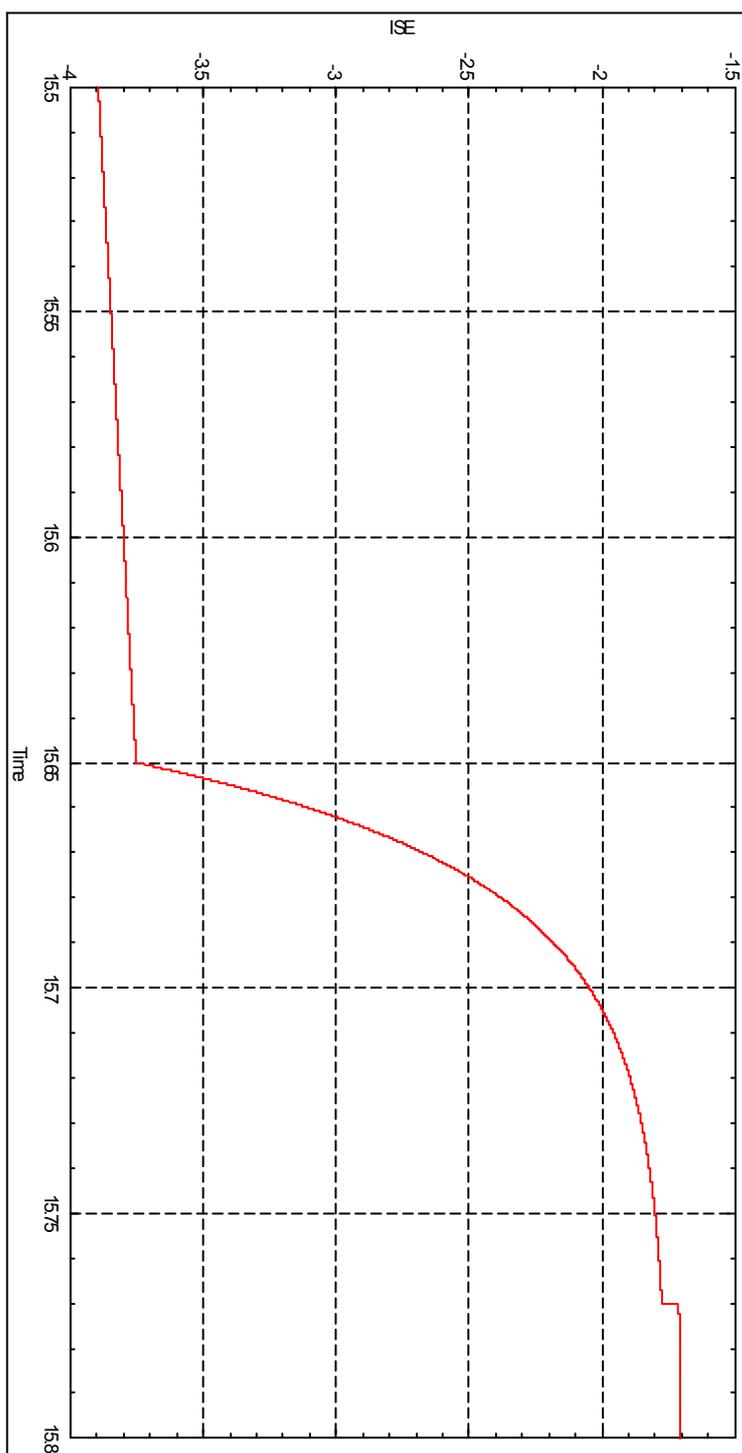


Figura 5-37 Sinal de controle visto pela planta durante uma transição de modos (ITSE).

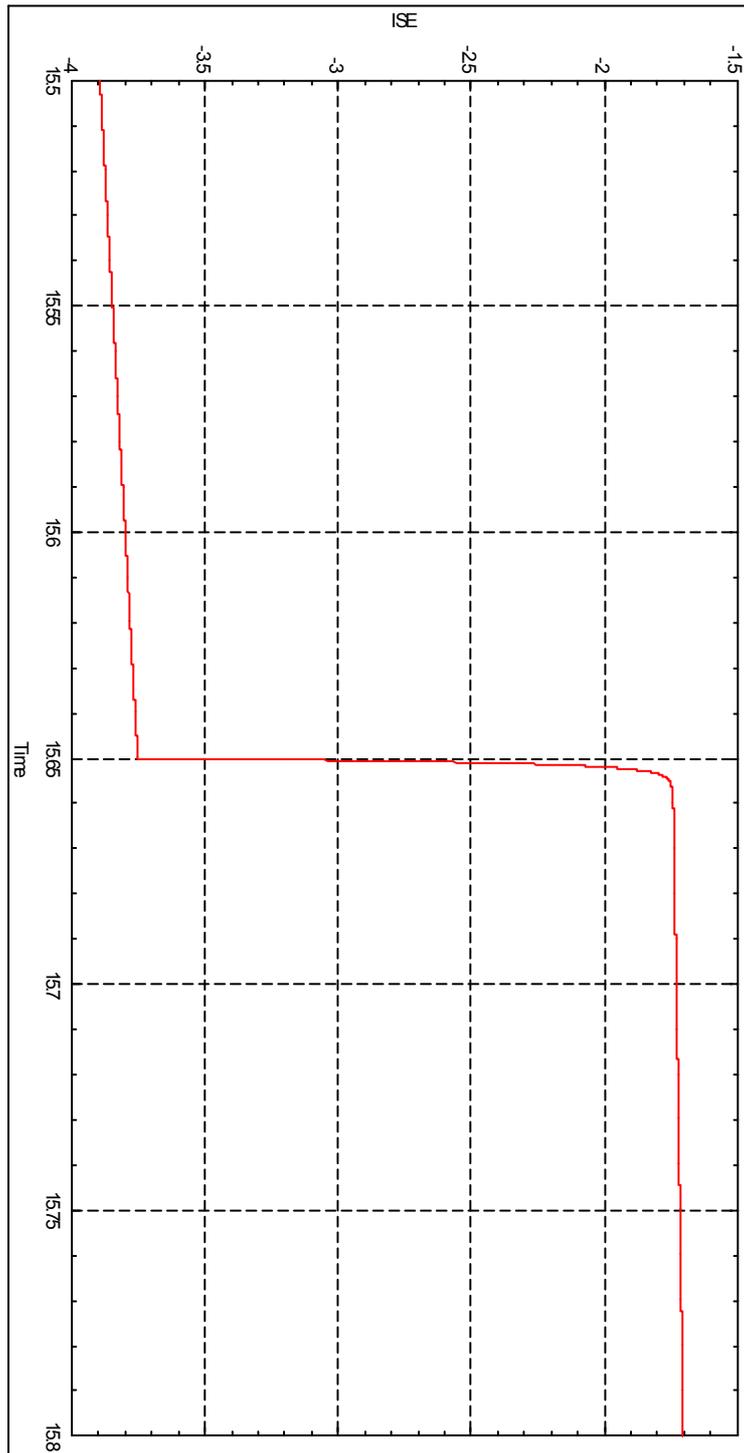


Figura 5-38: Sinal de controle visto pela planta durante uma transição de modos (IME).

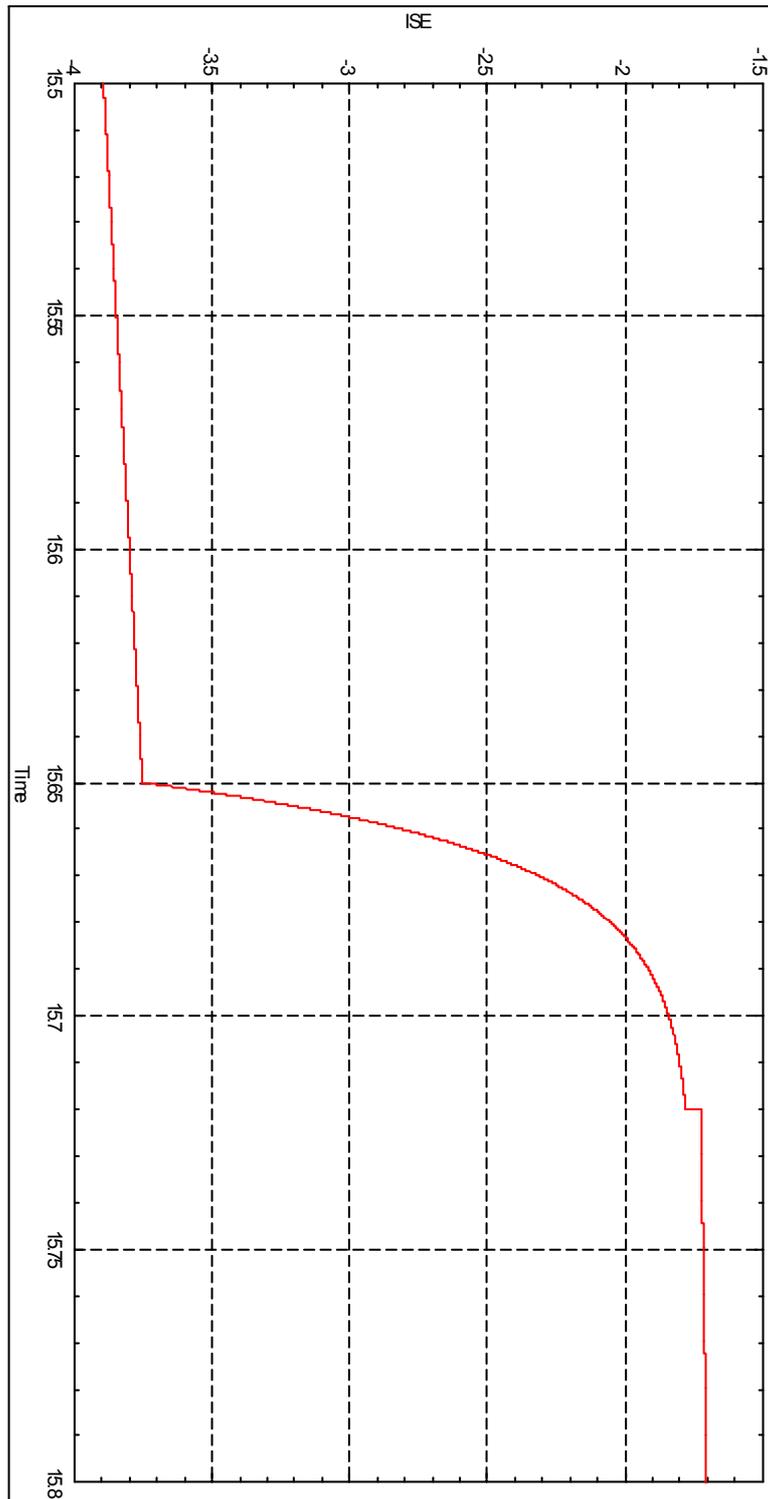


Figura 5-39: Sinal de controle visto pela planta durante uma transição de modos (ITME).

6 ESTUDO DE ESTABILIDADE SOBRE A RECONFIGURAÇÃO NO MODELO DA PMM

Neste Capítulo, apresentaremos os resultados da análise da estabilidade geral durante o chaveamento entre dois modos por “cross-fading”, utilizando os métodos descritos nas Sessões 4.2.6.1 e 4.2.6.2.

Usamos a expressão que representa o sistema durante uma transição de modos em “cross-fading”, em função da duração da transição, Δt . O controlador final é o mesmo PID usado em simulações da PMM ($k_p = 40,593$; $k_i = 1$, $k_d = 454,11$) e o controle inicial é uma versão com o coeficiente 5 vezes maior ($k_p = 40,593$; $k_i = 5$; e $k_d = 454,11$), ambos estáveis isoladamente. Atribuímos valores a Δt entre 0,01 e 100 segundos e observamos como os polos e zeros se posicionaram para checar a fidelidade e utilidade da representação.

Para cada duração Δt de intervalo, foi plotado um gráfico de polos e zeros no MatLab, para análise da sua influência na Função-Transferência.

Esta sessão mostra o comportamento dos polos do sistema em malha fechada, com o modelo da PMM de 1 eixo usando rodas de reação. Como exemplo, executamos a transição entre dois PIDs de coeficientes ($k_{p1} = 40,593$; $k_{i1} = 5$; $k_{d1} = 454,11$) para ($k_{p2} = 40,593$; $k_{i2} = 1$; $k_{d2} = 454,11$).

6.1. “Cross-fading” Linear Ilimitado

As Figuras de 6-1 a 6-6 mostram os gráficos de polos e zeros (H1 em vermelho) obtidos pela modelagem do “cross-fading” linear ilimitado, com diferentes tempos de duração. Para servir de comparação, também foram incluídos, em todos os gráficos, os polos e zeros do sistema completo com os controles inicial (H1 em azul) e final (H2 em verde) isoladamente.

A Figura 6-1 refere-se a uma transição com duração de 100 segundos. Observa-se que os polos tendem a coincidir com os polos da planta no estado inicial, e isso está de acordo com uma transição muito mais lenta do que a planta.

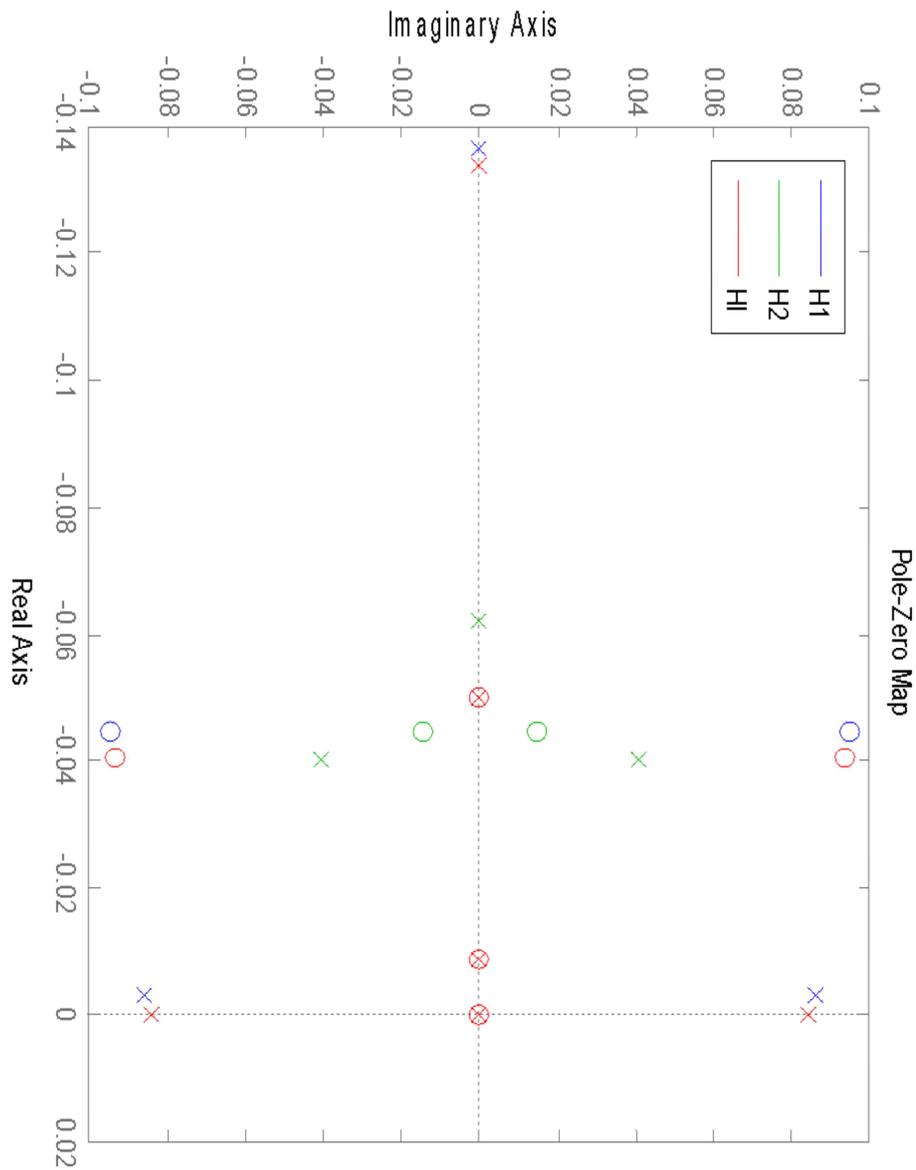


Figura 6-1: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 100$ s.

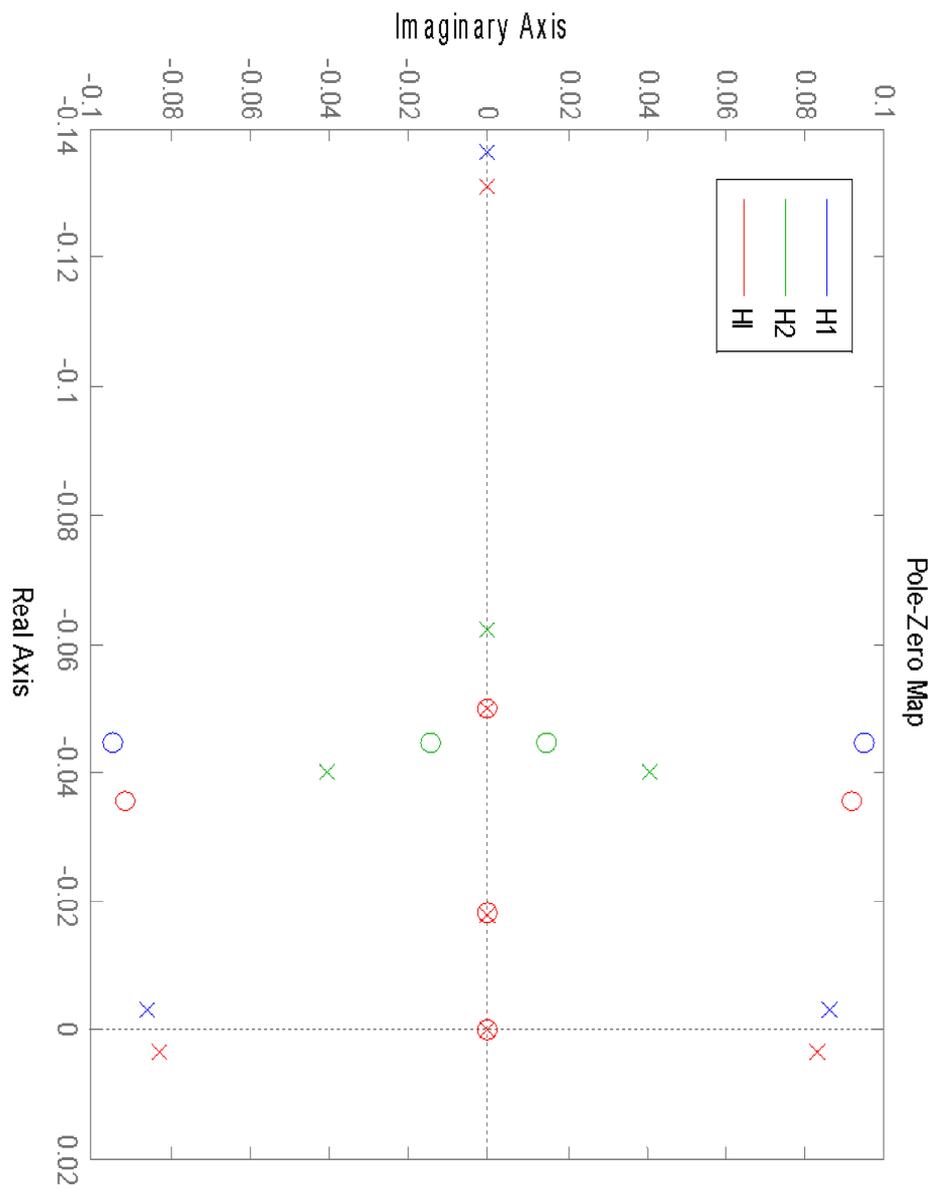


Figura 6-2: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 50$ s.

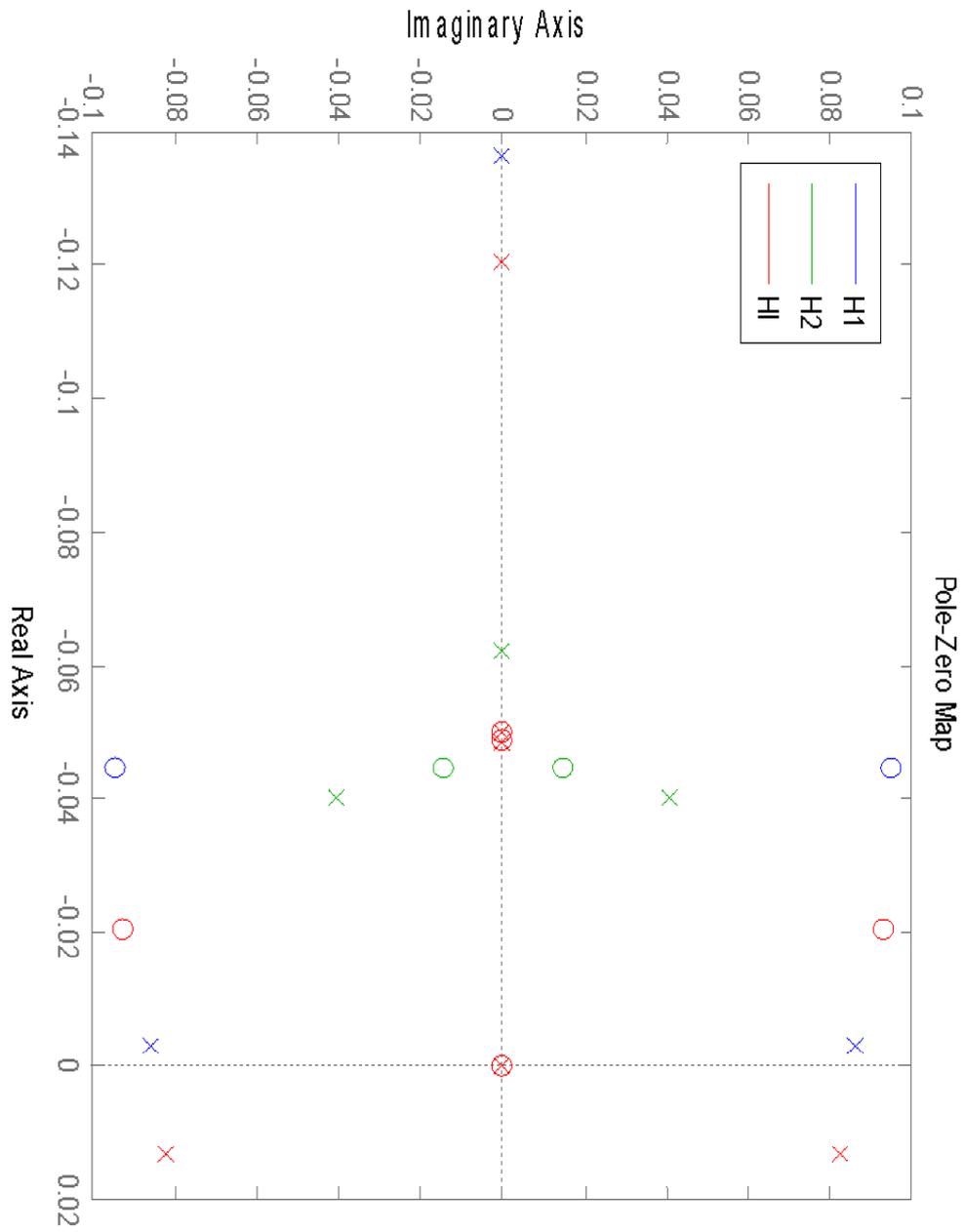


Figura 6-3: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 20$ s.

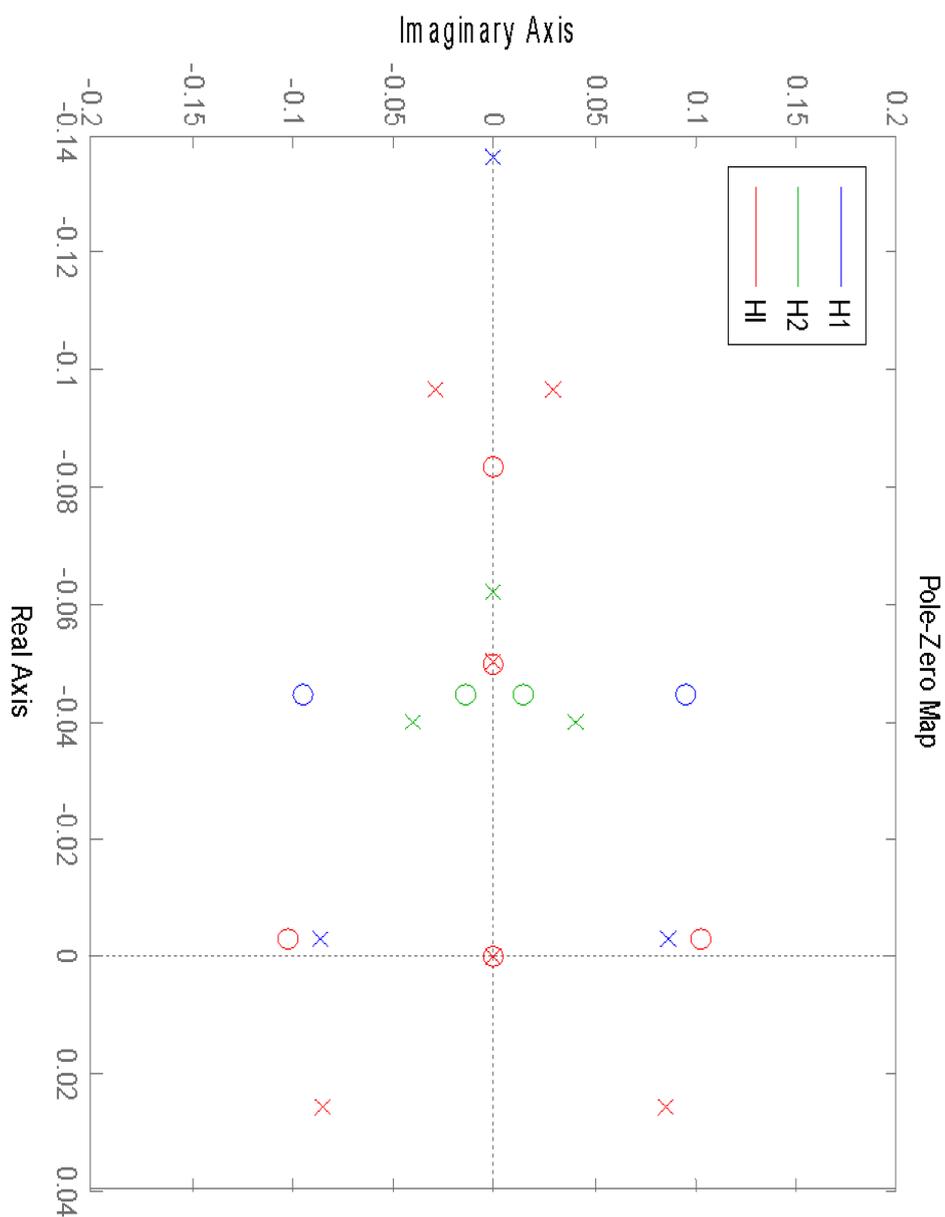


Figura 6-4: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 10$ s

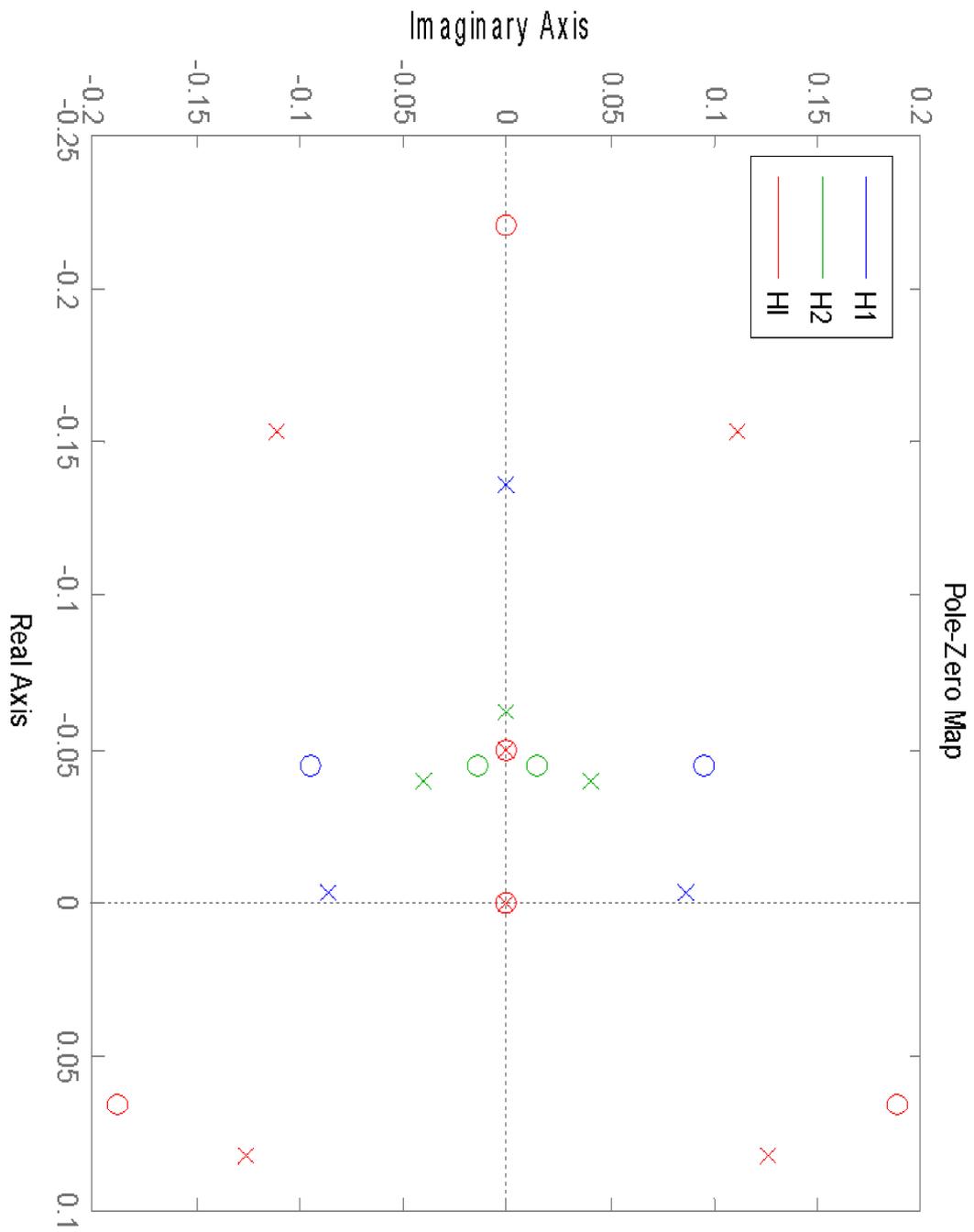


Figura 6-5: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 1$ s.

A Figura 6-6 mostra que, à medida que a transição se torna mais rápida, os polos não migram para perto dos polos do estado final da planta. Pelo contrário, alguns deles se deslocam para o semiplano direito (indicado em setas vermelhas). Isso é uma falha esperada da aproximação, que não modela o encerramento do “cross-fading”.

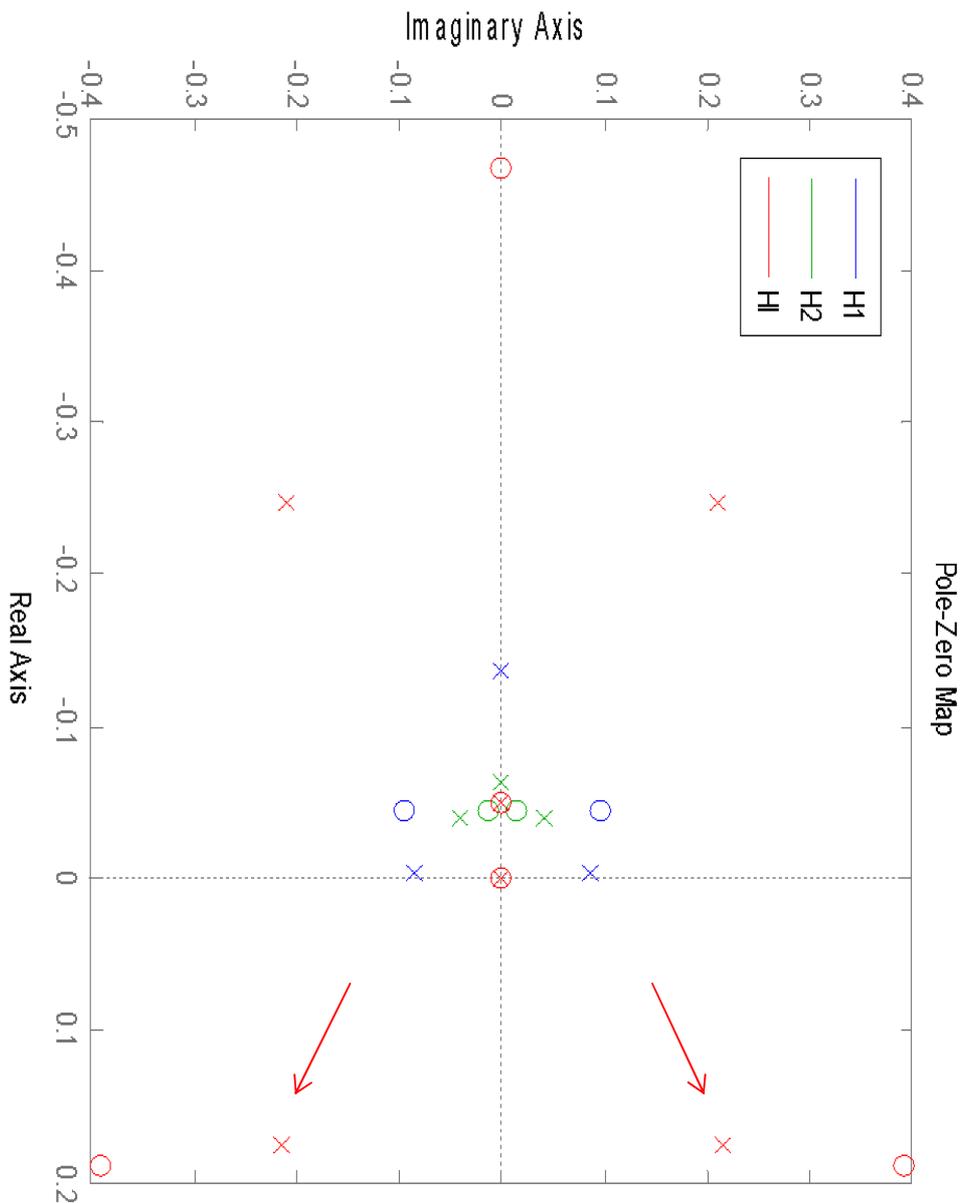


Figura 6-6: Diagrama de polos e zeros com “cross-fading” linear do modelo da PMM com $\Delta t = 0,1$ s.

6.2. “Cross-fading em Decaimento Exponencial

As Figuras 6–7 a 6–14 mostram os gráficos de polos e zeros (He em vermelho) obtidos pela modelagem do “cross-fading” linear ilimitado, com diferentes tempos de duração. Assim como no caso linear, também foram incluídos, em todos os gráficos, os polos e zeros do sistema completo com os controles inicial (H1 em azul) e final (H2 em verde) isoladamente.

A Figura 6-7 mostra que, assim como no caso linear, quanto mais lenta a transição, mais os polos do modelo em “cross-fading” tendem a coincidir com os polos da planta no estado inicial.

A Figura 6-8 mostra que, numa transição lenta, os polos na extrema esquerda ficam perto dos polos do sistema com controle final (H2 verde), indicando que o impacto é lento em relação ao resto do comportamento da planta.

A Figura 6-12 mostra que, à medida em que a transição se torna mais rápida, os polos do sistema em “cross-fading” (He em verde) tendem a coincidir com os polos da planta final (H2 em vermelho).

A Figura 6-13 mostra que, numa transição rápida, vários polos do sistema em “cross-fading” migram para o extremo esquerdo do plano, indicando um transiente rápido. De uma forma geral, a planta da PMM é lenta demais para ser um bom caso de testes.

O “cross-fading” exponencial introduz um impacto maior, com transições mais rápidas que seu equivalente linear de mesma duração, porém é mais simples de modelar. Se a análise por “cross-fading” exponencial indicar uma instabilidade, pode-se presumir que o “cross-fading” linear também será estável. Conclui-se que a modelagem pelo decaimento exponencial pode ser uma condição suficiente mas não necessária (majorante) para garantir uma transição estável.

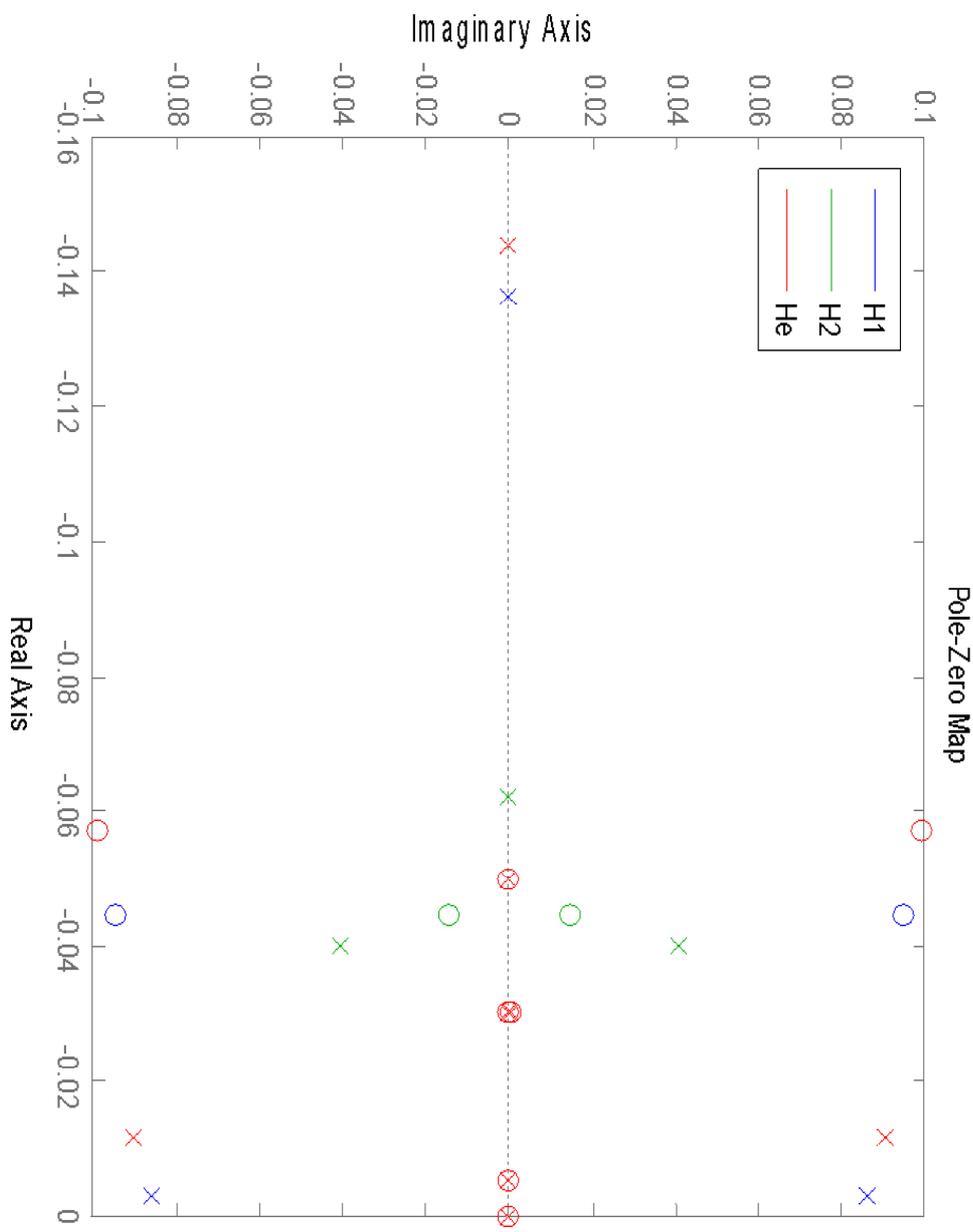


Figura 6-7: Polos do sistema da PMM durante “cross-fading” exponencial $\Delta t = 100$ s.

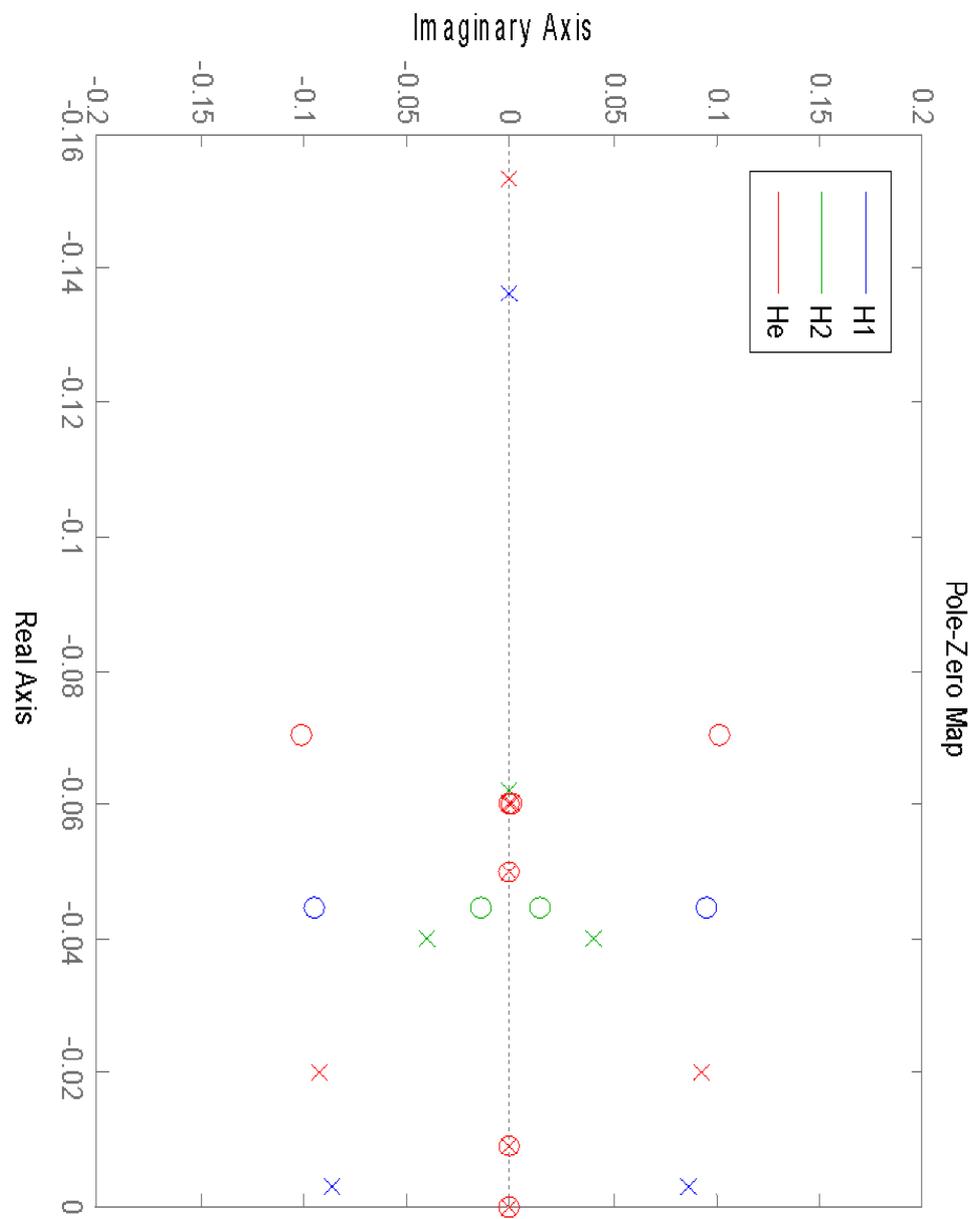


Figura 6-8: PMM exponencial $\Delta t = 50$ s.

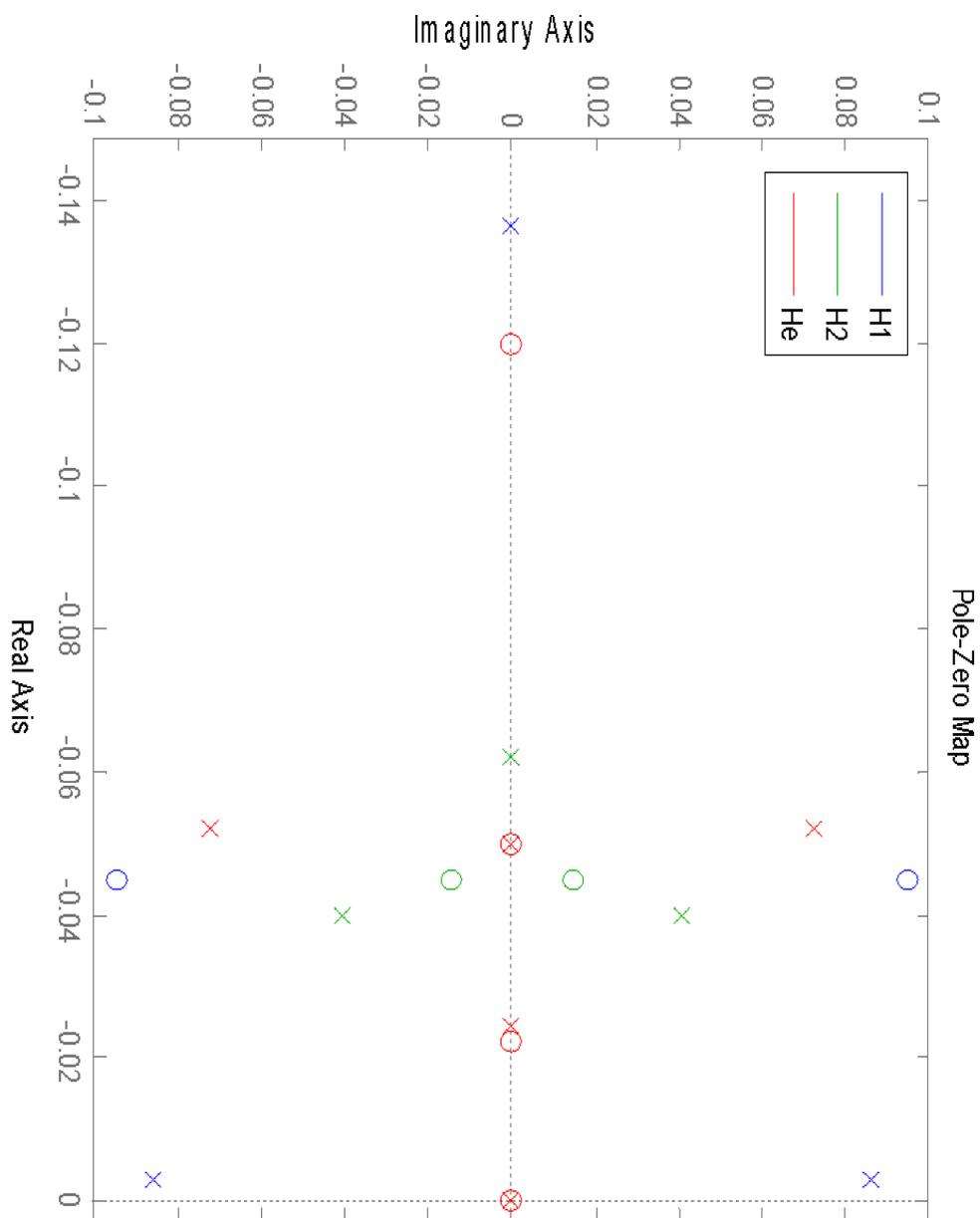


Figura 6-9: PMM exponencial $\Delta t = 10$ s.

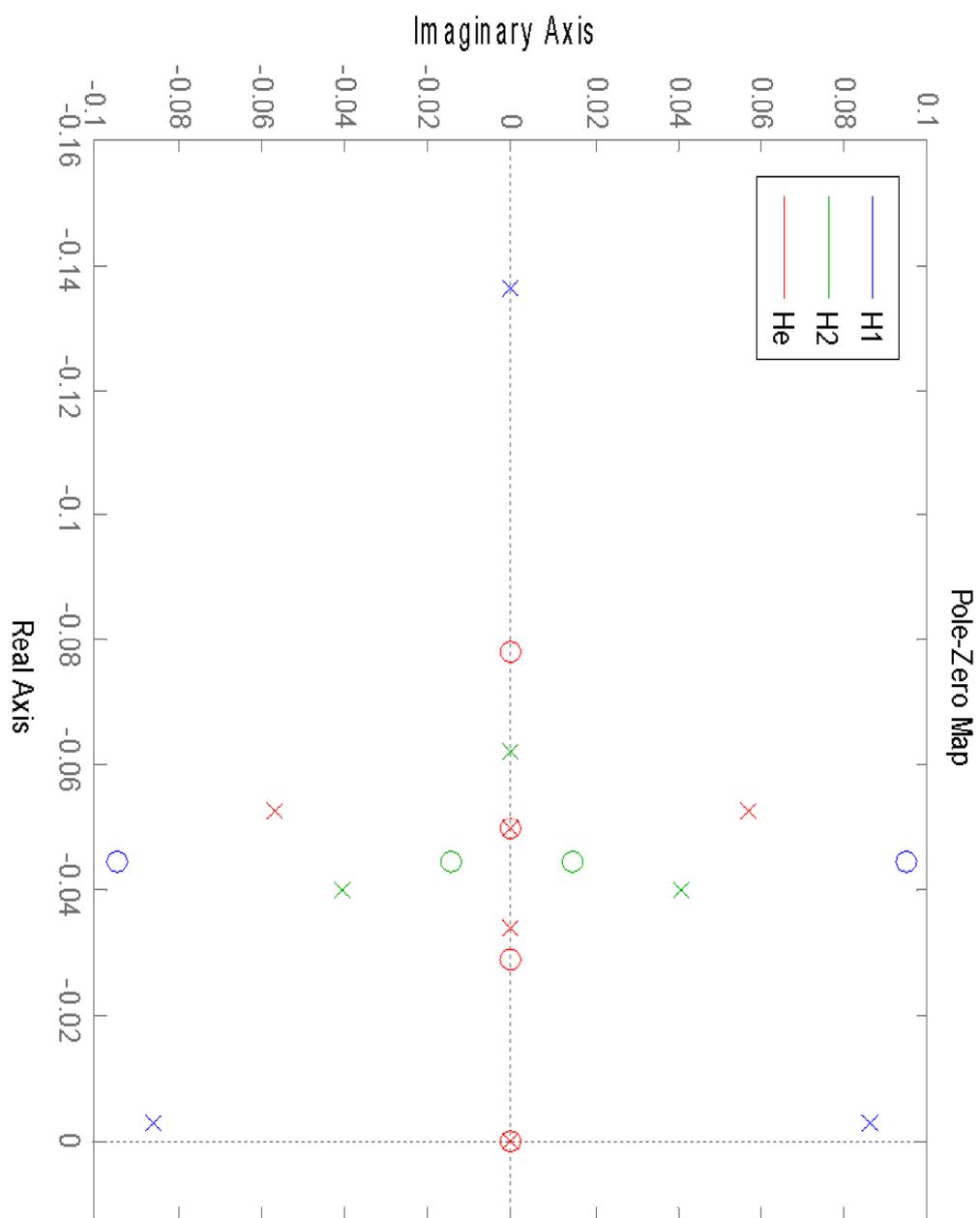


Figura 6-10: PMM exponencial $\Delta t = 5$ s.

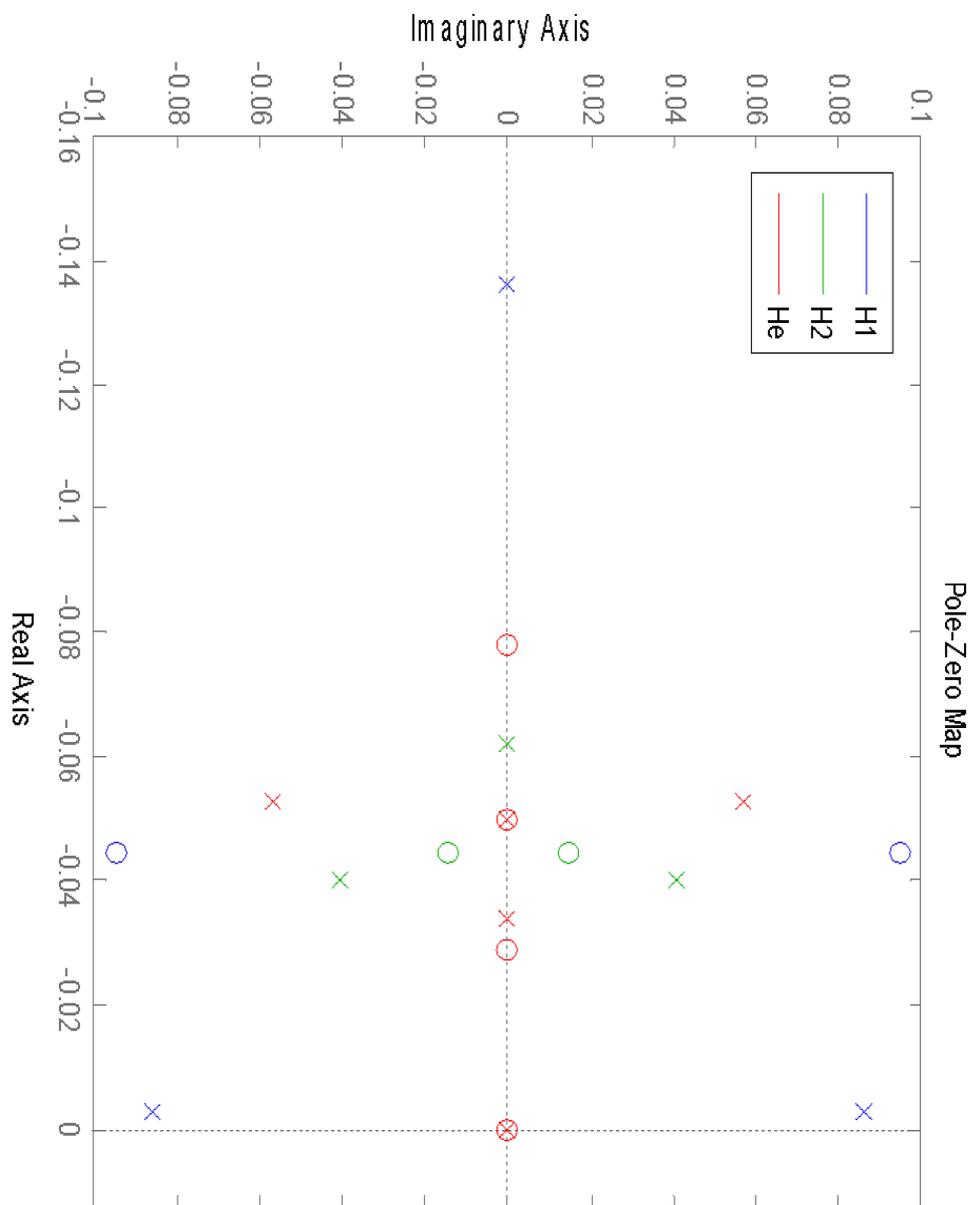


Figura 6-11: PMM exponencial $\Delta t = 1$ s.

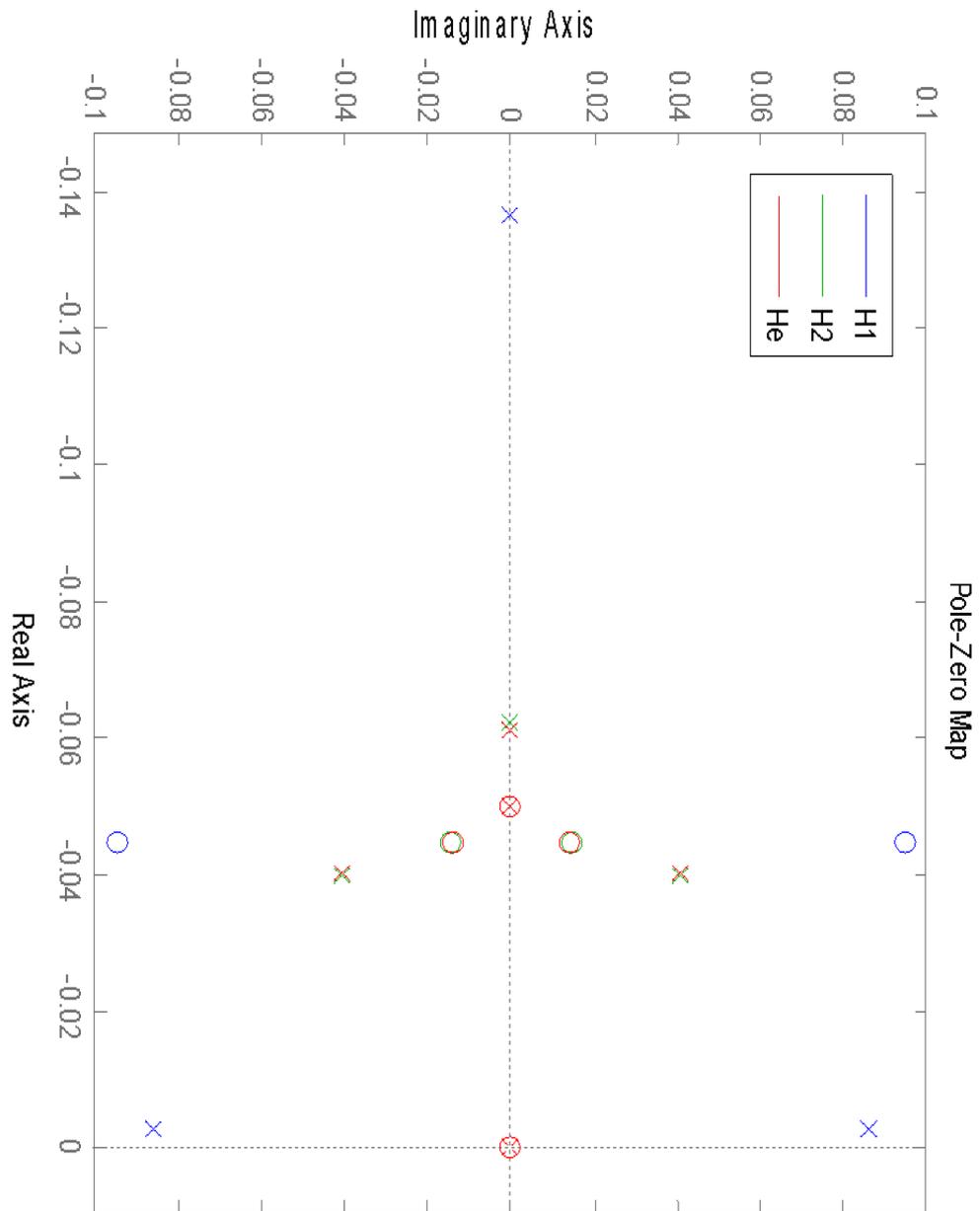


Figura 6-12: PMM exponencial $\Delta t = 0,1$ s.

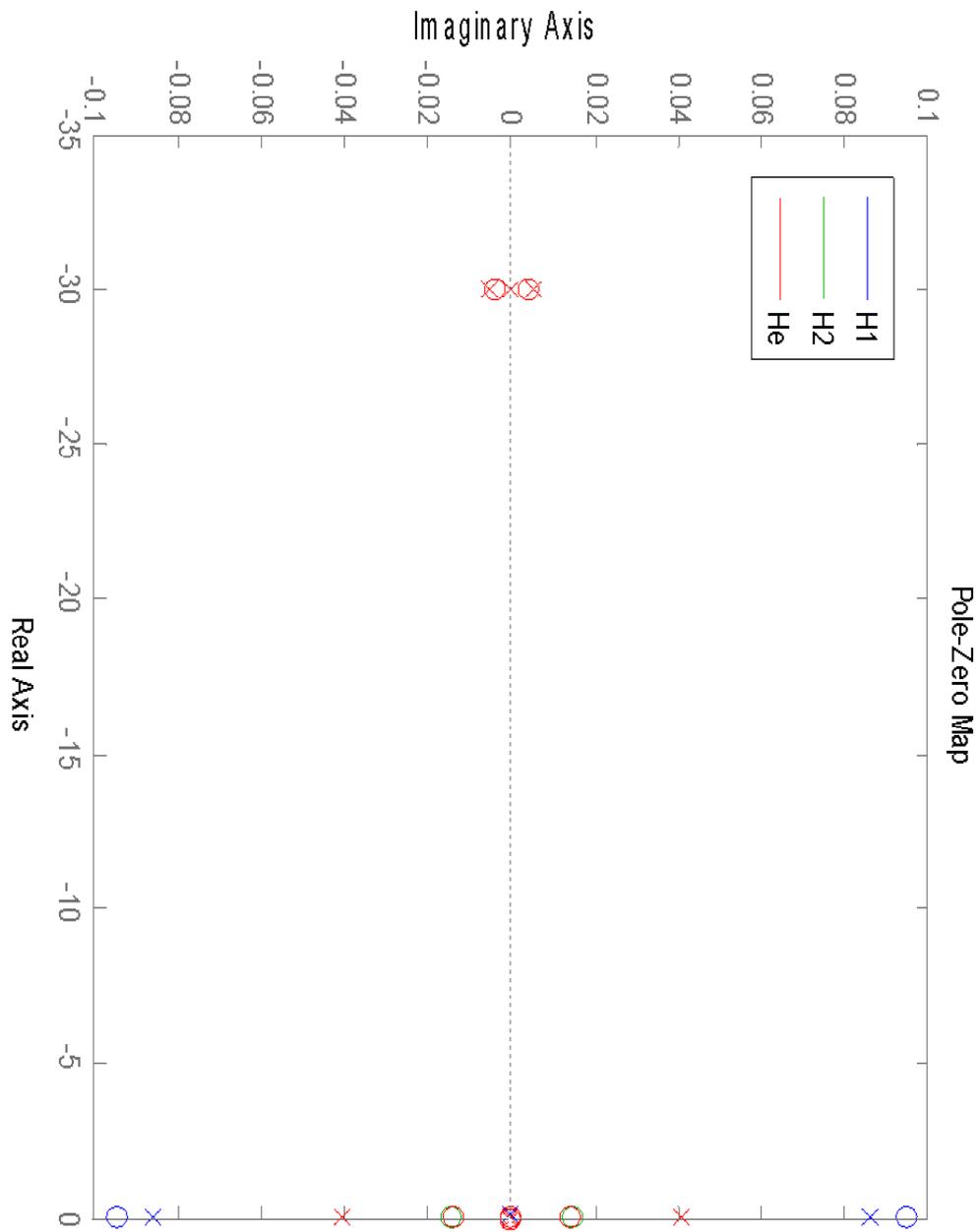


Figura 6-13: PMM exponencial $\Delta t = 0,1$ s, mostrando grupos de polos mais à esquerda, associados ao impacto da transição.

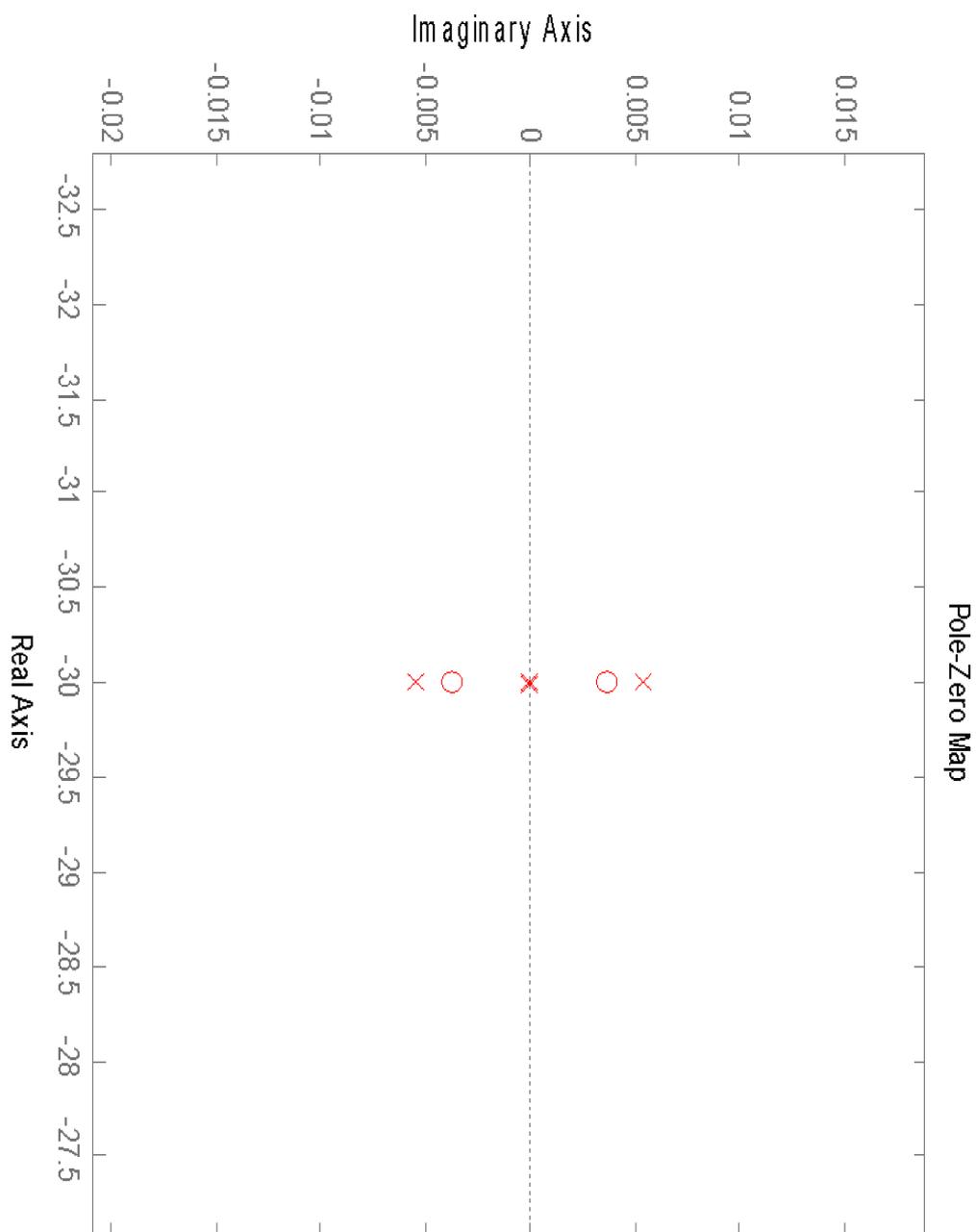


Figura 6-14: PMM exponencial Delta t = 0,1 s, detalhe no grupos de polos mais à esquerda, associados ao impacto da transição.

7 CONCLUSÕES E SUGESTÕES

7.1 CONCLUSÕES

- **Como contribuições originais:**
- Foi possível melhorar o desempenho do transitório com abordagens de “anti-windup”; “cross-fading”; “rastreamento de saída” em dois casos de estudo: PMM e 2 motores CC.
- Foi possível obter uma nova aplicação dos critérios integrais ISE, ITSE, IAE e ITAE no modelo da PMM e dos dois motores de indução, para determinar trajetórias de transição de sinais de controle durante um chaveamento entre blocos de controle.
 - Devido à forma como foi implementado, os critérios integrais indicaram que o valor ótimo ocorre quando o parâmetro tende a infinito, o que não pode ser obtido. No entanto, os gráficos permitem atribuir um valor qualitativo à transição, de forma mais abrangente do que critérios pontuais como sobressinal ou tempo de acomodação. Quando se estabelece um valor mínimo aceitável, os critérios integrais permitem encontrar um parâmetro que o satisfaça.
- Foi possível obter novas abordagens para o estudo da estabilidade:
 - A plotagem sequencial dos polos:
 - O “cross-fading” linear introduz polos falsos que podem ser desprezados quando são muito mais lentos do que a transição.
 - O “cross-fading” em decaimento exponencial, que é matematicamente exata e pode ser usada para garantir o comportamento da versão linear.

- A convolução no domínio da frequência, para modelar a ponderação no tempo de sinais de controle durante uma transição entre eles.

7.2 SUGESTÕES PARA TRABALHOS FUTUROS

- Aplicar as técnicas aqui apresentadas, melhoradas ou novas, a outras reconfigurações da PMM e do caso dos 2 motores CC.
- Aplicar as técnicas aqui apresentadas, melhoradas ou novas, a outras reconfigurações dos casos estudados por Liberzon e Diallo.
- Pesquisar novas formas de implementação do bloco de controle transitório, de forma que o cálculo dos critérios integrais tenha mínimos globais, e ainda sejam viáveis de calcular em tempo real.
- Investigar as propriedades do “cross-fading” linear, para determinar com exatidão quais polos positivos podem ser atribuídos à aproximação, e quais outras características podem ser fielmente aproveitadas como uma informação sobre a estabilidade geral do sistema.
- Explorar o uso do “cross-fading” exponencial e da convolução no domínio da frequência como forma de modelar exatamente a reconfiguração de sistemas de controle.

REFERÊNCIAS BIBLIOGRÁFICAS

AIZERMAN, M. A.; GANTMACHER, F. R. *Absolute stability of regulator systems*. San Francisco: Holden Day, 1963 (in Russian). English translation, Holden-Day, Inc., San Francisco, 1964.

AMARAL, J. C. *Análise, Projeto e simulação de uma arquitetura de controle reconfigurável para a plataforma multimissão*. 2009. 149 p. (INPE-15682-TDI/1456). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2008. Disponível em: <<http://urlib.net/8JMKD3MGP8W/349LRHS>>. Acesso em: 03 jul. 2013.

ASTRÖM, K. J.; WITTENMARK B. *Adaptive Control*. 2.. ed. New York, NY: Pearson Education, 2006.

CARINI, A.; MATHEWS, V. J.; SICURANZA, G. L. Sufficient stability bounds for slowly varying direct-form recursive linear filters and their applications in adaptive IIR filters. *IEEE Trans. on Signal Processing*, v. 47, n. 9, p. 2561-2567, Sept. 1999.

CHATTERJEE, D.; LIBERZON, D. Stabilizing randomly switched systems. *SIAM Journal on Control and Optimization*, v. 49, n. 5, p. 2008-2031, 2011. 2007.

COMPTON, K.; HAUCK, S. Reconfigurable Computing: A Survey of Systems and Software. *ACM Computing Surveys*, Vol. 34, No. 2, June 2002, pp. 171-210.

DIALLO, D.; BENBOUZID, M. H.; MAKOUF, A. A fault tolerant control architecture for induction motor drives in automotive applications. *IEEE Transactions on Vehicular Technology*, v. 53, n. 06; Nov. 2004.

GRAEBE, S. F.; AHLÉN, A. L. B. Dynamic transfer among alternative controllers and its relation to antiwindup controller design. *IEEE Transactions on Control Systems Technology*, v 4, n. 1, Jan. 1996.

GOBATO, M. F. *Controles monovariáveis e multivariáveis aplicados a sistemas aeroespaciais fracamente ou fortemente acoplados*. 2006. 388 p. (INPE-14494-TDI/1175). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2006. Disponível em: <<http://urlib.net/sid.inpe.br/MTC-m13@80/2006/06.30.18.04>>. Acesso em: 03 jul. 2013.

GÖRÇEK, C.; EUN, Y.; KABAMBA, P. T.; MEERKOV, S. M. An LQG approach to systems with saturating actuators and anti-windup implementation. *Mathematical Problems in Engineering*, v. 8, n.4-5, p. 311-321, 2002,.

GULER, M.; CLEMENTIS, S.; WILLS, L. M.; HECK, B. S.; VACHTSEVANOS, G. J. Transition management for reconfigurable hybrid control systems. *Control Systems Magazine, IEEE*, v. 23, n. 1, p. 36 - 49, Feb 2003.

HISKENS, I. A.; PAI, M. A. Trajectory sensitivities of hybrid systems *IEEE Transactions on Circuits and Systems - Part I: Fundamental Theory and Applications*, v. 47, n..2, p. 204- 220, Feb. 2000.

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS (INPE). *Multi-Mission Platform Attitude Control and Data Handling (ACDH) subsystem specification*. São José dos Campos, 2001. A822700-SPC-001/05.

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS (INPE). *Multimission Platform data package for system requirements*. São José dos Campos – SP, 2001. . Review (SRR) (A822000-DPK-01/D5a).

HÖPFLINGER, M.; KREBS, A.; PRADALIER, C.; LEE, C.; OBSTEI, R.; SIEGWART, R. Description of the Locomotion Control Architecture on the ExoMars Rover Breadbord. In: ESA WORKSHOP ON ADVANCED SPACE

TECHNOLOGIES FOR ROBOTICS AND AUTOMATION – ASTRA 2008. 10., 2008, Noordwijk, The Netherlands. *Proceedings...* Noordwijk: ESA/ESTEC, 2008.

KALMAN, R. E. Lyapunov functions for the problem of Lur'e in automatic control. *National Academy of Sciences, Proceedings*, v. 49, n. 2, p. 201-205, Feb. 1963.

LIBERZON, D.; MORSE A.S. *Basic Problems in stability and design of switched systems. Control Systems Magazine, IEEE*, v. 19, n. 5, p. 59 – 70, Oct 1999.

LIU, C.W.; THORP, J. S. New methods for computing power system dynamic response for real-time transient stability prediction. *IEEE Transactions on Circuits and Systems – Part I: Fundamental Theory and Applications*, v. 47, n.3, p. 324-337, Mar. 2000.

MAYYAS, K.; ABOULNASR, T. On transient error surfaces of output error IIR adaptive filtering. *IEEE Trans. on Signal Processing*, v. 46, n. 3, p. 766-771, Mar. 1998.

MIDDLEBROOK, R. D.; CUK, S. A general unified approach to modeling switching converter power stages. *International Journal of Electronics*, v. 42, n.6, p. 521-550, 1977.

MOREIRA, M. L. B. *Projeto e simulação de um controle discreto para a plataforma multi-missão e sua migração para um sistema operacional de tempo real*. 2006. 181 p. (INPE-14202-TDI/1103). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2006. Disponível em: <<http://urlib.net/sid.inpe.br/MTC-m13@80/2006/07.10.13.42>>. Acesso em: 03 jul. 2013.

MOUSTAKIDES, G. V. Study of the transient phase of the forgetting factor RLS. *IEEE Trans. on Signal Processing*, v. 45, n. 10, p. 2468-2476, Oct. 1997.

NARENDRA, K. S.; BALAKRISHNAN, J. A common lyapunov function for stable LTI systems with commuting A-Matrices. *IEEE Transactions on Automatic Control*, v. 39, n. 12, Dec. 1994.

OGATA, K. *Engenharia de controle moderno*. Rio de Janeiro, RJ: Prentice-Hall do Brasil, 1985.

OLIVEIRA, E. M.; SOUZA M. L. O. Investigation on techniques for accurate phase or time synchronization in reconfigurable control systems In: CONGRESSO SAE BRASIL, 2012, São Paulo, SP. *Anais...* São Paulo: SAE Brasil, 2012.

PACKARD, A. *ME 132*. Berkeley, CA: University of California at Berkeley, 2005.

PAPAKONSTANTINO, A.; LIANG, Y.; STRATTON, J. A.; GURURAJ K.; DEMING CHEN, HWU, W. W.; CONG, J. *Multilevel Granularity Parallelism Synthesis on FPGAs*. FCUDA-II-FCCM 2011.

PENG, Y.; VRANCIC, D.; HANUS, R.; WELLER, S. S. R. Anti-windup design for multivariable controllers. *Automatica*, v. 34 n. 12, p. 1559-1565, 1998.

POPOV, V. M. *Hyperstability of control systems*. Berlin Heidelberg, New York, NY: Editura Academiei Bucuresti/Springer-Verlag, 1973.

PRUDÊNCIO, S. V. *Simulação digital em tempo real de um sistema de controle de atitude magnético autônomo de um satélite*. 2000. 194 - no original não contaram o pré-texto p. (INPE-7514-TDI/727). Dissertação (Mestrado em Ciência Espacial) - Instituto Nacional de Pesquisas Espaciais, Sao Jose dos Campos, 1997. Disponível em: <<http://urlib.net/sid.inpe.br/iris@1905/2005/07.29.22.26>>. Acesso em: 03 jul. 2013.

SILVA, J. M. G. *Ajuste de Controladores PID*. Disponível em: www.ece.ufrgs.br/~jmgomes/pid/Apostila/apostila/node31.html. Acesso em: 03 de outubro de 2012.

SIMON, G.; KOVÁCHSÁZY, T.; PÉCELI G. Transient management in reconfigurable systems. *Lecture Notes in Computer Science*, v. 1936/2001, p.90-98, 2001.

TODMAN, T.J.; CONSTANTINIDES, G.A.; WILTON, S.J.E.; MENCER, O.; LUK, W.; CHEUNG, P.Y.K. Reconfigurable Computing: Architectures and Design Methods. *IEEE Proc.-Comput. Digit. Tech.*, v. 152, No. 2, March 2005.

TORFS, D. E.; VUERINCKX, R.; SWEVERS, J.; SCHOUKENS, J. Comparison of two feedforward design methods aiming at accurate trajectory tracking of the end point of a flexible robot arm. *IEEE Transactions on Control Systems Technology*, v. 6, n. 1, p 2-14, Jan. 1998.

VÄLIMÄKI, V.; LAAKSO, T.I. Suppression of transients in variable recursive digital filters with novel and efficient cancellation method. *IEEE Trans. on Signal Processing*, v. 46, n. 12, p. 3408-3414, Dec. 1998.

VINCI, L. *Exomars-le-Projet-de-Rover-Martien-Europeen*. Planet Techno Science, 2012. Disponível em: <http://www.planet-techno-science.com/ciel-et-espace/exomars-le-projet-de-rover-martien-europeen>. Acesso em: 01 Nov 2012.

WERTZ, J. R. *Spacecraft attitude determination and control*. 1. ed. Dordrecht, Holland: D. Reidel Publishing Co., 1978.

WIE, B. *Space vehicle dynamics and control*. 1a. ed. Reston, Virginia, USA: AIAA Educational Series, 1998. 661 p. 58, 59, 61.

YAMÉ, J. J.; KINNAERT, M. On bumps and reduction of switching transients in multicontroller systems. *Mathematical Problems in Engineering*, v. 2007, Article

ID 54212, 16 pages doi:10.1155/2007/54212. Research Article, Hindawi Publishing Corporation.

APÊNDICE A: CÓDIGOS EM MATLAB PARA CALCULAR E PLOTAR OS GRÁFICOS DOS CRITÉRIOS INTEGRAIS

Seguem abaixo as linhas de código em MatLab usados para calcular e plotar os gráficos dos critérios integrais.

Referentes ao Critério ISE

Determinação das Equações 4.32 e 4.33:

```
syms z kd kp ki
L1=(-z*kp+sqrt((z*kp)^2-4*(kd+z)*z*ki))/2*(kd+z);
L2=(-z*kp-sqrt((z*kp)^2-4*(kd+z)*z*ki))/2*(kd+z);
A= L1/(L1-L2);
B= L2/(L1-L2);
J= (-A^2/(2*L1)-2*A*B/(L1+L2)-B^2/(2*L2));
simplify(J)
```

```
J = 1/2/z*(z*kp^2+4*ki*kd+4*z*ki)/kp/(z*kp^2-4*ki*kd-4*z*ki)/(kd+z)
J = 1./2./z.*(z.*kp.^2+4.*ki.*kd+4.*z.*ki)./kp./(z.*kp.^2-4.*ki.*kd-4.*z.*ki)./(kd+z)
```

```
dJ = 1/2/z*(z*kp^2+4*ki*kd+4*z*ki)/kp/(z*kp^2-4*ki*kd-4*z*ki)/(kd+z)
dJ = 1/2/z.*(z.*kp.^2+4.*ki.*kd+4.*z.*ki)/kp./(z.*kp.^2-4.*ki.*kd-4.*z.*ki)./(kd+z)
```

Gráfico do caso da PMM:

```
kd = 40.593;
kp = 454.11;
ki = 1;
z = [0:0.0001:10];
J = 1./2./z.*(z.*kp.^2+4.*ki.*kd+4.*z.*ki)./kp./(z.*kp.^2-4.*ki.*kd-4.*z.*ki)./(kd+z);
plot(z,J)
```

Gráfico do caso dos motores CC:

```
kd = 0.04;
kp = 3;
ki = 1.15;
z = [0:0.01:1000];
J = 1./2./z.*(z.*kp.^2+4.*ki.*kd+4.*z.*ki)./kp./(z.*kp.^2-4.*ki.*kd-4.*z.*ki)./(kd+z);
plot(z,J)
```

Referentes ao Critério ITSE

Determinação das Equações 4.44 e 4.45:

```
syms z kd kp ki
```

```

L1=(-z*kp+sqrt((z*kp)^2-4*(kd+z)*z*ki))/2*(kd+z);
L2=(-z*kp-sqrt((z*kp)^2-4*(kd+z)*z*ki))/2*(kd+z);
A= L1/(L1-L2);
B= L2/(L1-L2);
J=A^2/(4*L1^2)+2*A*B/(L1+L2)^2+B^2/(4*L2^2);
simplify(J)
dJ = diff(J)
simplify(dJ)

```

```

J = -(z*(kp^2/2 + 2*ki) + 2*kd*ki)/(kp^2*z^2*(kd + z)^2*(- z*kp^2 +
4*kd*ki + 4*ki*z))

```

```

J = -(z.*(kp.^2./2 + 2.*ki) + 2.*kd.*ki)./(kp.^2.*z.^2.*(kd +
z).^2.*(- z.*kp.^2 + 4.*kd.*ki + 4.*ki.*z))

```

```

dJ = (z^3*(32*ki^2 - 2*kp^4) + 16*kd^3*ki^2 - kd*z^2*(- 80*ki^2 +
4*ki*kp^2 + kp^4) - kd^2*z*(4*ki*kp^2 - 64*ki^2))/(kp^2*z^3*(kd +
z)^3*(- z*kp^2 + 4*kd*ki + 4*ki*z)^2)

```

```

dJ = (z.^3.*(32.*ki.^2 - 2.*kp.^4) + 16.*kd.^3.*ki.^2 - kd.*z.^2.*(-
80.*ki.^2 + 4.*ki.*kp.^2 + kp.^4) - kd.^2.*z.*(4.*ki.*kp.^2 -
64.*ki.^2))./(kp.^2.*z.^3.*(kd + z).^3.*(- z.*kp.^2 + 4.*kd.*ki +
4.*ki.*z).^2)

```

Gráfico do caso da PMM:

```

kd = 40.593;
kp = 454.11;
ki = 1;
z = [0:0.0001:10];
J = -(z.*(kp.^2./2 + 2.*ki) + 2.*kd.*ki)./(kp.^2.*z.^2.*(kd +
z).^2.*(- z.*kp.^2 + 4.*kd.*ki + 4.*ki.*z));
plot(z,J)

```

Gráfico do caso dos motores CC:

```

kd = 0.04;
kp = 3;
ki = 1.15;
z = [0:0.01:1000];
J = -(z.*(kp.^2./2 + 2.*ki) + 2.*kd.*ki)./(kp.^2.*z.^2.*(kd +
z).^2.*(- z.*kp.^2 + 4.*kd.*ki + 4.*ki.*z));
plot(z,J)

```

Referentes ao Critério IME

Determinação das Equações 4.55 e 4.56:

```

syms z kd kp ki
L1=(-z*kp+sqrt((z*kp)^2-4*(kd+z)*z*ki))/2*(kd+z);
L2=(-z*kp-sqrt((z*kp)^2-4*(kd+z)*z*ki))/2*(kd+z);
A= L1/(L1-L2);
B= L2/(L1-L2);
J=-(A/L1+B/L2);
simplify(J);
dJ = diff(J);
simplify(dJ);

```

$$J = -2. / ((kd + z) * (kp.^2 * z.^2 - ki * z * (4 * kd + 4 * z)).^(1./2))$$

$$dJ = -(2 * (2 * ki * kd^2 - kd * kp^2 * z + 10 * ki * kd * z - 2 * kp^2 * z^2 + 8 * ki * z^2)) / ((kd + z)^2 * (kp.^2 * z.^2 - ki * z * (4 * kd + 4 * z)).^(3./2))$$

$$dJ = -(2 * (2 * ki * kd.^2 - kd * kp.^2 * z + 10 * ki * kd * z - 2 * kp.^2 * z.^2 + 8 * ki * z.^2)) / ((kd + z).^2 * (kp.^2 * z.^2 - ki * z * (4 * kd + 4 * z)).^(3./2))$$

Gráfico do caso da PMM:

```
kd = 40.593;
kp = 454.11;
ki = 1;
z = [0:0.0001:10];
J = 2./((kd + z).*(kp.^2.*z.^2- ki.*z.*(4.*kd + 4.*z)).^(1./2))
plot(z,J)
```

Valor mínimo de zeta para o qual o critério é válido:

```
kd = 40.593;
kp = 454.11;
ki = 1;
zeta = 4*ki*kd/(kp^2-4*ki)
```

Gráfico do caso dos motores CC:

```
kd = 0.04;
kp = 3;
ki = 1.15;
z = [0:0.0001:10];
J = 2./((kd + z).*(kp.^2.*z.^2- ki.*z.*(4.*kd + 4.*z)).^(1./2))
plot(z,J)
```

Valor mínimo de zeta para o qual o critério é válido:

```
kd = 0.04;
kp = 3;
ki = 1.15;
zeta = 4*ki*kd/(kp^2-4*ki)
```

Referentes ao Critério ITME

Determinação das Equações 4.62 e 4.63:

```
syms z kd kp ki
L1=(-z*kp+sqrt((z*kp)^2-4*(kd+z)*z*ki))/2*(kd+z);
L2=(-z*kp-sqrt((z*kp)^2-4*(kd+z)*z*ki))/2*(kd+z);
A= L1/(L1-L2);
B= L2/(L1-L2);
J=A/(L1 )^2 +B/(L2 )^2
simplify(J)
dJ= diff(J)
simplify(dJ)
```

$$J = -(4*kp)/(ki*(4*kd + 4*z)*(kd + z)^2*(kp^2*z^2 - ki*z*(4*kd + 4*z))^{(1/2)})$$

$$J = -(4.*kp)./(ki.*(4.*kd + 4.*z).*(kd + z).^2.*(kp.^2.*z.^2 - ki.*z.*(4.*kd + 4.*z)).^(1./2))$$

$$dJ = -(kp*(2*ki*kd^2 - kd*kp^2*z + 18*ki*kd*z - 4*kp^2*z^2 + 16*ki*z^2))/(ki*(kd + z)^4*(kp^2*z^2 - ki*z*(4*kd + 4*z))^{(3/2)})$$

$$dJ = -(kp.*(2.*ki.*kd.^2 - kd.*kp.^2.*z + 18.*ki.*kd.*z - 4.*kp.^2.*z.^2 + 16.*ki.*z.^2))./(ki.*(kd + z).^4.*(kp.^2.*z.^2 - ki.*z.*(4.*kd + 4.*z)).^(3./2))$$

Gráfico do caso da PMM:

```
kd = 40.593;
kp = 454.11;
ki = 1;
z = [0:0.0001:10];
J = (4.*kp)./(ki.*(4.*kd + 4.*z).*(kd + z).^2.*(kp.^2.*z.^2 - ki.*z.*(4.*kd + 4.*z)).^(1./2));
plot(z,J)
```

Valor mínimo de zeta para o qual o critério é válido:

```
kd = 40.593;
kp = 454.11;
ki = 1;
zeta = 4*ki*kd/(kp^2-4*ki)
```

Gráfico do caso dos motores CC:

```
kd = 0.04;
kp = 3;
ki = 1.15;
z = [0:0.0001:10];
J = (4.*kp)./(ki.*(4.*kd + 4.*z).*(kd + z).^2.*(kp.^2.*z.^2 - ki.*z.*(4.*kd + 4.*z)).^(1./2));
plot(z,J)
```

Valor mínimo de zeta para o qual o critério é válido:

```
kd = 0.04;
kp = 3;
ki = 1.15;
zeta = 4*ki*kd/(kp^2-4*ki)
```

APÊNDICE B: CÓDIGOS EM MATLAB PARA CALCULAR E PLOTAR OS GRÁFICOS DA ESTABILIDADE GERAL.

Considerando o sistema completo da PMM. A variável "k" determina o tamanho do intervalo de transição.

```
s = tf('s')
k= 1;
kp1=40.593;
kd1=454.11;
ki1=5;
kp2 = 40.593;
kd2 = 454.11;
ki2 = 1;
```

Sistema inicial:

```
A = 0.06*s/(s+1/20);
G = 1/(295.97401*s^2);
C = (kp1 + ki1/s + kd1*s);
H = (C*A*G)/(1+C*A*G);
[polos,zeros] = pzmap(H)
pzmap(H)
```

Sistema final:

```
A = 0.06*s/(s+1/20);
G = 1/(295.97401*s^2);
C = (kp2 + ki2/s + kd2*s);
H = (C*A*G)/(1+C*A*G);
[polos,zeros] = pzmap(H)
pzmap(H)
```

Sistema durante cross-fading linear:

```
A = 0.06*s/(s+1/20);
G = 1/(295.97401*s^2);
C = expressão do controle com crossfading linear
H = (C*A*G)/(1+C*A*G);
[polos,zeros] = pzmap(H)
pzmap(H)
```

Sistema durante cross-fading exponencial:

```
A = 0.06*s/(s+1/20);
G = 1/(295.97401*s^2);
C = expressão do controle com crossfading exponencial
H = (C*A*G)/(1+C*A*G);
[polos,zeros] = pzmap(H)
pzmap(H)
```

APÊNDICE C: CÓDIGO DO MATRIXX/SYSTEMBUILD PARA O BLOCO DE SIMULAÇÃO DE TRANSIÇÃO USANDO CRITÉRIOS INTEGRAIS.

```

/*****
*****
|               AutoCode/C (TM) Code Generator 62mx1118
|               INTEGRATED SYSTEMS INC.,   SUNNYVALE, CALIFORNIA
*****
*****
rtf filename      : PMM_proporcional_ISE.rtf
Filename         : PMM_proporcional_ISE.c
Dac filename     :
C:\isi\mx_62.2\case\ACC\templates\c_sim.dac
Generated on     : Sun May 12 19:33:54 2013
Dac file created on : Fri Nov 19 00:07:59 1999
Options         : -l c

--
--   Number of External Input : 0
--   Number of External Output: 3
--
--   Scheduler Frequency:      100.0
--
--   SUBSYSTEM  FREQUENCY  TIME_SKEW  OUTPUT_TIME  TASK_TYPE
--   -----   -
--   1          100.0     0.0        0.0          CONTINUOUS
--   2          100.0     0.0        0.0          PERIODIC

*/

#include <stdio.h>
#include <math.h>
#include "sa_sys.h"
#include "sa_defn.h"
#include "sa_types.h"
#include "sa_math.h"
#include "sa_matrix.h"
#include "sa_user.h"
#include "sa_utils.h"
#include "sa_time.h"
#include "sa_fuzzy.h"

```

```

/** System Data */

/***** Structure to drive disconnected input/output. *****/
struct _DcZero {
    RT_FLOAT dzero;
};

static const struct _DcZero dczero = {0.0};

#define EPSILON                1.4901161193847656E-008
#define EPS                    (4.0 * EPSILON)
#define ABSTOL                 EPSILON
#define XREMAP                 1

#define SCHEDULER_FREQ        100.0
#define NTASKS                2
#define NUMIN                 0
#define NUMOUT                 3
#define SCHEDULER_ID         0
#define PREEMPTABLE           2

enum TASK_STATE_TYPE { IDLE, RUNNING, BLOCKED, UNALLOCATED };

static RT_INTEGER             ERROR_FLAG [NTASKS+1];
static RT_BOOLEAN            SUBSYS_PREINIT [NTASKS+1];
static RT_BOOLEAN            SUBSYS_INIT [NTASKS+1];
static RT_DURATION           SUBSYS_TIME [NTASKS+1];
static enum TASK_STATE_TYPE  TASK_STATE [NTASKS+1];

/***** System Ext I/O and Sample-Hold type declarations.*****/
struct _Sys_ExtOut {
    RT_FLOAT controle_efetivo;
    RT_FLOAT atenua;
    RT_FLOAT controle_corretor;
};

struct _Subsys_1_in {
    RT_FLOAT Controle_Pid;
    RT_FLOAT atenua;
    RT_FLOAT sentido_da_transicao;
};

struct _Subsys_1_out {
    RT_FLOAT controle_corretor;
    RT_FLOAT diferenca_controles;
    RT_FLOAT controle_efetivo;
    RT_FLOAT Planta_graus;

```

```

};

struct _Subsys_2_in {
    RT_FLOAT controle_corretor;
    RT_FLOAT diferenca_controles;
    RT_FLOAT Planta_graus;
    RT_FLOAT dzero;
};

struct _Subsys_2_out {
    RT_FLOAT Controle_Pid;
    RT_FLOAT atenua;
    RT_FLOAT sentido_da_transicao;
};

/**** System Ext I/O and Subsystem I/O type definitions and
****
**** Pointers to SubSystem Outputs ReadOnly/Work areas.
****/
struct _Sys_ExtOut sys_extout;
struct _Subsys_1_in subsys_1_in;
struct _Subsys_1_out subsys_1_out[2];
struct _Subsys_1_out *ss1_out;
struct _Subsys_1_out *ss1_outw;
struct _Subsys_2_in subsys_2_in;
struct _Subsys_2_out subsys_2_out[2];
struct _Subsys_2_out *ss2_out;
struct _Subsys_2_out *ss2_outw;

static RT_FLOAT ExtIn [NUMIN+1];
static RT_FLOAT ExtOut [NUMOUT+1];

RT_BOOLEAN RESETC;

/***** Continuous Subsystem states and info type declarations.
*****/
struct _Subsys_1_states {
    RT_FLOAT Planta_S1;
    RT_FLOAT Planta_S2;
    RT_FLOAT PMM_proporcional_ISE_59_S1;
    RT_FLOAT PMM_proporcional_ISE_22_S1;
    RT_FLOAT PMM_proporcional_ISE_22_S2;
    RT_FLOAT PMM_proporcional_ISE_47_S1;
    RT_FLOAT PMM_proporcional_ISE_47_S2;
    RT_FLOAT PMM_proporcional_ISE_11_S1;
    RT_FLOAT Atuador_roda_S1;
};
struct _Subsys_1_info {
    RT_INTEGER iinfo[5];
    RT_FLOAT rinfo[5];
};

```

```

/***** Continuous Subsystem states and info definitions.
*****/
struct _Subsys_1_states subsys_1_states[2];
struct _Subsys_1_info subsys_1_info;
RT_INTEGER *ssl_iinfo = &subsys_1_info.iinfo[0];
RT_FLOAT *ssl_rinfo = &subsys_1_info.rinfo[0];

/***** Tasks declarations *****/

/***** Continuous Subsystem *****/

/***** Subsystem 1 *****/
extern void subsys_1( struct _Subsys_1_in *U, struct _Subsys_1_out
*Y
, struct _Subsys_1_states *S, struct _Subsys_1_info *I);

extern void rungekutta2( RT_INTEGER n,
RT_FLOAT *x, RT_FLOAT *dxdt, RT_FLOAT t, RT_FLOAT h);

/***** Subsystem 2 *****/
extern void subsys_2( struct _Subsys_2_in *U, struct _Subsys_2_out
*Y);

/***** Tasks code *****/

/* The function rungekutta2 employs the second-order Runge-Kutta
method with
Kutta's coefficients to integrate a system of n simultaneous
first
order ordinary differential equations  $dxdt[j] = dx[j]/dt$ ,
( $j=1,2,\dots,n$ ),
across one step of length h in the independent variable t,
subject
to initial conditions  $x[j]$ , ( $j=1,2,\dots,n$ ). Each  $dxdt[j]$ , the
derivative
of  $x[j]$ , must be computed two times per integration step by
calling the
state derivatives and output equations function (subsys_1()).

```

```

    savex(j) is used to save the initial value of x(j) and phi(j) is
the
    increment function for the j(th) equation.
    As written, n may be no larger than 9.
    (Modified Euler)
*/
void rungekutta2( RT_INTEGER n,
                 RT_FLOAT *x, RT_FLOAT *dxdt, RT_FLOAT t, RT_FLOAT h)
{
    RT_FLOAT phi[9];
    RT_FLOAT savex[9];
    RT_INTEGER j, retval;
    RT_FLOAT hh = t;

    ssl_rinfo[0] = hh;
    ssl_iinfo[2]=1; ssl_iinfo[3]=1; ssl_iinfo[4]=8;
    subsys_1(&subsys_1_in, ssl_outw, subsys_1_states,
&subsys_1_info);

    for (j=0; j<n; j++) {
        savex[j] = x[j];
        phi[j] = dxdt[j];
        x[j] = savex[j] + h*dxdt[j];
    }
    hh=t+h;
    ssl_rinfo[0] = hh;
    ssl_iinfo[2]=1; ssl_iinfo[3]=1; ssl_iinfo[4]=0;
    subsys_1(&subsys_1_in, ssl_outw, subsys_1_states,
&subsys_1_info);

    for (j=0; j<n; j++) x[j] = savex[j] + (phi[j] + dxdt[j])*h/2.0;
}

/***** Continuous Subsystem *****/

/***** Subsystem 1 *****/

void subsys_1( struct _Subsys_1_in *U
,struct _Subsys_1_out *Y
,struct _Subsys_1_states *S
,struct _Subsys_1_info *I
)
{
    RT_INTEGER *iinfo = &I->iinfo[0];
    RT_FLOAT *rinfo = &I->rinfo[0];
    RT_INTEGER INIT      = iinfo[1];
    RT_INTEGER STATES    = iinfo[2];
    RT_INTEGER OUTPUTS   = iinfo[3];
    const RT_DURATION TIME = rinfo[0];
    const RT_DURATION TSAMP = rinfo[1];

    /***** States' and State Derivatives' Pointers. *****/

```

```

struct _Subsys_1_states *X = &S[0];
struct _Subsys_1_states *XD = &S[1];

/***** Parameters. *****/

static RT_FLOAT R_P[3];
RT_INTEGER cnt;
static const RT_FLOAT _R_P[3] = {0.0};

/***** Local Block Outputs. *****/

RT_FLOAT referencia_1;
RT_FLOAT Planta_1;
RT_FLOAT PMM_proporcional_ISE_98_1;
RT_FLOAT PMM_proporcional_ISE_1_1;
RT_FLOAT controle2;
RT_FLOAT memoria_pid;
RT_FLOAT controle2_1;
RT_FLOAT memoria_p;
RT_FLOAT PMM_proporcional_ISE_46_1;
RT_FLOAT PMM_proporcional_ISE_25_1;
RT_FLOAT PMM_proporcional_ISE_7_1;
RT_FLOAT PMM_proporcional_ISE_8_1;
RT_FLOAT PMM_proporcional_ISE_47_1;
RT_FLOAT atual;
RT_FLOAT PMM_proporcional_ISE_28_1;
RT_FLOAT derivada;
RT_FLOAT PMM_proporcional_ISE_26_1;
RT_FLOAT atuador;

/***** Algorithmic Local Variables. *****/

RT_FLOAT Xtemp[2];
RT_INTEGER i;
RT_INTEGER j;
RT_INTEGER k;
RT_FLOAT xn;
RT_FLOAT alpha;

/***** Initialization. *****/

if( INIT ) {
    for( cnt=0;cnt<3;cnt++ ) {
        R_P[cnt] = _R_P[cnt];
    }
    Planta_1 = 0.5;
}

/***** Output Update. *****/

if( OUTPUTS ) {
    /* ----- Algebraic Expression */

```

```

/* {PMM proporcional ISE.referencia.3} */
referencia_1 = 0.0;
/* ----- Gain Poles-Zeros */
/* {PMM proporcional ISE.Planta.4} */
if (!INIT) {
    Planta_1 = 0.054058800635907185*X->Planta_S1;
    Planta_1 = Planta_1 - 0.054058800635907185*X-
>Planta_S2;
}
/* ----- Algebraic Expression */
/* {PMM proporcional ISE..98} */
PMM_proporcional_ISE_98_1 = 0.0;
/* ----- Gain Poles-Zeros */
/* {PMM proporcional ISE..59} */
Y->controle_corretor = 0.08*X->PMM_proporcional_ISE_59_S1;
/* ----- Summer */
/* {PMM proporcional ISE..1} */
PMM_proporcional_ISE_1_1 = referencia_1 - Planta_1;
/* ----- PID Controller */
/* {PMM proporcional ISE..22} */
if (XREMAP) {
    if (INIT) {
        X->PMM_proporcional_ISE_22_S1 =
PMM_proporcional_ISE_1_1 - 0.1*
        X->PMM_proporcional_ISE_22_S1;
    }
}
controle2 = PMM_proporcional_ISE_1_1*50.0;
controle2 = controle2 + X->PMM_proporcional_ISE_22_S2;
controle2 = controle2 + (PMM_proporcional_ISE_1_1 - X->
    PMM_proporcional_ISE_22_S1)*400.0/0.1;
/* ----- BlockScript Block */
/* {PMM proporcional ISE.linhas de codigo.5} */
if (U->atenua == 0) {
    R_P[0] = controle2;
}
memoria_pid = R_P[0];
/* ----- PID Controller */
/* {PMM proporcional ISE..13} */
controle2_1 = PMM_proporcional_ISE_1_1*10.0;
/* ----- BlockScript Block */
/* {PMM proporcional ISE.linhas de codigo.6} */
if (U->atenua == 0) {
    R_P[1] = controle2_1;
}
memoria_p = R_P[1];
/* ----- Summer */
/* {PMM proporcional ISE..46} */
PMM_proporcional_ISE_46_1 = memoria_p - memoria_pid;
/* ----- ElementProduct */
/* {PMM proporcional ISE..25} */
PMM_proporcional_ISE_25_1 = U->sentido_da_transicao*
    PMM_proporcional_ISE_46_1;
/* ----- Summer */

```

```

/* {PMM proporcional ISE..24} */
Y->diferenca_controles = Y->controle_corretor;
Y->diferenca_controles = Y->diferenca_controles -
    PMM_proporcional_ISE_25_1;
/* ----- Data Path Switch */
/* {PMM proporcional ISE..7} */
if (U->atenua <= 0.0) {
    PMM_proporcional_ISE_7_1 = PMM_proporcional_ISE_98_1;
}
else if (U->atenua < 0.001) {
    alpha = U->atenua/0.001;
    PMM_proporcional_ISE_7_1 =
alpha*PMM_proporcional_ISE_25_1 + (1.0 -
    alpha)*PMM_proporcional_ISE_98_1;
}
else {
    PMM_proporcional_ISE_7_1 = PMM_proporcional_ISE_25_1;
}
/* ----- Summer */
/* {PMM proporcional ISE..8} */
PMM_proporcional_ISE_8_1 = PMM_proporcional_ISE_7_1;
PMM_proporcional_ISE_8_1 = PMM_proporcional_ISE_8_1 - Y->
    controle_corretor;
/* ----- PID Controller */
/* {PMM proporcional ISE..47} */
if (XREMAP) {
    if (INIT) {
        X->PMM_proporcional_ISE_47_S1 =
PMM_proporcional_ISE_8_1 - 0.1*
        X->PMM_proporcional_ISE_47_S1;
    }
}
PMM_proporcional_ISE_47_1 =
PMM_proporcional_ISE_8_1*40.593;
PMM_proporcional_ISE_47_1 = PMM_proporcional_ISE_47_1 + X-
>
    PMM_proporcional_ISE_47_S2;
PMM_proporcional_ISE_47_1 = PMM_proporcional_ISE_47_1 + (
    PMM_proporcional_ISE_8_1 - X-
>PMM_proporcional_ISE_47_S1)*454.11/
    0.1;
/* ----- BlockScript Block */
/* {PMM proporcional ISE.linhas de codigo.10} */
if (U->sentido_da_transicao == 1) {
    atual = memoria_pid;
}
else if (U->sentido_da_transicao == -1) {
    atual = memoria_p;
}
/* ----- Summer */
/* {PMM proporcional ISE..28} */
PMM_proporcional_ISE_28_1 = atual - Y-
>diferenca_controles;
/* ----- PID Controller */

```

```

/* {PMM proporcional ISE..11} */
if (XREMAP) {
    if (INIT) {
        X->PMM_proporcional_ISE_11_S1 = Y-
>diferenca_controles - 0.1*X->
        PMM_proporcional_ISE_11_S1;
    }
}
derivada = (Y->diferenca_controles - X-
>PMM_proporcional_ISE_11_S1)/
0.1;
/* ----- Data Path Switch */
/* {PMM proporcional ISE..26} */
if (U->Controle_Pid <= 0.0) {
    PMM_proporcional_ISE_26_1 = controle2;
}
else if (U->Controle_Pid < 0.001) {
    alpha = U->Controle_Pid/0.001;
    PMM_proporcional_ISE_26_1 = alpha*controle2_1 + (1.0 -
alpha)*
    controle2;
}
else {
    PMM_proporcional_ISE_26_1 = controle2_1;
}
/* ----- Data Path Switch */
/* {PMM proporcional ISE..9} */
if (U->atenua <= 0.0) {
    Y->controle_efetivo = PMM_proporcional_ISE_26_1;
}
else if (U->atenua < 0.001) {
    alpha = U->atenua/0.001;
    Y->controle_efetivo = alpha*PMM_proporcional_ISE_28_1 +
(1.0 -
    alpha)*PMM_proporcional_ISE_26_1;
}
else {
    Y->controle_efetivo = PMM_proporcional_ISE_28_1;
}
/* ----- Gain Poles-Zeros */
/* {PMM proporcional ISE.Atuador roda.12} */
atuador = 0.05366563145999495*X->Atuador_roda_S1;
atuador = atuador + 0.06*Y->controle_efetivo;
/* ----- Algebraic Expression */
/* {PMM proporcional ISE.conversao radianos graus.14} */
Y->Planta_graus = Planta_1*180.0/3.14159;
}

/***** State Update. *****/
if( STATES ) {
    /* ----- Gain Poles-Zeros */
    /* {PMM proporcional ISE.Planta.4} */
    if (INIT && XREMAP) {

```

```

        Xtemp[0] = 18.498375625*Planta_1;
        Xtemp[1] = 0.0;
        X->Planta_S1 = Xtemp[0];
        X->Planta_S2 = Xtemp[1];
    }
    XD->Planta_S1 = 0.0;
    XD->Planta_S2 = 0.0;
    XD->Planta_S1 = XD->Planta_S1 + X->Planta_S2 +
0.0625*atuador;
    XD->Planta_S2 = XD->Planta_S2 + 0.0625*atuador;
    /* ----- Gain Poles-Zeros */
    /* {PMM proporcional ISE..59} */
    XD->PMM_proporcional_ISE_59_S1 =
0.125*PMM_proporcional_ISE_47_1;
    /* ----- PID Controller */
    /* {PMM proporcional ISE..22} */
    XD->PMM_proporcional_ISE_22_S2 = PMM_proporcional_ISE_1_1;
    XD->PMM_proporcional_ISE_22_S1 = (PMM_proporcional_ISE_1_1
- X->
        PMM_proporcional_ISE_22_S1)/0.1;
    /* ----- PID Controller */
    /* {PMM proporcional ISE..47} */
    XD->PMM_proporcional_ISE_47_S2 = PMM_proporcional_ISE_8_1;
    XD->PMM_proporcional_ISE_47_S1 = (PMM_proporcional_ISE_8_1
- X->
        PMM_proporcional_ISE_47_S1)/0.1;
    /* ----- PID Controller */
    /* {PMM proporcional ISE..11} */
    XD->PMM_proporcional_ISE_11_S1 = (Y->diferenca_controles -
X->
        PMM_proporcional_ISE_11_S1)/0.1;
    /* ----- Gain Poles-Zeros */
    /* {PMM proporcional ISE.Atuador roda.12} */
    XD->Atuador_roda_S1 = - 0.05*X->Atuador_roda_S1;
    XD->Atuador_roda_S1 = XD->Atuador_roda_S1 -
0.055901699437494741*Y->
        controle_efetivo;
    }

    if(iinfo[1]) {
        SUBSYS_INIT[1] = FALSE;
        iinfo[1] = 0;
    }
    return;

EXEC_ERROR: ERROR_FLAG[1] = iinfo[0];
    iinfo[0]=0;
}

/***** Subsystem 2 *****/

```

```

/***** States type declaration. *****/
struct _Subsys_2_states {
    RT_FLOAT administrador_2_S1;
    RT_FLOAT maquina_de_estados2_S1;
    RT_FLOAT maquina_de_estados2_S2;
    RT_FLOAT maquina_de_estados2_S3;
    RT_FLOAT maquina_de_estados2_S4;
    RT_FLOAT maquina_de_estados2_S5;
    RT_INTEGER maquina_de_estados2_S6;
    RT_INTEGER maquina_de_estados2_S7;
};

void subsys_2(    struct _Subsys_2_in *U
                ,struct _Subsys_2_out *Y
                )
{

    /***** States Array. *****/

    static struct _Subsys_2_states ss_2_states[2];

    /***** Current and Next States Pointers. *****/

    static struct _Subsys_2_states *X;
    static struct _Subsys_2_states *XD;
    static struct _Subsys_2_states *XTMP;
    static RT_INTEGER iinfo[4];
    static RT_INTEGER INIT;
    const RT_DURATION TSAMP = 0.01;

    /***** Parameters. *****/

    static RT_FLOAT R_P[1];
    RT_INTEGER cnt;
    static const RT_FLOAT _R_P[1] = {0.0};

    /***** Local Block Outputs. *****/

    RT_FLOAT administrador_2_1;
    RT_FLOAT Controle_P;
    RT_FLOAT p_para_pid;
    RT_FLOAT pid_para_p;
    RT_FLOAT memoria_p;

    /***** Algorithmic Local Variables. *****/

    RT_FLOAT xn;
    RT_INTEGER KState;

    /***** Initialization. *****/

    if (SUBSYS_PREINIT[2]) {
        iinfo[0] = 0;

```

```

iinfo[1] = 1;
iinfo[2] = 1;
iinfo[3] = 1;
INIT = 1;
X = &ss_2_states[0];
XD = &ss_2_states[1];
X->administrador_2_S1 = 0.0;
X->maquina_de_estados2_S1 = 0.0;
X->maquina_de_estados2_S2 = 0.0;
X->maquina_de_estados2_S3 = 0.0;
X->maquina_de_estados2_S4 = 0.0;
X->maquina_de_estados2_S5 = 0.0;
X->maquina_de_estados2_S6 = 0;
X->maquina_de_estados2_S7 = 0;
XD->administrador_2_S1 = 0.0;
XD->maquina_de_estados2_S1 = 0.0;
XD->maquina_de_estados2_S2 = 0.0;
XD->maquina_de_estados2_S3 = 0.0;
XD->maquina_de_estados2_S4 = 0.0;
XD->maquina_de_estados2_S5 = 0.0;
XD->maquina_de_estados2_S6 = 0;
XD->maquina_de_estados2_S7 = 0;
R_P[0] = _R_P[0];
X->maquina_de_estados2_S6 = 2;
SUBSYS_PREINIT[2] = FALSE;
return;
}

/***** Output Update. *****/
/* ----- PID Controller */
/* {administrador..2} */
if (XREMAP) {
    if (INIT) {
        X->administrador_2_S1 = (-(0.1 + TSAMP)*X-
>administrador_2_S1 + U->
        controle_corretor)/TSAMP;
    }
}
administrador_2_1 = (U->controle_corretor - TSAMP*X-
>administrador_2_S1)/(
    0.1 + TSAMP);
/* ----- State Transition Diagram */
/* {administrador.maquina de estados2.12} */
XD->maquina_de_estados2_S1 = X->maquina_de_estados2_S1;
XD->maquina_de_estados2_S2 = X->maquina_de_estados2_S2;
XD->maquina_de_estados2_S3 = X->maquina_de_estados2_S3;
XD->maquina_de_estados2_S4 = X->maquina_de_estados2_S4;
XD->maquina_de_estados2_S5 = X->maquina_de_estados2_S5;
XD->maquina_de_estados2_S6 = X->maquina_de_estados2_S6;
if (!INIT) {
    KState = X->maquina_de_estados2_S6;
    switch (KState)
    {
    case 1:

```

```

        if (U->Planta_graus*U->Planta_graus > 100) {
            XD->maquina_de_estados2_S6 = 3;
        }
        break;
    case 2:
        if (U->Planta_graus*U->Planta_graus < 100) {
            XD->maquina_de_estados2_S6 = 4;
        }
        break;
    case 3:
        if (U->diferenca_controles*U->diferenca_controles <
0.02 ||
administrador_2_1*administrador_2_1 < 0.05) {
            XD->maquina_de_estados2_S6 = 2;
        }
        break;
    case 4:
        if (U->diferenca_controles*U->diferenca_controles <
0.02 ||
administrador_2_1*administrador_2_1 < 0.05) {
            XD->maquina_de_estados2_S6 = 1;
        }
        break;
    default:
        break;
}
}
Y->Controle_Pid = XD->maquina_de_estados2_S1;
Controle_P = XD->maquina_de_estados2_S2;
Y->atenua = XD->maquina_de_estados2_S3;
p_para_pid = XD->maquina_de_estados2_S4;
pid_para_p = XD->maquina_de_estados2_S5;
KState = XD->maquina_de_estados2_S6;
switch (KState)
{
case 1:
    Y->Controle_Pid = 1.0;
    break;
case 2:
    Controle_P = 1.0;
    break;
case 3:
    Y->atenua = 1.0;
    p_para_pid = 1.0;
    break;
case 4:
    Y->atenua = 1.0;
    pid_para_p = 1.0;
    break;
default:
    break;
}
/* ----- Algebraic Expression */
/* {administrador..13} */

```

```

Y->sentido_da_transicao = p_para_pid - pid_para_p;
/* ----- BlockScript Block */
/* {administrador.linhas de codigo.6} */
if (U->dzero == 0) {
    R_P[0] = U->dzero;
}
memoria_p = R_P[0];

/***** State Update. *****/
/* ----- PID Controller */
/* {administrador..2} */
XD->administrador_2_S1 = (0.1*X->administrador_2_S1 + U->
    controle_corretor)/(0.1 + TSAMP);

/* Copy back(s) and/or duplicate(s) */

/***** Swap state pointers. *****/

XTMP = X;
X = XD;
XD = XTMP;
INIT = 0;

if(iinfo[1]) {
    SUBSYS_INIT[2] = FALSE;
    iinfo[1] = 0;
}
return;

EXEC_ERROR: ERROR_FLAG[2] = iinfo[0];
iinfo[0]=0;
}

/***** (Continuous) Subsystem 1 initialization declaration.
*****/
extern void subsys_1_init( struct _Subsys_1_out *Y, struct
_Subsys_1_states *S, struct _Subsys_1_info *I);

/***** (Continuous) Subsystem 1 initialization definition.
*****/
void subsys_1_init( struct _Subsys_1_out *Y
,struct _Subsys_1_states *S
,struct _Subsys_1_info *I
)
{

/***** States' and State Derivatives' Pointers. *****/

struct _Subsys_1_states *X = &S[0];
struct _Subsys_1_states *XD = &S[1];

```

```

if (SUBSYS_PREINIT[1]) {
    I->iinfo[0] = 0;
    I->iinfo[1] = 1;
    I->iinfo[2] = 1;
    I->iinfo[3] = 1;
    I->iinfo[4] = 1;
    I->rinfo[0] = 0.0;
    I->rinfo[1] = 0.010000;
    I->rinfo[2] = 0.0;
    I->rinfo[3] = 0.0;
    I->rinfo[4] = 0.0;
    X = &S[0];
    XD = &S[1];
    X->Planta_S1 = 0.0;
    X->Planta_S2 = 0.0;
    X->PMM_proporcional_ISE_59_S1 = 0.0;
    X->PMM_proporcional_ISE_22_S1 = 0.0;
    X->PMM_proporcional_ISE_22_S2 = 0.0;
    X->PMM_proporcional_ISE_47_S1 = 0.0;
    X->PMM_proporcional_ISE_47_S2 = 0.0;
    X->PMM_proporcional_ISE_11_S1 = 0.0;
    X->Atuador_roda_S1 = 0.0;
    XD->Planta_S1 = 0.0;
    XD->Planta_S2 = 0.0;
    XD->PMM_proporcional_ISE_59_S1 = 0.0;
    XD->PMM_proporcional_ISE_22_S1 = 0.0;
    XD->PMM_proporcional_ISE_22_S2 = 0.0;
    XD->PMM_proporcional_ISE_47_S1 = 0.0;
    XD->PMM_proporcional_ISE_47_S2 = 0.0;
    XD->PMM_proporcional_ISE_11_S1 = 0.0;
    XD->Atuador_roda_S1 = 0.0;
    SUBSYS_PREINIT[1] = FALSE;
}
}

static void Init_Application_Data ()
{
    RT_INTEGER cnt;

    /* Declare %var/varblk initialization data */

    /* External outputs initialization. */
    sys_extout.controle_efetivo = -EPSILON;
    sys_extout.atenua = -EPSILON;
    sys_extout.controle_corretor = -EPSILON;

    /* Subsystem outputs initialization. */
    ssl_outtr = &subsys_1_out[0];
    ssl_outw = &subsys_1_out[1];
    subsys_1_out[0].controle_corretor = -EPSILON;
    subsys_1_out[0].diferenca_controles = -EPSILON;
    subsys_1_out[0].controle_efetivo = -EPSILON;
    subsys_1_out[0].Planta_graus = -EPSILON;
    subsys_1_out[1].controle_corretor = -EPSILON;

```

```

    subsys_1_out[1].diferenca_controles = -EPSILON;
    subsys_1_out[1].controle_efetivo = -EPSILON;
    subsys_1_out[1].Planta_graus = -EPSILON;

    ss2_outtr = &subsys_2_out[0];
    ss2_outw = &subsys_2_out[1];
    subsys_2_out[0].Controle_Pid = -EPSILON;
    subsys_2_out[0].atenua = -EPSILON;
    subsys_2_out[0].sentido_da_transicao = -EPSILON;
    subsys_2_out[1].Controle_Pid = -EPSILON;
    subsys_2_out[1].atenua = -EPSILON;
    subsys_2_out[1].sentido_da_transicao = -EPSILON;

    for( cnt=0; cnt<NUMOUT; cnt++ ){
        ExtOut[cnt] = -EPSILON;
    }

    SUBSYS_PREINIT[1] = SUBSYS_INIT[1];
    RESETC = FALSE;

    /***** Continuous Subsystem's initialization. *****/

    subsys_1_init(ss1_outw, subsys_1_states, &subsys_1_info);
    *ss1_outtr = *ss1_outw;
    SUBSYS_PREINIT[2] = SUBSYS_INIT[2];
    subsys_2(&subsys_2_in, ss2_outw);

}

/*-----*
 *-- SCHEDULER --*
 *-----*/

/** Scheduler Data **/

enum SUBSYSTEM_TYPE { CONTINUOUS, PERIODIC, ENABLED_PERIODIC,
                      TRIGGERED_ASYNC, TRIGGERED_ANT,
                      TRIGGERED_ATR,
                      TRIGGERED_SAF, NONE };
static const RT_DURATION SCHEDULER_INTERVAL
=
                      (RT_DURATION)
(1.0/SCHEDULER_FREQ);
static const enum SUBSYSTEM_TYPE TASK_TYPE
[NTASKS+1] =
    {NONE, CONTINUOUS, PERIODIC};
static const enum TASK_STATE_TYPE INITIAL_TASK_STATE
[NTASKS+1] =
    {UNALLOCATED, IDLE, IDLE};
static const RT_INTEGER START_COUNT
[NTASKS+1] =
    {0, 0, 0};

```

```

static const RT_INTEGER          SCHEDULING_COUNT
[NTASKS+1] =
    {0, 0, 0};
static const RT_INTEGER          OUTPUT_COUNT
[NTASKS+1] =
    {0, 0, 0};

static long int                  TIME_COUNT;
static RT_DURATION               ELAPSED_TIME;
static RT_INTEGER                TSK;
static RT_INTEGER                SCHEDULER_STATUS;
static RT_INTEGER                CURRENT_PRIORITY = NTASKS+1;
static RT_INTEGER                LEVEL           = 0;
static RT_INTEGER                READY_COUNT;
static RT_INTEGER                READY_QUEUE [NTASKS+1];
static RT_BOOLEAN                DISPATCH     [NTASKS+1];
static RT_INTEGER                PRIORITY    [NTASKS+1];
static volatile RT_INTEGER       DISPATCH_COUNT;

struct TCB_TYPE
{
    enum SUBSYSTEM_TYPE          TASK_TYPE;
    RT_BOOLEAN                   ENABLED;
    RT_INTEGER                   START;
    RT_INTEGER                   START_COUNT;
    RT_INTEGER                   SCHEDULING_COUNT;
    RT_INTEGER                   OUTPUT;
    RT_INTEGER                   OUTPUT_COUNT;
    RT_BOOLEAN                   DS_UPDATE;
    RT_BOOLEAN                   EDGE_TRIGGER;
};
static struct TCB_TYPE           TCB [NTASKS+1];

/* Work area side indices for subsystems. */
static RT_INTEGER                SSWORKSIDE [NTASKS+1];
static RT_INTEGER                SSREADSIDE;

unsigned short                   SCHEDULER_STATE;

#define Queue_Task(NTSK) \
    READY_COUNT++; \
    READY_QUEUE[READY_COUNT] = NTSK; \
    DISPATCH[NTSK] = TRUE; \
    TASK_STATE[NTSK] = RUNNING

#define Signal_An_Error(NTSK) \
    if( ERROR_FLAG[NTSK] == OK ) Error( NTSK, TIME_OVERFLOW ); \
    else Error( NTSK, ERROR_FLAG[NTSK] );

void Update_Outputs( RT_INTEGER NTSK )
{
    SSREADSIDE = SSWORKSIDE[NTSK];
    SSWORKSIDE[NTSK] = 1 - SSREADSIDE;
    switch(NTSK) {

```

```

        case 1 :
            ss1_outr = &subsys_1_out[SSREADSIDE];
            ss1_outw = &subsys_1_out[1 - SSREADSIDE];
            break;
        case 2 :
            ss2_outr = &subsys_2_out[SSREADSIDE];
            ss2_outw = &subsys_2_out[1 - SSREADSIDE];
            break;

        default:
            break;
    }
    return;
}

void System_Extout_Copy() {
    ExtOut[0] = ss1_outr->controle_efetivo;
    ExtOut[1] = ss2_outr->atenua;
    ExtOut[2] = ss1_outr->controle_corretor;
}

void Update_DS_With_Externals() {

}

void Init_Scheduler()
{
    RT_INTEGER NTSK;
    for( NTSK=1; NTSK<=NTASKS; NTSK++ ) {
        TCB[NTSK].TASK_TYPE          = TASK_TYPE[NTSK];
        TCB[NTSK].ENABLED            = FALSE;
        TCB[NTSK].START              = START_COUNT[NTSK];
        TCB[NTSK].START_COUNT        = START_COUNT[NTSK];
        TCB[NTSK].SCHEDULING_COUNT   = SCHEDULING_COUNT[NTSK];
        TCB[NTSK].OUTPUT              = OUTPUT_COUNT[NTSK];
        TCB[NTSK].OUTPUT_COUNT       = OUTPUT_COUNT[NTSK];
        TCB[NTSK].EDGE_TRIGGER       = FALSE;
        TASK_STATE[NTSK]              = INITIAL_TASK_STATE[NTSK];
        DISPATCH[NTSK]               = FALSE;
        ERROR_FLAG[NTSK]              = 0;
        SUBSYS_INIT[NTSK]             = TRUE;
        SUBSYS_TIME[NTSK]             = 0.0;
        if(TASK_TYPE[NTSK]==TRIGGERED_ATR ||
TASK_TYPE[NTSK]==TRIGGERED_SAF ||
TASK_TYPE[NTSK]==TRIGGERED_ASYNC){
            SSWORKSIDE[NTSK] = 0;
            TCB[NTSK].DS_UPDATE = TRUE;
        } else {
            SSWORKSIDE[NTSK] = 1;
            TCB[NTSK].DS_UPDATE = FALSE;
        }
    }
}

```

```

DISPATCH_COUNT    = 0;
CURRENT_PRIORITY   = NTASKS+1;
READY_COUNT        = 0;
READY_QUEUE[0]     = 0;
READY_QUEUE[1]     = 0;
SSWORKSIDE[0]      = 0;
ERROR_FLAG[0]      = 0;
DISPATCH[0]       = FALSE;
SUBSYS_INIT[0]     = FALSE;
SCHEDULER_STATUS   = OK;
ELAPSED_TIME       = 0.0;
TIME_COUNT         = -1;
}

void SCHEDULER()
{
    register    RT_INTEGER    NTSK;
    register    RT_INTEGER    I;
               RT_INTEGER    ITSK;

    TIME_COUNT = TIME_COUNT + 1;

    /*** System Input ***/

    SCHEDULER_STATUS = External_Input();

    if( SCHEDULER_STATUS != OK ){
        return;
    }

    /*** Clear Ready Queue ***/

    READY_COUNT    = 0;
    READY_QUEUE[1] = 0;

    /*** Task Scheduling ***/

    for( NTSK=NTASKS; NTSK>=1; NTSK-- ){

        switch( TASK_STATE[NTSK] ){
            case IDLE :

                switch( TCB[NTSK].TASK_TYPE ){
                    case CONTINUOUS :
                    case PERIODIC :
                        if( TCB[NTSK].START == 0 ){
                            Queue_Task(NTSK);
                            Update_Outputs(NTSK);
                            TCB[NTSK].START =
TCB[NTSK].SCHEDULING_COUNT;
                        } else {

```

```

        TCB[NTSK].START = TCB[NTSK].START - 1;
    }
    break;

case ENABLED_PERIODIC :
    if( !TCB[NTSK].ENABLED ){
        TASK_STATE[NTSK] = BLOCKED;
    }else if( TCB[NTSK].START == 0 ){
        Queue_Task(NTSK);
        Update_Outputs(NTSK);
        TCB[NTSK].START = TCB[NTSK].SCHEDULING_COUNT;
    }else{
        TCB[NTSK].START = TCB[NTSK].START - 1;
    }
    break;

case TRIGGERED_ASYNC :
    if( TCB[NTSK].OUTPUT == 0 ){
        Update_Outputs(NTSK);
        TASK_STATE[NTSK] = BLOCKED;
        if( TCB[NTSK].START == 0 ){
            Queue_Task(NTSK);
            TCB[NTSK].START = 1;
        }
    }
    break;

case TRIGGERED_ANT :
    if( TCB[NTSK].START == 0 ){
        Queue_Task(NTSK);
        Update_Outputs(NTSK);
        TCB[NTSK].START = 1;
    }
    break;

case TRIGGERED_ATR :
    if( TCB[NTSK].OUTPUT == 0 ){
        Update_Outputs(NTSK);
        TASK_STATE[NTSK] = BLOCKED;
        if( TCB[NTSK].START == 0 ){
            Queue_Task(NTSK);
            TCB[NTSK].OUTPUT =
TCB[NTSK].OUTPUT_COUNT;
            TCB[NTSK].START = 1;
        }
    } else {
        TCB[NTSK].OUTPUT = TCB[NTSK].OUTPUT - 1;
    }
    break;

case TRIGGERED_SAF :
    if( TCB[NTSK].OUTPUT == 0 ){
        Update_Outputs(NTSK);
        TASK_STATE[NTSK] = BLOCKED;

```

```

        if( TCB[NTSK].START == 0 ){
            Queue_Task(NTSK);
            TCB[NTSK].START = 1;
        }
    }
    break;
}
break;

case RUNNING :

switch( TCB[NTSK].TASK_TYPE ){
    case CONTINUOUS :
    case PERIODIC :
        if( TCB[NTSK].START > 0 ){
            TCB[NTSK].START = TCB[NTSK].START - 1;
        } else {
            Signal_An_Error(NTSK);
            return;
        }
        break;

    case ENABLED_PERIODIC :
        if( TCB[NTSK].START > 0 ){
            TCB[NTSK].START = TCB[NTSK].START - 1;
        } else {
            Signal_An_Error(NTSK);
            return;
        }
        break;

    case TRIGGERED_ASYNC :
        if( ERROR_FLAG[NTSK] != 0 ){
            Signal_An_Error(NTSK);
            return;
        }
        break;

    case TRIGGERED_ANT :
        if( TCB[NTSK].START == 0 ){
            Signal_An_Error(NTSK);
            return;
        }
        break;

    case TRIGGERED_ATR :
        if( ( TCB[NTSK].OUTPUT > 0 ) && (TCB[NTSK].START >
0)) {
            TCB[NTSK].OUTPUT = TCB[NTSK].OUTPUT - 1;
        } else {
            Signal_An_Error(NTSK);
            return;
        }
        break;
}

```

```

        case TRIGGERED_SAF :
            if( ERROR_FLAG[NTSK] != 0 ){
                Signal_An_Error(NTSK);
                return;
            }
            break;
    }
    break;

    case BLOCKED :

        switch( TCB[NTSK].TASK_TYPE ){
            case ENABLED_PERIODIC :
                if( TCB[NTSK].ENABLED ){
                    Queue_Task(NTSK);
                    Update_Outputs(NTSK);
                    TCB[NTSK].START =
TCB[NTSK].SCHEDULING_COUNT;
                }
                break;

            case TRIGGERED_ASYNC :
                if( TCB[NTSK].START == 0 ){
                    Queue_Task(NTSK);
                    TCB[NTSK].START = 1;
                }
                break;

            case TRIGGERED_ATR :
                if( TCB[NTSK].START == 0 ){
                    Queue_Task(NTSK);
                    TCB[NTSK].OUTPUT = TCB[NTSK].OUTPUT_COUNT;
                    TCB[NTSK].START = 1;
                }
                break;

            case TRIGGERED_SAF :
                if( TCB[NTSK].START == 0 ){
                    Queue_Task(NTSK);
                    TCB[NTSK].START = 1;
                }
                break;
        }
        break;
    }
}

/** System Output */

Update_DS_With_Externals();

System_Extout_Copy();

```

```

SCHEDULER_STATUS = External_Output ();

/**/ Update elapsed time ***/

ELAPSED_TIME = ((RT_DURATION)TIME_COUNT)*SCHEDULER_INTERVAL;

/**/ Task Input Sample and Hold ***/

for( I=READY_COUNT; I>=1; I-- ){
    TSK    = READY_QUEUE[I];
    switch (TSK){
        case 1:
            subsys_1_in.Controle_Pid = ss2_out->Controle_Pid;
            subsys_1_in.atenua = ss2_out->atenua;
            subsys_1_in.sentido_da_transicao = ss2_out->sentido_da_transicao;

            break;
        case 2:
            subsys_2_in.controle_corretor = ss1_out->controle_corretor;
            subsys_2_in.diferenca_controles = ss1_out->diferenca_controles;
            subsys_2_in.Planta_graus = ss1_out->Planta_graus;
            subsys_2_in.dzero = dczero.dzero;

            break;
        default:
            break;
    }
    SUBSYS_TIME[TSK] = ELAPSED_TIME;
}

/**/ Signal End of Critical Section ***/

if( READY_COUNT > 0 ){
    if( READY_QUEUE[1] > DISPATCH_COUNT ){
        DISPATCH_COUNT = READY_QUEUE[1];
    }
    ITSK = READY_QUEUE[READY_COUNT];
    SCHEDULER_STATE = PREEMPTABLE;
} else{
    SCHEDULER_STATE = PREEMPTABLE;
    return;
}

/**/ Task Dispatching ***/

while( ITSK < CURRENT_PRIORITY  && ITSK <= DISPATCH_COUNT ){
    Disable;
    if( DISPATCH[ITSK] ){
        LEVEL++;
        PRIORITY[LEVEL] = CURRENT_PRIORITY;

```

```

CURRENT_PRIORITY = ITSK;
DISPATCH[ITSK]  = FALSE;
Enable;
switch (ITSK){
    case 1 :
        if( TIME_COUNT == 0 ) {
            ssl_rinfo[0] = (RT_FLOAT)SUBSYS_TIME[1];
            subsys_1(&subsys_1_in, ssl_outw,
subsys_1_states, &
                subsys_1_info);
        }
        rungekutta2(9,
            (RT_FLOAT *)(&subsys_1_states[0]),
            (RT_FLOAT *)(&subsys_1_states[1]),
            (RT_FLOAT)SUBSYS_TIME[1],
            0.01);
        ssl_iinfo[1] = 0;
        ssl_iinfo[2] = 0;
        ssl_iinfo[3] = 1;
        ssl_iinfo[4] = 7;
        ssl_rinfo[0] = (RT_FLOAT)SUBSYS_TIME[1];
        subsys_1(&subsys_1_in, ssl_outw, subsys_1_states,
&
            subsys_1_info);
        break;
    case 2 :
        subsys_2(&subsys_2_in, ss2_outw);
        break;

    default : break;
}
Disable;
if( ERROR_FLAG[ITSK] == OK ){
    TASK_STATE[ITSK] = IDLE;
}
CURRENT_PRIORITY = PRIORITY[LEVEL];
LEVEL--;
}
Enable;
ITSK++;
}
if( ITSK > DISPATCH_COUNT ){
    DISPATCH_COUNT = 0;
}
}

/*-----*
*-- MAIN --*
*-----*/

main()
{

```

```
/** Initialize Scheduler */
Init_Scheduler();
/* Initialize Application Data */
Init_Application_Data();
/** User Initialization */
Implementation_Initialize (&ExtIn[0], NUMIN, &ExtOut[0], NUMOUT,
                          SCHEDULER_FREQ);

/** Start Scheduler */
SCHEDULER_STATUS = Background ();

/** User Termination */
Implementation_Terminate();
exit(0);
}
```