



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2015/04.29.19.28-TDI

**INVESTIGAÇÃO DA FUSÃO DA TÉCNICA DE
CONTROLE FUZZY E/OU SDRE/ H_{∞} VISANDO
PROJETAR UM CONTROLADOR ROBUSTO PARA UM
SATÉLITE RÍGIDO-FLEXÍVEL**

Severino Pinto dos Santos

Tese de Doutorado do Curso de Pós-Graduação em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle, orientada pelo Dr. Luiz Carlos Gadelha de Souza, aprovada em 28 de maio de 2015.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34P/3JDEP6H>>

INPE

São José dos Campos

2015

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

**COMISSÃO DO CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO
DA PRODUÇÃO INTELECTUAL DO INPE (DE/DIR-544):****Presidente:**

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

Membros:

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

Dr. Amauri Silva Montes - Coordenação Engenharia e Tecnologia Espaciais (ETE)

Dr. André de Castro Milone - Coordenação Ciências Espaciais e Atmosféricas
(CEA)

Dr. Joaquim José Barroso de Castro - Centro de Tecnologias Espaciais (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)

Clayton Martins Pereira - Serviço de Informação e Documentação (SID)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Simone Angélica Del Ducca Barbedo - Serviço de Informação e Documentação
(SID)

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

EDITORAÇÃO ELETRÔNICA:

Marcelo de Castro Pazos - Serviço de Informação e Documentação (SID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2015/04.29.19.28-TDI

**INVESTIGAÇÃO DA FUSÃO DA TÉCNICA DE
CONTROLE FUZZY E/OU SDRE/ H_{∞} VISANDO
PROJETAR UM CONTROLADOR ROBUSTO PARA UM
SATÉLITE RÍGIDO-FLEXÍVEL**

Severino Pinto dos Santos

Tese de Doutorado do Curso de Pós-Graduação em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle, orientada pelo Dr. Luiz Carlos Gadelha de Souza, aprovada em 28 de maio de 2015.

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34P/3JDEP6H>>

INPE

São José dos Campos

2015

Dados Internacionais de Catalogação na Publicação (CIP)

Santos, Severino Pinto dos.
Sa59i Investigação da fusão da técnica de controle fuzzy e/ou SDRE/ H_∞ visando projetar um controlador robusto para um satélite rígido-flexível / Severino Pinto dos Santos. – São José dos Campos : INPE, 2015.
xxvi + 194 p. ; (sid.inpe.br/mtc-m21b/2015/04.29.19.28-TDI)

Tese (Doutorado em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2015.
Orientador : Dr. Luiz Carlos Gadelha de Souza.

1. Controle fuzzy. 2. Controle de atitude. 3. Controle SDRE.
4. Controle de algoritmos genéticos. I.Título.

CDU 629.783:629.7.05

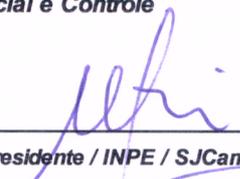


Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](https://creativecommons.org/licenses/by-nc/3.0/).

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

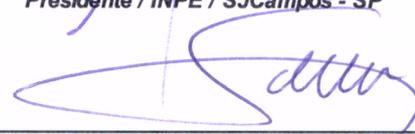
Aprovado (a) pela Banca Examinadora
em cumprimento ao requisito exigido para
obtenção do Título de **Doutor(a)** em
**Engenharia e Tecnologia Espaciais/Mecânica
Espacial e Controle**

Dr. Mario Cesar Ricci



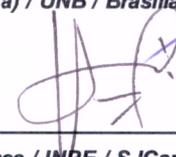
Presidente / INPE / SJCampos - SP

Dr. Luiz Carlos Gadelha de Souza



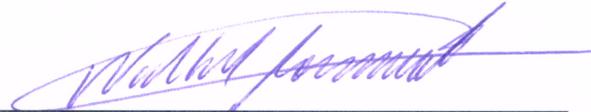
Orientador(a) / UNB / Brasília - DF

Dr. Hans-Ulrich Pilchowski



Membro da Banca / INPE / SJCampos - SP

Dr. Nilton Itiro Morimoto



Convidado(a) / PSI/EPUSP / São Paulo - SP

Dr. Valdir Gil Pillat



Convidado(a) / UNIVAP / São José dos Campos - SP

Este trabalho foi aprovado por:

() maioria simples

unanimidade

Título: "Investigação da fusão da técnica de controle Fuzzy e/ ou SDRE/H[∞] visando projetar um controlador robusto para um satélite rígido-flexível."

Aluno (a): **Severino Pinto dos Santos**

São José dos Campos, 28 de Maio de 2015

“Quando estás certo, ninguém se lembra de ti; quando estás errado, ninguém te esquece”.

Provérbio irlandês

AGRADECIMENTOS

Agradeço a Deus A quem devo a vida e todas as minhas realizações.

Meus agradecimentos ao Instituto Nacional de Pesquisas Espaciais, na pessoa de seus diretores, professores e dedicados funcionários.

Em especial, agradeço aos meus orientadores pela constante presença, paciência e valiosas orientações que me dedicaram, durante todo o meu curso neste importante Instituto; meu reconhecimento e admiração aos senhores:

Professor Doutor Luiz Carlos Gadelha de Souza e

Professor Doutor Márcio Porto Basgalupp

Estendo minha gratidão a todos os professores deste Instituto que, em várias ocasiões, dissiparam minhas dúvidas e desconhecimentos; muito agradecido senhores:

Professores Doutores:

Evandro Marconi Rocco;

Hélio Koiti Kuga;

Hans Ulrich Pilchowski;

Ijar Milagre da Fonseca;

Marcelo Lopes de Souza;

Mário Cesar Ricci;

Valdemir Carrara e

Eduardo Fávoro Pacheco da Luz (Computação)

Meus agradecimentos aos colegas, funcionários e amigos:

Alain, Eloy, Willer, Walter, Jairo, Pierre, Wagner, Rafael, José, Ximena,
sr^a Valdirene e sr^a Queila,

pelo incentivo, apoio e preciosa amizade.

Minha especial gratidão a minha esposa e filhos

Rosilda

Roberto e

Rossan,

pelo incentivo, constante apoio e compreensão.

RESUMO

Consultando-se a literatura sobre controle de sistemas, constata-se o número sempre crescente de projetos de controle usando as técnicas de controle robusto, dentre as quais se destacam as técnicas LQR (controle linear quadrático), H_{∞} e SDRE (equação de Ricatti dependente do estado). O emprego destas técnicas na área aeroespacial é cada vez maior, principalmente para controle de satélites e plataformas espaciais. Os projetos de controle robusto usando as técnicas LQR, SDRE e H_{∞} , atualmente, apresentam o inconveniente que consiste na trabalhosa determinação por tentativa das matrizes de ponderação destas técnicas, sendo estas matrizes: \mathbf{Q} e \mathbf{R} para as técnicas LQR e SDRE e as matrizes \mathbf{W}_s e \mathbf{W}_c para a técnica H_{∞} . Este trabalho propõe minimizar o inconveniente citado acima, investigando e projetando controladores multivariáveis para o controle e estabilização de atitude satélites, usando a fusão das técnicas de controle *Fuzzy* e/ou algoritmos genéticos com as técnicas de controle LQR, SDRE e H_{∞} . Inicialmente são feitas revisões de um modelo de satélite rígido, representado pelo simulador de satélite construído pelo INPE (Instituto Nacional de Pesquisas Espaciais) e de um modelo de satélite rígido-flexível, representado pelo dispositivo SRV2, fornecido pela Quanser[®], que consiste de um cubo no qual está engastada uma haste flexível. Em seguida, usando a técnica de controle *Fuzzy*, obtêm-se de forma automática a matriz \mathbf{Q} constante, para os controladores LQR e SDRE, a matriz \mathbf{R} é mantida como uma matriz diagonal unitária. Finalmente, usando as técnicas de algoritmos genéticos, são determinadas as matrizes de ponderação de sensibilidade e de sensibilidade complementar, respectivamente \mathbf{W}_s e \mathbf{W}_c , de um controlador H_{∞} sobre o sistema rígido-flexível.

FUSION RESEARCH *FUZZY* CONTROL AND / OR SDRE / H^∞ AIMING TO DESIGN A ROBUST CONTROLLER FOR A RIGID AND RIGID-FLEXIBLE SATELLITE

ABSTRACT

Referring to the literature concerning on control systems, it can be seen an increasing number of control designs using robust control techniques, among which stands out techniques LQR (linear quadratic control), H_{∞} and SDRE (state-dependent Ricatti equation). The use of these techniques in the aerospace area is increasing, mainly to control satellites and spatial base. The robust control designs using the LQR techniques, SDRE and H_{∞} currently have the disadvantage that is the laborious determination by trial of the weighting matrices of these techniques, and the Q and R matrices for the LQR and SDRE techniques and W_S matrices and W_C for H_{∞} technique. This paper proposes minimize the inconvenience cited above, investigating and designing multivariable controllers for control and stabilization of satellites attitude, using the fusion of *Fuzzy* control techniques and / or genetic algorithms with the LQR control techniques, and SDRE H_{∞} . Initially revisions are made of a rigid satellite model, represented by the satellite simulator built by INPE (National Institute for Space Research) and a rigid-flexible satellite model, represented by SRV2 device, provided by Quanser®, consisting of a cube in which is embedded a flexible stem. Then, using *Fuzzy* control technique, one obtains automatically a constant matrix Q for the LQR and SDRE controllers. The R matrix is maintained as a unitary diagonal matrix ($R=1$)

Finally, using the techniques of genetic algorithms, determine the sensitivity and complementary sensitivity weighting matrices, respectively W_S and W_C , for H_{∞} controller for a rigid-flexible system.

LISTA DE FIGURAS

	<u>Pág.</u>
Figura 2.1 - Sistema referencial	13
Figura 2.2 - Sistemas de referência	15
Figura 2.3a,b - Sequência 3-2-1 de Euler	17
Figura 2.4 - Simulador de satélite do INPE	18
Figura 2.5 -Positivo SRV2 e haste flexível	19
Figura 2.6 – Sistema rígido-flexível	19
Figura 2.7 – Satélite sino-brasileiro CBERS-1.....	20
Figura 2.8 - Plataforma do simulador de satélite	21
Figura 2.9 - Plataforma simplificada - Simplificação da Figura (2.1)	22
Figura - 2.10 – Reapresentação da Figura 2.5, mostrando F_i e F_b e o CM	28
Figura 2.11 – Localização dos sisemas de coordenadas no dispositivo SRV2	33
Figura 2.12-Sistema rígido-flexível.....	33
Figura 3.1 - Função de pertinência triangular.....	41
Figura 3.2-Variável linguística de temperatura	42
Figura 3.3 - Estrutura básica de um sistema de controle <i>Fuzzy</i>	43
Figura 3.4-Processo de inferência f.....	46
Figura 3.5 - Regra gráfica formada por funções de pertinências.....	48
triangulares e trapezoidais	48
Figura 3.6 - Fluxograma do funcionamento do algoritmo genético.....	53
Figura 3.7 - Ilustração do funcionamento da roleta	55
Figura 3.8 - Ilustração do funcionamento da reprodução por cruzamento	58
Figura 3.9 - Ilustração da operação de mutação	59
Figura 3.10 - Configuração de controle com um grau de liberdade.....	62
Figura 3.11 - Representação das formas desejadas para S e T	66
Figura 3.12 - Norma H_{∞} de $G(s)$	66
Figura 3.13 - Sistema multivariável	67
Figura 3.14 - Planta generalizada	67
Figura 3.15 - Diagrama bloco geral	68
Figura 3.16 - Diagrama da planta aumentada	70
Figura 3.17 - Representação gráfica da função de transferência GK.....	71
Figura 3.18 - Função de ponderação W_S	72
Figura 3.19 - Função de ponderação W_C	72
Figura 3.20 - Função sensibilidade S e função Complementar T	72
Figura 3.21 - Sistema com um grau de liberdade.....	75
Figura 3.22 - Representação generalizada do sistema mostrado na Figura 3.21	75
Figura 3.23 - Frequência de ressonância do sistema rígido-flexível	78
Figura 3.24 - Diagrama Simulink do Controlador H_{∞}	79
Figura 3.25 - a) Deslocamento angular e b) Velocidade de deslocamento angular	80
Figura 3.26 - a) Deslocamento elástico e b) Velocidade de deslocamento elástico	81

Figura 3.27 - Valores singulares de S e T	82
Figura 3.28 - Curvas de $\sigma(S)$ e $1/ W_s $	83
Figura 3.29 - Curvas de $\sigma(T)$ e $1/ W_c $	83
Figura 3.30a,b - Mapas dos pólos do sistema: a) antes e b)após a alocação de pólos.....	92
Figura 4.1 - Diagrama Simulink do Controlador por Alocação de Pólos.....	97
Figura 4.2 - Diagrama Simulink do controlador LQR/ <i>Fuzzy</i>	98
Figura 4.3a,b - Respostas transitória de atitude - Alloc. de pólos versus LQR/ <i>Fuzzy</i>	99
Figura 4.4a,b - Resposta transitória de velocidade – Alloc. pólos versus LQR/ <i>Fuzzy</i>	100
Figura 4.5a,b - Resposta transitória de torque – Alloc. pólos versus LQR/ <i>Fuzzy</i>	100
Figura 4.6a,b – Resposta de regime permanente de atitude - Alocação de pólos versus LQR/ <i>Fuzzy</i>	101
Figura 4.7a,b –Resposta de regime permanente de velocidades- Alocação de pólos versus LQR/ <i>Fuzzy</i>	101
Figura 4.8a,b – Resposta de regime permanente de torque - Alocação pólos versus LQR/ <i>Fuzzy</i>	102
Figura 4.9 - Resposta do controlador LQR clássico	104
Figura 4.10 - Consumo de energia do controlador LQR clássico R=1	105
Figura 4.11 - Resultados do controlador LQR/ <i>Fuzzy</i> inicialmente sem controle	106
Figura 4.12 - Resultados do controlador LQR/ <i>Fuzzy</i> com controle	107
Figura 4.13 - Consumo de energia.....	108
Figura 4.14 - Respostas para a Simulação 2 da Tabela 4.2.	110
Figura 4.15 - Resposta para a Simulação 4 da Tabela 4.2	111
Figura 4.16 - Resposta para a Simulação 6 da Tabela 4.3	112
Figura 4.17 - Consumo de Energia da Simulação 2 da Tabela 4.2	113
Figura 4.18 - Consumo de Energia da Simulação 4 da Tabela 4.2.	113
Figura 4.19 - Consumo de Energia da Simulação 6 da Tabela 4.2	114
Figura 4.20 - Ângulos de atitude do controlador SDRE clássico.....	115
Figura 4.21 - Velocidades das rodas de reação Controlador SDRE clássico.	116
Figura 4.22 - Torques controlador SDRE clássico	116
Figura 4.23 a,b,c - Respostas para simulação 1 da Tabela 4.3	118
Figura 4.24a,b,c - Respostas da simulação 3 da Tabela 4.3	119
Figura 4.25a,b,c - Respostas da simulação 5 da Tabela 4.3	120
Figura 4.26- Controle <i>Fuzzy</i> do controlador SDRE/ <i>Fuzzy</i>	124
Figura 4.27 - Controlador SDRE/ <i>Fuzzy</i>	125
Figura 4.28 - Incorporação do controle <i>Fuzzy</i>	126
Figura 4.29a,b - Comparação entre Respostas – Ângulos de atitude	127
Figura 4.30a,b - Comparação entre Respostas –Velocidades das rodas de reação	128
Figura 4.31a,b - Comparação entre Respostas – Torques	128
Figura 4.32a,b- Respostas transitórias de atitude - Alloc. de pólos versus LQR/ <i>Fuzzy</i>	129

Figura 4.33a,b - Resposta transitória de velocidade – Alloc. pólos versus LQR/ <i>Fuzzy</i>	130
Figura 4.34a,b - Resposta transitória de torque – Alloc. pólos versus LQR/ <i>Fuzzy</i>	130
Figura 4.35- Satélite Sino-brasileiro CBERS-1	133
Figura 4.36- Satélite rígido-flexível CBERS-1	133
Figura 4.37: Fluxograma do Contr. H_{INF} Auxiliado AG-	137
Figura 4.38a,b: Ângulo de atitude ϕ H_{INF}/AG original e proposto do CBERS	141
Figura 4.39a,b: Ângulo de atitude θ $H_{INFINITO}/AG$ original e proposto- CBERS	142
Figura 4.40a,b: Ângulo de atitude ψ $H_{INFINITO}/AG$ original e proposto - CBERS	142
Figura 4.41a,b: Deslocamentos elásticos q_x dos controladores original e proposto do	143
Figura 4.42a,b - Deslocamentos elásticos q_y dos controladores original e proposto	144
Figura 4.43a,b: Vibrações do painel no eixo X nos controladores original e simplificado do CBERS-1	144
Figura 4.44a,b: Vibrações do painel no eixo Y nos controladores original e proposto	145
Figura 4.45: Curvas de S, $1/ W_S $, T e $1/ W_C $	146
Figura 4.46a - Variação do número de iteração gama (γ) do controlador $H_{INFINITO}/AG$ original.....	151
Figura 4.47b – Variação do tempo de execução do controlador $H_{INFINITO}/AG$ original.....	151
Figura 48c - Variação do índice de desempenho do controlador $H_{INFINITO}/AG$ original.....	151
Figura 4.49a – Variação do número de iteração gama (γ) do controlador $H_{INFINITO}/AG$ proposto	152
Figura 4.50b – Variação do tempo de execução do controlador $H_{INFINITO}/AG$ proposto	152
Figura 4.51c - Variação do índice de desempenho do controlador $H_{INFINITO}/AG$ proposto.....	152
Figura 4.52 – Variação de W_{S1} e W_{C1} do controlador $H_{INFINITO}/AG$ original ..	153
Figura 4.53 – Variação de W_{S1} e W_{C1} do controlador $H_{INFINITO}/AG$ proposto	153

LISTA DE TABELAS

	<u>Pág.</u>
Tabela 3. 1 - Controle <i>Fuzzy</i>	49
Tabela 4.1- Relações entre Q e R	109
Tabela 4.2 - Relações entre Q e R	109
Tabela 4.3 Relações entre Q e R no SDRE/ <i>Fuzzy</i>	117
Tabela 4.4- Execuções do programa original	148
Tabela 4.5- Execuções do programa simplificado	149
Tabela 4.6- Valores médios de gama, tempo de execução e desempenho IP	150

LISTA DE SIGLAS E ABREVIATURAS

-Siglas:

AG	Algoritmos genéticos (Portugues);
GA	Algoritmos genéticos (Ingles);
CM	Centro de massa;
GPS	Controle global de posição através de satélite
INPE	Instituto Nacional de Pesquisas Espaciais;
LQR	Regulador linear quadrático;
LQG	Linear quadrático gaussiano;
LF	Lógica <i>Fuzzy</i> ;
PID	Controlador proporcional, integral e derivativo;
SRV02	Dispositivo simulador de sistema rigido-flexível da Qanser®;
SDRE	State-Dependent Riccati Equation
SID	Serviço de Documentação e Informação
SPG	Serviço de Pós-Graduação
TDI	Teses e Dissertações Internas

-Abreviatura

calc.	Cálculo;
aloc.	Alocação;

LISTA DE SÍMBOLOS

A	Matriz dinâmica de um sistema;
A_T	Margem de estabilidade desejada;
A\cupB	União do conjunto A com o conjunto B ;
A\capB	Interseção do conjunto A com o conjunto B ;
B	Matriz de controle;
B_{eq}	Coefficiente de atrito viscoso
$b_{1,2,3}$	Vetores unitários do sistema de coordenadas B;
\bar{B}	Sistema de coordenada fixa no satélite;
C	Matriz de controle;
D	Matriz de transferência direta;
D	Comprimento do arco de deflexão ;
d, d_i	Distúrbio
diag	Diagonal (abreviação usada pelo Matlab);
eye()	Matriz diagonal unitária (Matlab);
Diag, diag	Matriz diagonal (símbolo do Matlab);
G(s)	Função de transferência;
$H_{INFINITO}$	Controle robusto H com norma infinita;
H_2	Controle robusto H com norma dois;
H_∞	Controle robusto H com norma infinita
$H(x,u,\lambda)$	Hamiltoniano
I	Momento de inércia
I	Matriz identidade;
IP	Índice de desempenho;
J	Símbolo da função custo;
J_h	Momento de inércia da haste;
K_{stiff}	Coefficiente de rigidez da haste;
K_{GAGE}	Fator de calibração do strain gage;
K	Símbolo de ganho de um sistema ou função;
L	Comprimento da haste;;

M_S, M_C	Constantes de ponderação para W_S e W_C ;
M_p	Sobre sinal ou "overshoot"
m	Massa da haste;
\vec{N}	Sistema de coordenada inercial;
$n_{1, 2, 3}$	Vetores unitários do sistema inercial \vec{N}
P	Conjunto solução para a matriz de Riccati;
\mathbf{P}	Planta generalizada;;
P_r	Probabilidade de cruzamento no GA;
P_m	Probabilidade de mutação no GA;
$\mathbf{Q}, \mathbf{Q}(x)$	Matrizes usadas para ponderação de estado no LQR, LQG e SDRE;
$\mathbf{R}, \mathbf{R}(x)$	Matrizes usadas para ponderação de controle no LQR, LQG e SDRE;
$R_{1,2,3}$	Rodas de reação;
$\mathbf{R}^3, \mathcal{R}$	Sistema tridimensional;
\vec{r}	Vetor de posição;
r	Posição Escalar
\mathbf{S}	Função Sensibilidade;
\mathbf{T}	Função Sensibilidade complementar;
$T_{1,2,3}$	Torque das rodas de reação R_1, R_2 e R_3 ;
$U, \{u\}$	Universo de discurso para funções de pertinência da L F;
u	Sinal de controle;
\Leftrightarrow	Se e somente se;
\mathbf{W}	Entradas exógenas da planta generalizada;
$\mathbf{W}_S, \mathbf{W}_C$	Matrizes de compensação do controle H_{INFINITO} ;
$\ \mathbf{W}_S \mathbf{T}\ _{\infty}$	Símbolo de sensibilidade mista usando a função complementar T;
$\ \mathbf{W}_S \mathbf{S}\ $	Símbolo de sensibilidade mista usando a função sensibilidade S;
t_a	Tempo de acomodação ou estabilização;
t_s	Tempo de subida
$\alpha(t)$	Deflexão elástica da haste;
δ_s	Erro de regime permanente de w_s ;

δ_c	Erro de regime permanente de w_c ;
$v(t)$	Velocidade;
γ	Parâmetro de iteração para calculo do controlador H_{INFINITO} sub ótimo
$\gamma(t)$	Deflexão elástica total da haste;
$\omega_i, \omega(t)$	Velocidade angular;
ω_b	Frequência de W_s ;
ω_{bs}	Frequência de W_c ;
$\theta_{1,2,3}$	Ângulos de Euler;
$\theta(t)$	Deslocamento angular do servo motor
μ_A	Função de pertinência da lógica <i>Fuzzy</i> ;
$\sigma(\cdot)$	Valor singular de uma matriz;
$\bar{\sigma}(\cdot)$	Valor singular máximo de uma matriz;
∂	Símbolo de derivada parcial;
x, y, z	Eixos coordenados de R^3 ;
ϕ, θ, ψ	Ângulos de Euler do simulador de satélite;
p, q, r	Taxa de variação no tempo dos ângulos de Euler ϕ, θ, ψ ;
λ	Multiplicador de Lagrange;

SUMÁRIO

1 INTRODUÇÃO	1
1.1 Justificativas e motivações	3
1.2 Revisão bibliográfica	7
1.3 Objetivos e contribuições deste trabalho.....	10
1.4 Organização desta tese.....	11
2 REVISÃO DA MODELAGEM DOS SISTEMAS USADOS NESTE TRABALHO	13
2.1 Sistema de referência	13
2.2 Sistema de referência móvel.....	14
2.3 Sistema de referência do satélite	14
2.4 Ângulos de Euler	16
2.5 Conceitos de atitude e de movimento de atitude.....	17
2.6 Modelagem dos sistemas.....	18
2.7 Modelo do simulador de satélite.....	20
2.8 Modelo Linearizado do Simulador	30
2.9 Satélite rígido-flexível	32
2.10 Modelagem do sistema rígido-flexível	33
3 REVISÃO DAS TÉCNICAS DE CONTROLE USADAS NESTE TRABALHO	39
3.1 Lógica <i>Fuzzy</i> e elementos de controle <i>Fuzzy</i>	39
3.1.1 Sistema de controle <i>Fuzzy</i>	40
3.1.2 Classificação dos raciocínios <i>Fuzzy</i>	41
3.1.3 Operações com conjunto <i>Fuzzy</i>	42
3.1.4 Estrutura básica de um controlador <i>Fuzzy</i>	43
3.1.5 Exemplo da aplicação do método direto de Mandani.....	44
3.1.6 Controlador por lógica <i>Fuzzy</i>	47
3.2 Elementos de algoritmos genéticos.....	49
3.2.1 Definições básicas	50
3.2.2 Diferenças entre os algoritmos genéticos e os métodos determinísticos	51
3.2.3 Estrutura dos algoritmos genéticos	52
3.2.4 Representação das variáveis em algoritmos genéticos	53
3.2.5 Geração da população inicial	54
3.2.6 Avaliação da população	54
3.2.7 Mecanismo de seleção.....	55
3.2.8 Mecanismo de reprodução	57
3.2.9 Reprodução por cruzamento (crossover simples)	57
3.2.10 Reprodução por mutação	58
3.2.11 Eficiência de um algoritmo genético	59
3.2.12 Tamanho da população – N ou L	60
3.2.13 Taxa de probabilidade de cruzamento – pc.....	60
3.2.14 Taxa de probabilidade de mutação – pm	60
3.3 Conceitos básicos de controle H_{∞}	61
3.3.1 Diagrama típico de controle clássico	62
3.3.2 Relações fundamentais	63

3.3.3 Compromisso de projeto entre s e t em função de seus valores singulares	65
3.3.4 Definição de norma H_{∞} de uma função de transferência	66
3.3.5 Formulação geral do problema de controle H_{∞}	67
3.3.6 Descrição matemática do sistema.....	68
3.3.7 Sensibilidade mista.....	70
3.3.8 Escolha das funções de ponderação W_S E W_C	73
3.3.9 Controlador H_{∞} sub-ótimo	74
3.3.10 Procedimentos básicos para projeto de um controlador H_{∞}	75
3.3.11 Exemplo de controle H_{∞} de um sistema rígido flexível.....	77
3.3.12 Desempenho do <i>Mixsyn</i>	81
3.4 Fundamentos de um controlador SDRE.....	83
3.4.1 Controlador SDRE.....	85
3.4.2 Obtenção do controlador SDRE	86
3.5 Conceitos básicos de alocação de pólos.....	89
3.5.1 Alocação de pólos usando o método das matrizes companheiras.....	90
Dado o sistema a seguir:.....	90
4 CONTROLE E SIMULAÇÃO	95
4.1 Comparação dos controladores por alocação de pólos e LQR/ <i>Fuzzy</i>	95
4.1.1 Comparação do comportamento das matrizes Q e R dos.....	102
controladores LQR/ <i>Fuzzy</i> e SDRE/ <i>Fuzzy</i>	102
4.1.2 Matrizes do sistema	103
4.1.3 Resultados do controlador lqr clássico (matriz q obtida por tentativas). 104	
4.1.4 Resultados de um controlador LQR/ <i>Fuzzy</i>	105
4.1.5 Verificação de relação entre Q e R	108
4.2 Controlador SDRE clássico e SDRE/ <i>Fuzzy</i>	114
4.2.1 Resultados do Controlador SDRE.....	115
4.2.2 Verificação de relação entre Q e R no controlador SDRE/ <i>Fuzzy</i>	117
4.2.3 Análise das respostas dos controlador LQR/ <i>Fuzzy</i>	121
4.2.4 Controladores SDRE convencional e SDRE/ <i>Fuzzy</i>	123
4.2.5 Funcionamento do controlador SDRE/ <i>Fuzzy</i>	123
4.2.6 Diferença entre os controladores SDRE e SDRE/ <i>Fuzzy</i>	124
4.2.7 Comparação dos resultados entre controladores SDRE e	127
SDRE/ <i>Fuzzy</i>	127
4.2.8 Considerações sobre os resultados	129
4.3 Controlador H_{∞} convencional e H_{∞}/AG	132
4.3.1 Modelagem do satélite rígido-flexível	132
4.3.2 Controlador H_{∞} original auxiliado por algoritmos genéticos	139
4.3.3 Desenvolvimento do controlador H_{∞}/AG proposto	139
4.3.4 Robustez do controlador H_{∞}/AG proposto.....	145
4.3.5 Ganhos obtidos com o programa simplificado.....	147
REFERÊNCIAS BIBLIOGRÁFICAS	159
APÊNDICE A - PROGRAMAS DE CONTROLE	167
APÊNDICE B	193

1 INTRODUÇÃO

Alguns dos grandes desafios na ciência de controle são o desenvolvimento e estabelecimento de uma técnica capaz de controlar um determinado sistema físico real de forma determinística e precisa; as dificuldades começam pelo modelamento sempre imperfeito que não capta todas as características do sistema a ser modelado, quer por desconhecimento do projetista, quer por simplificações intencionais, visando reduzir a complexidade matemática decorrente de um modelamento mais completo. Usualmente, esta falta de exatidão na construção do modelo é contornada pelo emprego de conceitos probabilísticos, ou seja, considerando imprecisões e incertezas como ruído branco, adoção de hipóteses simplificadoras, tais como desprezo de algum parâmetro, adoção de constantes e/ou parâmetros arbitrários; ao aplicar o controle sobre estes modelos imperfeitos, chegam-se a resultados aceitáveis após várias simulações, trocando-se em cada simulação, valores de algumas constantes ou de algum parâmetro, ou seja, obtêm-se resultados aceitáveis por tentativa e erro. Esta realidade, atualmente, ocorre com todas as técnicas de controle mais utilizadas conforme será mostrada na próxima seção. No contexto aeroespacial, tem-se ainda a realidade atual representada pelo aumento do emprego de satélites em missões mais complexas, com o conseqüente aumento de equipamentos e das estruturas destes artefatos.. Como aumento da estrutura do satélite, pode-se citar maiores painéis solares, que trazem como uma das conseqüências indesejáveis o aumento de vibrações que, além de deteriorar precocemente toda a estrutura pode comprometer o desempenho do controle.

Embora, vibrações e desvio de atitude sejam causadas por diferentes fenômenos naturais, tais como: vento solar, gradiente de gravidade, impacto de meteoritos, etc., comumente as vibrações surgem pelas ações de controle exercidas sobre o satélite. Daí a necessidade de um sistema de controle e estabilização que mantenha a atitude desejada do satélite e, ao fazer correções quando necessárias, não produza vibrações excessivas e prolongadas. Muitos

trabalhos, buscando um controle sem introduzir grandes vibrações, são encontrados na literatura de controle (KAPLAN, 1976; JUNKINS; TUNER, 1986; TSIOTRAS, 1996).

O objetivo deste trabalho, além de seu foco principal que é propor o uso de lógica *Fuzzy* para otimizar projetos de controladores LQR e SDRE, é dar continuidade natural aos trabalhos de Fonseca (2011), Gonzales (2009) e Cardenas (2005). Fonseca usou a técnica de controle alocação de pólos para controlar o simulador de satélite, usando o modelo linearizado. Neste trabalho foram usados os resultados obtidos por Fonseca para comparar com os resultados de um controlador LQR em que a matriz de ponderação \mathbf{Q} foi determinada pela técnica *Fuzzy*. Gonzales (2009), seguindo os resultados obtidos por Campa (2002), aplicou a técnica SDRE (State-Dependent Riccati Equation ou Equação de Riccati dependente do tempo) de controle através do Simulink[®] para controlar um simulador de satélite, usando o seu modelo não linear. Cardenas (2005) aplicou os conceitos de algoritmos genéticos, (GA) para determinar todos os componentes das funções de transferência dos compensadores usados no controlador H_{INFINITO} . Inspirado no trabalho de Cardenas, serão aplicados algoritmos genéticos (AG) para determinar apenas os parâmetros M_S , ω_s , M_C e ω_{bc} e não todos os parâmetros conforme feito pelo citado autor; com esta simplificação procura-se reduzir a carga temporal de controle. Especificamente, de Fonseca (2011) foi usado o modelo linear do simulador existente no INPE, que também foi desenvolvido por vários autores: Snider (2010), Souza e Arena (2013). Sobre este modelo aplicamos um controle usando o regulador quadrático linear (LQR) em que a matriz de ponderação de estado \mathbf{Q} foi determinada por um controlador *Fuzzy*, sendo esta inovação parte do escopo desta tese. De Gonzales (2009) e Campa (2002) utilizou-se o controlador SDRE, em que os citados autores determinaram a matriz \mathbf{Q} de forma convencional (tentativa e erro). A contribuição deste trabalho consiste na utilização de um controlador *Fuzzy* para determinar \mathbf{Q} de forma automática. Com relação ao trabalho de Cardenas (2005), a contribuição foi de

apenas simplificação que consiste de, ainda usando GA, deixar menos parâmetros a ser determinada pelo programa, tendo em vista que os parâmetros de projeto já estão determinados pelo projetista; com isto obtém-se maior rapidez de controle.

1.1 Justificativas e motivações

As técnicas de controle dominantes usuais e encontradas na literatura de controle, conforme Dafta (2003), são representadas pelos controladores: proporcional integral derivativo (PID), regulador linear quadrático (LQR), controlador linear quadrático gaussiano (LQG), controlador usando a equação de Riccati dependente do estado (SDRE) e os controladores robustos, representados pelas técnicas H_2 e H_{∞} , sendo que esta última técnica se destaca por admitir o uso de incertezas, conforme Skogestad (2001) e Maciejowski (1991). Não trataremos da técnica H_2 nesta tese.

O controle proporcional integral derivativo (PID) clássico é uma técnica muito usada nas mais diversas áreas e, principalmente em controle de plantas e equipamentos industriais, como robôs e máquinas operatrizes; a correta aplicação desta técnica exige algumas características do sistema a ser controlado, como linearidade, embora, na prática industrial o controlador PID seja aplicado em sistemas não lineares, desde que seja escolhido como ponto de trabalho, uma região mais plana da curva de resposta do processo e sejam usados pequenos sinais, determinados por experimentações.

Outra exigência da técnica PID é a necessidade de adequada sintonia de seus parâmetros proporcional, integral e derivativo (ASTROM; HAGGLUND, 1994).

O controlador LQR, o qual é uma técnica que procura minimizar uma função de custos, obedecendo às equações do sistema, para apresentar bons resultados, deve ser aplicado em sistemas lineares, ou linearizados, admitindo-se que todos os estados estejam disponíveis para realimentação e na ausência de

distúrbios e incertezas de qualquer natureza; uma dificuldade atual encontrada no projeto do controlador LQR é a escolha das matrizes de ponderação de custo de estado \mathbf{Q} e de ponderação de custo de controle \mathbf{R} , que, atualmente, são feitas por tentativa e erro até obter-se uma resposta aceitável, considerando que os procedimentos determinísticos existentes para a determinação destas matrizes não são precisos, ver Anderson e Moore (2007).

O controlador LQG, embora seja um controlador que considera a presença de ruídos e de incertezas, apresenta inconvenientes análogos aos do controlador LQR, ou seja, as necessidades de determinações, por tentativa e erro, das matrizes de ponderação de estado e de controle. Neste trabalho é proposto usar controladores *Fuzzy* para determinar de forma automática a matriz \mathbf{Q} .

O controle robusto H_{∞} , embora não tenha um bom desempenho na presença de não linearidades, conforme Maciejowski (1991), apresenta desempenho superior aos controladores tratados anteriormente; contudo também depende da experiência e habilidade do projetista para escolher os parâmetros que otimizam o desempenho deste método.

Resumindo, vê-se que a grande desvantagem das técnicas de controle mencionadas anteriormente é a inexistência atual de um procedimento sistemático, preciso e simples para determinar as matrizes de ponderação dos parâmetros que otimizam o desempenho de cada técnica, ver Anderson e Moore (2007), citado anteriormente.

Para suprir as deficiências citadas anteriormente, para os controladores LQR e SDRE, a exemplo do que já é feito para determinar os coeficientes dos controladores PID (PETERMAN, 1997) propõem-se a usar controladores *Fuzzy* para determinar automaticamente a matriz de ponderação \mathbf{Q} ; a matriz \mathbf{R} , que também está presente na função custo da técnica LQR, é considerada como uma matriz unitária diagonal.

É comum encontrar na literatura de controle atual o emprego da lógica *Fuzzy* (ZADEH, 1968) como um controle auxiliar destinado a otimizar o controle

principal: é o caso, por exemplo, do uso de lógica *Fuzzy* para determinar os coeficientes de um controlador PID.

O presente trabalho, inspirado em pesquisas que buscaram objetivos semelhantes (GOLDBERG, 1995; PETERMANN, 1997; CARDENAS, 2005), investiga o projeto de um controlador robusto multi-variável para a estabilização e controle de atitude de um pequeno satélite rígido-flexível, através da fusão das técnicas de controle usando algoritmo genético e controle Hinfinito. Aqui o termo rígido-flexível está colocado em seu sentido mais amplo, não significando apenas satélite, mas também braços robóticos, hastes, e estruturas de um modo geral.

Controle de atitude de satélite, que é o processo de orientar o satélite em uma direção fixa e pré-determinada, embora não seja um tema novo, continua sendo importante, tendo em vista a grande quantidade de trabalhos relativamente recentes sobre o assunto (FOSSSEN, 2011; SANIAL; BARAI, 2012; MENG; ZHANG, 2013). A razão disto é que estes artefatos, em suas órbitas, devem estar apontados para alguma referência tal como um ponto fixo na Terra, no Sol ou em alguma outra estrela; é sabido que uma vez o satélite orientado para uma determinada referência, esta orientação não permanece fixa, desviando-se com o transcorrer do tempo devido às ações dos agentes externos na forma de torques: torque aerodinâmico, torque de gradiente gravitacional, torque de à pressão solar e torque magnético (WERTZ, 1978); o torque causado por impacto de meteoritos é desprezível em relação aos anteriores, embora seja um fato real (HUGUES, 1986). O efeito destes agentes é tanto maior quanto menor for a massa do satélite; daí a necessidade de se monitorar e controlar os parâmetros relacionados com sua atitude de forma cada vez mais precisa. Deve ser levado em consideração que, devido à necessidade de uma maior precisão requerida por algumas missões, tais como levantamento fotográfico, fornecimento de posição (GPS), entre outras atividades, o problema de controle de atitude torna-se mais crítico. Em função desta realidade é justificável que pesquisadores continuem buscando o

aperfeiçoamento das técnicas de controle existentes ou mesmo técnicas novas para o controle e estabilização de atitudes de satélites. Com estes objetivos muitos trabalhos são encontrados na literatura especializada (ZHOU, 1995; NEECK, 2004).

É tendência atual a inclusão nos satélites de estruturas mais leves como painéis solares, braços robóticos e antenas cada vez maiores e, portanto, mais flexíveis, fazendo-se necessário considerar o efeito das flexões no modelamento destas estruturas. Assim, neste trabalho será considerado também um modelo linearizado de um satélite contendo parte rígida e parte flexível, dando como resultado um modelo híbrido (PETERMANN, 1997; FENILI; SOUZA, 2004), que seja capaz de apresentar boa precisão mesmo para grandes variações de seus ângulos de atitude. Sobre este modelo linear, a título de exemplo, será aplicado um controle H_{infinito} convencional, no qual os parâmetros M_s e M_c de suas matrizes são determinados por tentativa e erro; os parâmetros M_s e M_c são componentes das matrizes de ponderação do controle H_{INFINITO} . Usando o mesmo modelo será pesquisado um controle H_{INFINITO} no qual os referidos parâmetros venham a ser obtidos por algoritmos genéticos, seguindo a direção de trabalhos existentes (SKOGESTAD, 2007; CARDENAS, 2005; ZHOU, 1995). Este trabalho é uma simplificação do trabalho de Cardenas (2007) por procurar a determinação apenas das constantes M_T e M_S e da largura de banda ω_b e ω_s , obtendo com esta simplificação maior rapidez do controle.

1.2 Revisão bibliográfica

Os fundamentos e os conceitos matemáticos básicos utilizados no estudo de controle de atitude de satélites artificiais são encontrados em Junkins e Youdan (1983), Wertz (1978), Kaplan (1976) e Hugues (1986). O problema de modelagem e controle de atitude de satélites artificiais com estruturas flexíveis é objeto de trabalho de vários de pesquisadores, entre os quais citamos Meirovitch (1970), Kwakernaak (1986), Junkins e Youdan (1983) e Carrara (2006), que investigam a atitude e também os efeitos de vibrações sobre plataforma com apêndice flexível em movimento, procurando controlar sua atitude e reduzir os efeitos das vibrações.

Existem duas abordagens para a obtenção das equações da dinâmica destas estruturas: o método clássico e o método que utiliza a energia do sistema, denominado de lagrangeano (MEIROVITCH, 1970; JUNKINS; YODAN, 1983; FRIEDLAND, 1986). O método clássico tem a vantagem de ser transparente para o projetista e é aplicado para sistemas simples; a abordagem Lagrangiana é utilizada para estruturas mais complexa, porém não é intuitivo para o projetista, embora seja uma metodologia de fácil aplicação. Usualmente é aplicada a abordagem Lagrangiana para a obtenção das equações da dinâmica destas estruturas aeroespaciais (JUNKINS; YODAN, 1983); este autor reúne diversos trabalhos sobre modelagem usando a abordagem lagrangiana.

Uma das grandes dificuldades para se controlar movimento de um sistema rígido-flexível é o acoplamento dinâmico natural entre o movimento do corpo flexível e o movimento do corpo rígido; existem vários trabalhos na literatura de controle sobre este fenômeno (MEIROVITCH, 1970; JUNKINS; YODAN, 1970; SOUSA, 1994). Sabe-se que a modelagem matemática de um sistema físico é apenas uma aproximação do sistema real, devido a simplificações intencionais para minimizar o tratamento computacional ou devido à dificuldade de captação e expressão de todas as características do referido sistema em uma equação matemática, conforme mencionado na Seção (1.1)

(MEIROVITCH, 1970; OGATA, 2005; FRANKLIN et al, 1986). Outro importante fator de imprecisão é o fato de o sistema, para sua representação ideal por parâmetros distribuídos, precisar de um número infinito de modos e, na prática atual de modelagem, considerar-se um número finito de modos. Reduzindo-se o número de modos introduzem-se incertezas no modelo, conhecidas como incertezas não paramétricas (MEIROVITCH, 1970; KWAKERNAAK, 1972; SKOGESTAD, 2007); este último autor define incerteza paramétrica como sendo o conhecimento de um modelo físico, cuja estrutura é conhecida, inclusive, a sua ordem, mas que alguns dos parâmetros são incertos. As diversas formas de representação e parametrização de atitude de satélites, levando em conta estes fatores, são tratadas em vários autores, entre eles citam-se Maciejowski (1989), Souza (1994), Wertz (1990) e Cardenas (2005).

Devido ao fato de neste trabalho, conforme proposta inicial, serem utilizados lógica *Fuzzy* (LF) e algoritmos genéticos (GA), faz-se a seguir uma breve revisão bibliográfica sobre estas técnicas.

Em função da crescente complexidade funcional exigida dos artefatos espaciais, tornando-os cada vez mais sofisticados, é necessário encontrar métodos novos e mais eficientes para controlar suas velocidades e atitudes com maior precisão, usando algoritmos de controle que não apresente grande sobre carga ao sistema computacional. Este fato tem atraído muita atenção dos pesquisadores e gerado muitos trabalhos nestas últimas décadas (DONALD, 1997). A atenção dos pesquisadores tem se voltada para a lógica *Fuzzy* e, em alguns casos para os algoritmos genéticos (AG), conforme tratado nos próximos parágrafos.

- Logica *Fuzzy*

Nos trabalhos mais recentes o controle por lógica *Fuzzy* tem sido proposto como uma abordagem alternativa auxiliar às técnicas convencionais de controle para controlar sistemas complexos não lineares (PEDRYCZ, 2006;

ALBERTOS, 1992; ZADEH, 1965; PETERMANN, 1997); Petermann usou lógica *Fuzzy* para controlar um pequeno satélite através de jato de gás.

Em engenharia espacial a lógica *Fuzzy* tornou-se um dos mais investigados tema de pesquisa; Wang (1996) propôs um novo método de controle de rastreamento por lógica *Fuzzy*, que é usado no rastreamento da atitude de satélites reusáveis. Chiang e Jang (1994) desenvolveram um controle *Fuzzy* de atitude para o satélite Cassini; eles mostraram que o controlador *Fuzzy* pode acompanhar a referência de comando melhor que o controlador convencional, para controle de atitude do satélite. Pedrycz et al (2006), mostraram que as características não lineares dos controladores *Fuzzy* contribuem para uma melhor característica de controle para o sistema geral.

O controle *Fuzzy* foi apontado por Lee, 1990, como empregado em uma vasta gama de controles não lineares e com sucesso (LEE, 1990). Também Berenji (1993), propôs um sistema de controle para estabilização de atitude de satélite usando jatos de gás (BERENJI et AL,1993). Satyadas e Krishnakumar (1996), propôs um controle GA otimizado por lógica *Fuzzy* para controle de atitude de uma estação espacial (SATYADAS; KRISHNAKUMAR, 1996).

- Algoritmos genéticos

Com relação aos algoritmos genéticos, apresenta-se o resumo a seguir, começando por John Holland que, em 1960, como pioneiro, procurou explicar de forma rigorosa os processos de seleção e adaptação que ocorrem nos sistemas biológicos, descoberto por Darwin (HOLLAND, 1960); contribuições atualizadas sobre AG estão disponíveis em obras clássicas tais como: Goldberg, Michalewics, Mitchell (GOLDBERG, 1989; MICHALEWICS, 1996; MITCHELL, 2010). Ainda, apresentando os conceitos básicos de AG, tem-se a obra em português de Linden (2012).

Na literatura encontram-se muitos trabalhos aplicando algoritmos genéticos, dos quais citamos: em determinação de rotas Goldberg (1985), em otimização combinatória, Fourman (1986), em ordenação e classificação Oliver (2003), em

controle Krishnakumar (1982) e Goldberg e Kuo (1987). Um dos trabalhos pioneiro usado em otimização de controle aplicado em aeroespacia foi desenvolvido por Krishnakumar e Goldberg, trata-se do projeto de auto piloto e um controlador de limpador para-brisa (GOLDBERG, 1989).

Referente à modelagem e controle de estruturas flexíveis dentre outros temos o trabalho de Kwakernaak (1976), Kikuchi et al (2002). Em controle de atitude usando algoritmos genéticos encontram-se os trabalhos citados em Ross (2010) e Linden (2012).

1.3 Objetivos e contribuições deste trabalho

As contribuições desta tese consistem na criação e melhoria de procedimentos alternativos para os desenvolvimentos das técnicas de controle LQR, SDRE e H_{∞} , evitando o procedimento empírico e tedioso, usado atualmente, na determinação da matriz de ponderação \mathbf{Q} , para as técnicas LQR e SDRE e nas determinações das constantes M_S e M_C para a técnica de controle H_{∞} .

Espera-se que estas contribuições possam ser alcançadas através dos itens a seguir:

- 1)-Rever os modelos matemáticos dos sistemas tratados neste trabalho;
- 2)-Da aplicação de controladores *Fuzzy* obter, de forma automática, a matriz de ponderação \mathbf{Q} , constante, para controladores LQR e SDRE; esta matriz, atualmente, é obtida por tentativa e erro. Destaca-se este item como sendo a maior contribuição deste trabalho. A matriz de ponderação de custo de controle \mathbf{R} , em todos os controladores citados acima será considerada uma matriz diagonal unitária.
- 3)-Usar algoritmos genéticos, (GA), para determinar os parâmetros M_S , M_C , das funções de ponderação do controlador H_{∞} , que usualmente são obtidas empiricamente por tentativas.

Os procedimentos tratados nos itens acima propiciam um controle sub-ótimo e têm como principais objetivos as determinações automática das matrizes e coeficientes de ponderação aqui tratados.

O item 3 foi inspirado em trabalho existente (CARDENAS, 2005); este autor usou algoritmos genéticos para determinar todos os componentes das matrizes de ponderação \mathbf{W}_S e \mathbf{W}_C do controlador H_{INFINITO} , incluindo largura de banda e erros de regime permanente. Este procedimento torna o controle lento e seu principal inconveniente é deixar a critério do algoritmo genético as determinações dos erros de regime permanente que são parâmetros estipulados pelo projetista.

Com relação ao item 3, este trabalho é uma melhoria do trabalho de Cardenas, objetivando maior rapidez de controle e também respeitar a precisão do controle, que é um parâmetro que deve ser especificado pelo projetista. Assim nosso programa de GA deverá determinar apenas os valores das sensibilidades de pico M_S e M_C e os limites de faixa ω_b e ω_{bc} das funções de ponderação W_S e W_C , respectivamente do sistema em estudo de Zhou, (2005) e Skogestad (2009).

1.4 Organização desta tese

No **Capítulo 1** são apresentados: Introdução, Justificativa, Motivação, Revisão Bibliográfica, Organização e Objetivos deste trabalho. No **Capítulo 2**, tendo em vista que este trabalho é multi objetivo, são revistos Sistemas de referência, o modelo linearizado de um corpo rígido, representado pelo simulador de satélite construído pelo Instituto Nacional de Pesquisas Espaciais (INPE), e o modelo, com um grau de liberdade, de satélite rígido-flexível, representados pelos dispositivos SRV02 fornecido pela Quanser® e pelo satélite desenvolvido pelo INPE em parceria com a China CBERS-1. No **Capítulo 3** é apresentada uma breve revisão dos conceitos básicos de: Lógica *Fuzzy*, Algoritmos Genético, Controle H_{INFINITO} , Controle SDRE e Alocação de pólos, conceitos estes necessários para o desenvolvimento deste trabalho.

No **Capítulo 4** são tratados os problemas de simulação, controle e comparação de resultados para cada um dos modelos citados no Capítulo 3.

O **Capítulo 5** trata das conclusões finais e sugestões para continuação e aprofundamento deste trabalho.

No **Capítulo 6** estão contidas as referências bibliográficas e finalmente, nos apêndices A1, A2, A3, A4, A5, B₁ e B₂ são apresentados os programas Matlab usados neste trabalho.

2 REVISÃO DA MODELAGEM DOS SISTEMAS USADOS NESTE TRABALHO

2.1 Sistema de referência

Para descrever o movimento de um corpo é necessário inicialmente que seja definido um sistema de referência no qual os vetores de posição, velocidade e aceleração, bem como os vetores representativos de torques e forças possam ser representados. Este sistema é denominado de sistema referencial e tem como base um triedro dextrógiro de vetores unitários.

A seleção de um sistema de coordenada que possibilite uma melhor descrição de orientação de um corpo é fundamental e foi amplamente estudado por Euler, Jacobi, Hamilton, Gibbs, Rodrigues, entre outros, conforme Kaplan (1976) e Cardenas (2005). Um sistema adequado de coordenadas além de fornecer a correta descrição do satélite, deve reduzir as singularidades geométricas determinadas pelos sistemas físicos; este assunto é tratado por diversos autores (NASA, 2010; HUGUES, 1986; ZHOU et al, 1995). A seguir são revistos alguns conceitos relacionados ao movimento de um corpo em relação a seu referencial:

Vetor de posição. A Figura 2.1 mostra a representação de um vetor de posição \vec{r} de um ponto A em relação ao sistema inercial $F_i (X,Y,Z)$. Usando uma notação vetorial, tem-se: $\vec{r} = x\vec{i} + y\vec{j} + z\vec{k}$

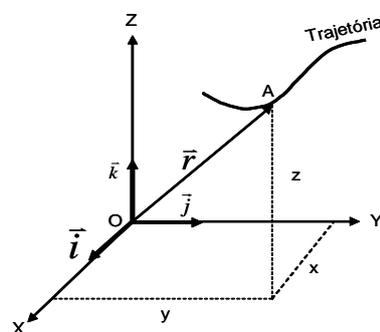


Figura 2.1 - Sistema referencial

Vetor velocidade. O vetor velocidade absoluta, conceitualmente é aceito como a variação no tempo do vetor posição, é dado matematicamente pela derivada do vetor posição quando este vetor está representado no sistema inercial. ou seja $\vec{v} = \dot{x}\vec{i} + \dot{y}\vec{j} + \dot{z}\vec{k}$.

Vetor aceleração. O vetor aceleração absoluta, conceitualmente aceito como a variação da velocidade no tempo, é matematicamente definido como a derivada segunda do vetor posição considerada no sistema inercial.

2.2 Sistema de referência móvel

Em muitos casos reais, a descrição de determinadas órbitas e movimento de corpos quaisquer se torna mais simples se forem usados dois ou mais sistemas de referência, sendo um destes sistemas tomado como sistema inercial. O objetivo do uso de mais de um sistema de referência é representar movimentos mais complexos, decompondo-os em vários movimentos mais simples que, posteriormente se somam vetorialmente para compor o movimento absoluto (em relação ao sistema inercial).

A presença, em casos reais, de mais de um sistema de coordenada relacionando um mesmo corpo, requer um sistema redundante de coordenadas e uma sistemática de relacionamento entre os sistemas, possibilitando a passagem de um sistema para o outro sem grandes trabalhos. O sistema redundante mais usado em parametrização de atitude faz uso de uma matriz ortogonal de nove elementos, denominada de matriz de cossenos diretores dos ângulos de atitudes, também denominada de matriz de atitude (HUGUES, 1986; CARDENAS, 2005).

2.3 Sistema de referência do satélite

O estudo da dinâmica de um satélite é descrita considerando-se dois sistemas de coordenadas, sendo um sistema fixo no satélite com origem em seu centro de massa, CM, e um segundo sistema tomado como referência, designado,

portanto, como sistema inercial com origem, para fins deste trabalho, no centro da Terra. Estes sistemas são formados por um conjunto de três eixos mutuamente ortogonais entre si, sendo cada conjunto representado por um conjunto de três vetores unitários que têm sua origem comum em um ponto denominado centro do sistema; são dextrógiros, ou seja, obedecem à regra da mão direita. Neste trabalho, o sistema fixo no satélite, \vec{F}_b , têm seus eixos orientados segundo os eixos geométricos principais do satélite (no caso deste trabalho o satélite é representado pelo simulador de satélite), representado pela plataforma, e com origem no centro de massa, CM, da plataforma; o sistema inercial, \vec{F}_i , para fins deste trabalho, tem seu eixo Z paralelo ao eixo sul-norte da Terra e sua origem transferida para o centro de massa da plataforma, portanto as origens dos dois sistemas são coincidentes.

Neste trabalho, quando for necessário representar relações entre sistemas, serão usados, para sistema fixo no satélite, os eixos coordenados representados por $\vec{F}_b = [x \ y \ z]$ e para sistema de referência os eixos coordenados representados por $\vec{F}_i = [X \ Y \ Z]$. As Figuras 2.2a, b mostram estes sistemas; na Figura 2.2a os sistemas estão coincidentes e na Figura 2.2b o sistema fixo no corpo, $\vec{F}_b = [x \ y \ z]$, apresenta uma atitude genérica representada pelos ângulos de atitudes ϕ , θ e ψ .

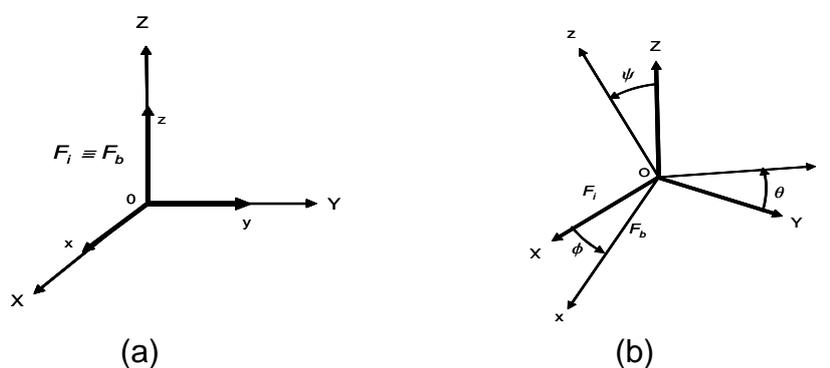


Figura 2.2 - Sistemas de referência

Estes sistemas de coordenadas, por serem dextrógiros têm as propriedades dadas a seguir:

1) A propriedade de ortogonalidade do sistema \vec{F}_i (ou \vec{F}_b) pela aplicação do produto interno de vetores a seus vetores unitários fornece as relações a seguir: $\vec{b}_i \cdot \vec{b}_j = 1$ se $\hat{i} = \hat{j}$ e $\vec{b}_i \cdot \vec{b}_j = 0$ se $\hat{i} \neq \hat{j}$ e

2) A propriedade de o sistema \vec{F}_i (ou \vec{F}_b) por ser dextrogiro, pela aplicação do produto vetorial a seus vetores unitários fornece a igualdade $\vec{b}_i \times \vec{b}_j = \varepsilon_{ijk} \vec{b}_k$ em que ε_{ijk} é dada pela relação a seguir:

$$\varepsilon_{ijk} = \begin{cases} 1 & \text{se } \hat{i}, \hat{j}, \hat{k} \text{ for uma permutação par de } 1, 2, 3 \\ -1 & \text{se } \hat{i}, \hat{j}, \hat{k} \text{ for uma permutação ímpar de } 1, 2, 3 \end{cases}$$

Lembramos que uma permutação é de ordem par ou de ordem ímpar, dependendo de se efetuar um número par ou um número ímpar de permutações entre os elementos i, j, k , partindo de sua ordem natural inicial ijk

2.4 Ângulos de Euler

As atitudes e movimentos de atitudes de aeronaves e satélites são descritas usualmente pelos ângulos de Euler. Estes ângulos são obtidos por rotações sequenciais em torno dos eixos coordenados; são possíveis doze grupos destas sequências. A sequência usada neste trabalho é dada por 3-2-1, ou seja, o sistema de referência fixo no corpo, \vec{B} , tem sua atitude definida em relação ao sistema de referência inercial, \vec{N} , através das rotações: primeiro de um ângulo ϕ em torno do eixo \vec{Z} , (eixo 3), depois de um ângulo θ em torno do eixo \vec{Y} , (eixo 2) e finalmente a rotação de um ângulo ψ em torno do eixo \vec{X} , (eixo 1). A Figura 2.3a,b ilustra os procedimentos delineado acima, para o caso

genérico; Na Figura 2.3b é mostrado a adaptação do sistema apresentado pela Figura 2.3a, b em navegação aeroespacial.

Estes ângulos descrevem os três coordenadas de atitude dos movimentos rotacionais em R^3 ; esta descrição mapeia todos os possíveis movimentos de um arco ou de um ponto genérico em R^3 . Embora, apresentando, ainda, algumas singularidades (HUGUES, 1986; CARDENAS, 2005; KUIPERS, 1998), é muito usado para a descrição de atitude satélites. No caso deste trabalho, a singularidade da sequência 3-2-1, que ocorre quando o ângulo $\theta = \pm 90^\circ$ não constitui um problema tendo em vista que fisicamente este ângulo não é permitido ocorrer, devido à limitação estrutural da plataforma (os graus de liberdade nos eixos x e y da plataforma sobre seu mancal, em forma de rótula, é menor que 90 graus).

A Figura 2.3a b ilustra os ângulos de Euler.

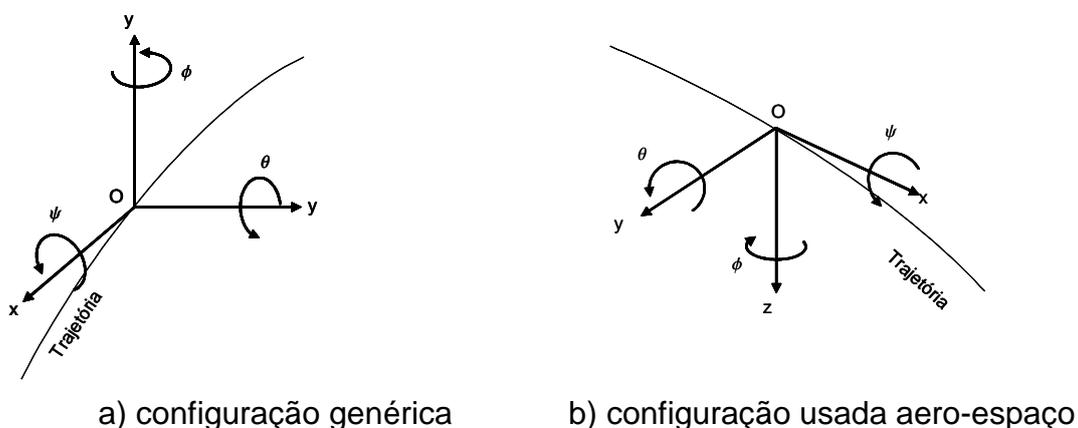


Figura 2.3a,b - Sequência 3-2-1 de Euler

2.5 Conceitos de atitude e de movimento de atitude

Atitude é conceituada como sendo a orientação do satélite em relação a um sistema de referência considerado como inercial, cuja origem pode ser o centro

da Terra; a variação desta atitude com o passar do tempo é denominado de movimento de atitude.

A manutenção da atitude de um satélite, realizada pelo controle da missão, é obtida através do controle de atitude; o controle de atitude também pode reorientar o satélite para atender novas missões, a critério do centro de controle do sistema.

2.6 Modelagem dos sistemas

Neste capítulo apresentamos uma breve revisão das modelagens dos sistemas que serão utilizados neste trabalho, sobre os quais aplicamos as técnicas de controle LQR, SDRE, e H_{∞} , auxiliadas por controle *Fuzzy* e algoritmos genéticos. Serão revistos os modelos de dois sistemas: o primeiro sistema cujo modelo será revisto é simulador de satélite construído pelo Instituto Nacional de Pesquisas Espaciais (INPE), mostrado no lado esquerdo da Figura 2.4, abaixo.



Figura 2.4 - Simulador de satélite do INPE

O segundo sistema do qual é revisto apenas o modelo linearizado é o dispositivo que consiste de uma lâmina flexível de alumínio acoplada ao servo motor SRV2, fornecido pela Quanser®, mostrado na Figura 2.5, que é um sistema rígido-flexível.

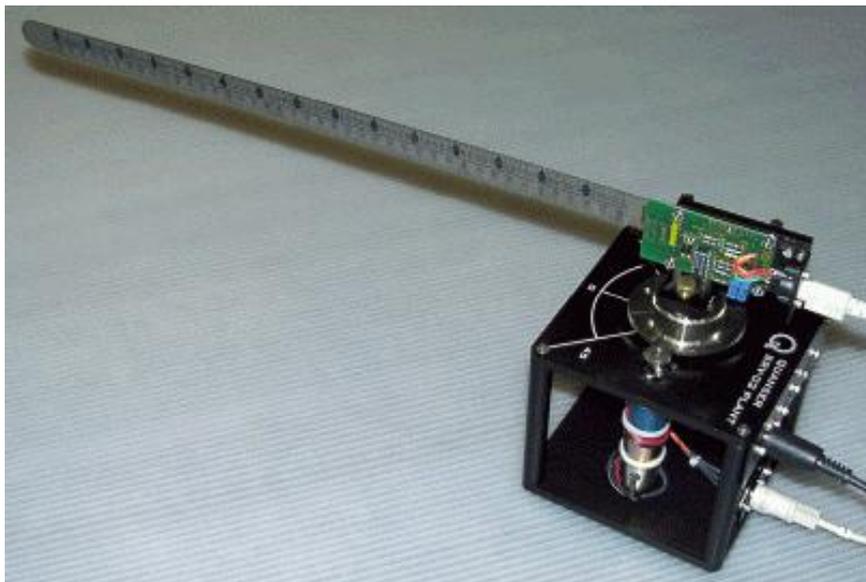


Figura 2.5 -Positivo SRV2 e haste flexível

Fonte: Extraído do manual da Quanser®

Para fins de simplificação, neste trabalho, o dispositivo SRV2 da Figura 2.5 acima foi representado pelo sistema mostrado na Figura 2.6, abaixo:

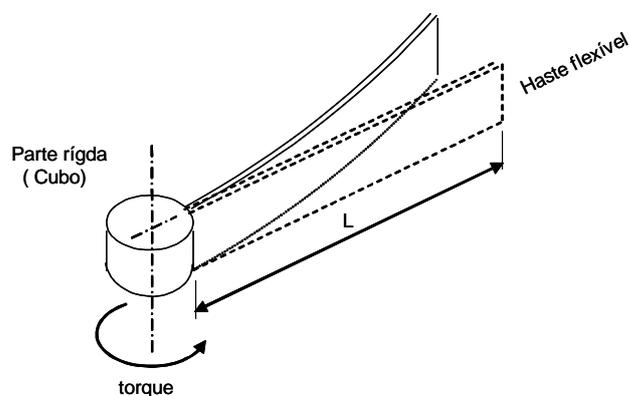


Figura 2.6 – Sistema rígido-flexível

O terceiro sistema destinado a atender o último objetivo desta tese é outro satélite rígido-flexível, o satélite sino-brasileiro denominado por CBERS-1 mostrado pela Figura 2.7, cujo modelamento não será revisto neste trabalho; serão utilizadas as matrizes deste artefato desenvolvidas por Cárdenas (CARDENAS, 2005).



Figura 2.7 – Satélite sino-brasileiro CBERS-1

2.7 Modelo do simulador de satélite

O modelo deste simulador já é conhecido na literatura de controle e pode ser desenvolvido por dois métodos: o método lagrangiano e o método usado na mecânica clássica (MEIROVITCH, 1970). Por ser mais simples e fornecer, durante todo o seu desenvolvimento, maior sensibilidade ao projetista, optou-se pelo modelo gerado pelo método clássico. Neste trabalho foi usado o desenvolvimento utilizado por Doruk (2008), Fonseca (2011), Gonzales (2009), Vilela (2009), Souza e Arena (2013) e Souza (2010), entre outros.

O simulador de satélite, mostrado pela Figura 2.4, cujo modelo é revisto nesta seção, consiste de uma plataforma metálica com diâmetro aproximado de 1200 milímetros, com um mancal em forma de rótula fixo na parte inferior da plataforma com seu eixo vertical coincidente com o eixo principal da plataforma. A rótula é encaixada em um alojamento esférico côncavo. Quando em funcionamento, entre a rótula e seu alojamento é injetado ar comprimido de modo criar um filme de ar entre as superfícies, propiciando um atrito muito

pequeno, aproximando do atrito encontrado pelo satélite quando no espaço. Este simulador, além dos componentes necessários ao desempenho de suas missões, está equipado com três rodas de reação conforme mostrado na Figura 2.8 que é um esboço mostrando apenas a plataforma e as rodas de reação do simulador apresentado na Figura 2.4.

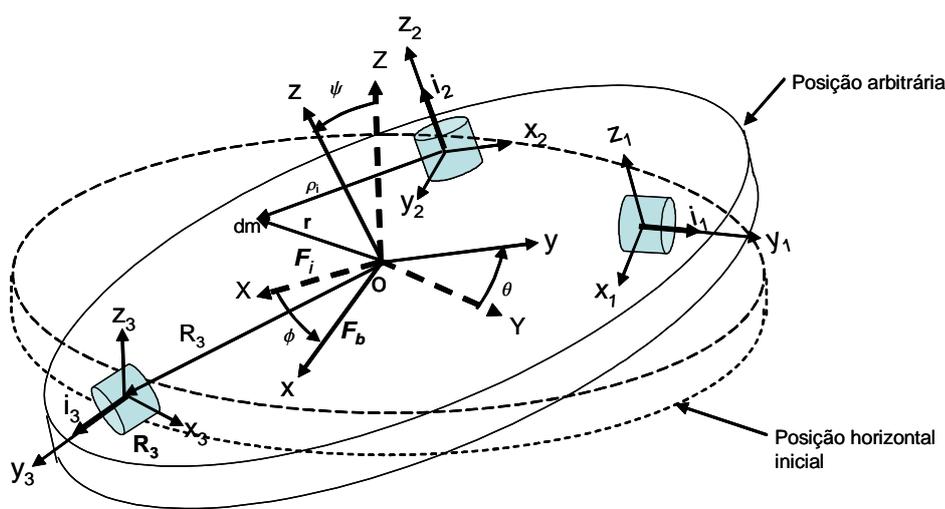


Figura 2.8 - Plataforma do simulador de satélite

O simulador pode girar em torno dos três eixos sendo que os giros em torno dos eixos X e Y estão limitados, por razões físicas estruturais a valores menores que ± 90 graus. Conforme mostra a Figura 2.8, o sistema de coordenadas (x, y, z) tem origem no centro geométrico da mesa e é solidário com ela, é o sistema fixo no corpo, F_b ; o sistema de coordenada (X, Y, Z) é o sistema inercial, F_i , fixo, para fins de simplificação na base da plataforma, as origens de F_b e F_i são coincidentes. Cada roda de reação tem seu próprio sistema de coordenada dado por (x_i, y_i, z_i) , onde $i = 1, 2$ e 3 designam as rodas 1, 2 e 3, respectivamente. A velocidade angular da plataforma em relação ao sistema F_i é dada por $\bar{\Omega}$ e as velocidades angulares das rodas de reação em relação aos seus próprios sistemas de coordenadas são dadas por

$\vec{\omega}_1$, $\vec{\omega}_2$ e $\vec{\omega}_3$. Os vetores $\mathbf{R}_{1,2,3}$ indicam a localização dos centros de massa das rodas de reação em relação às coordenadas \mathbf{F}_i e os vetores $\vec{\rho}_i$, em que i varia de 1 a m , indicam a posição dos elementos de massa dm da plataforma em relação ao sistema de coordenada de cada roda. Na figura 2.8 é apresentada também a atitude do simulador representada pelos ângulos de Euler ϕ , θ e ψ que mostram as posições dos eixos coordenados do sistema \mathbf{F}_b em relação aos eixos coordenados de \mathbf{F}_i .

Com o objetivo de facilitar a derivação do modelo do simulador, a partir da Figura 2.8, foi criada a Figura 2.9 na qual apenas os parâmetros necessários ao modelamento foram conservados.

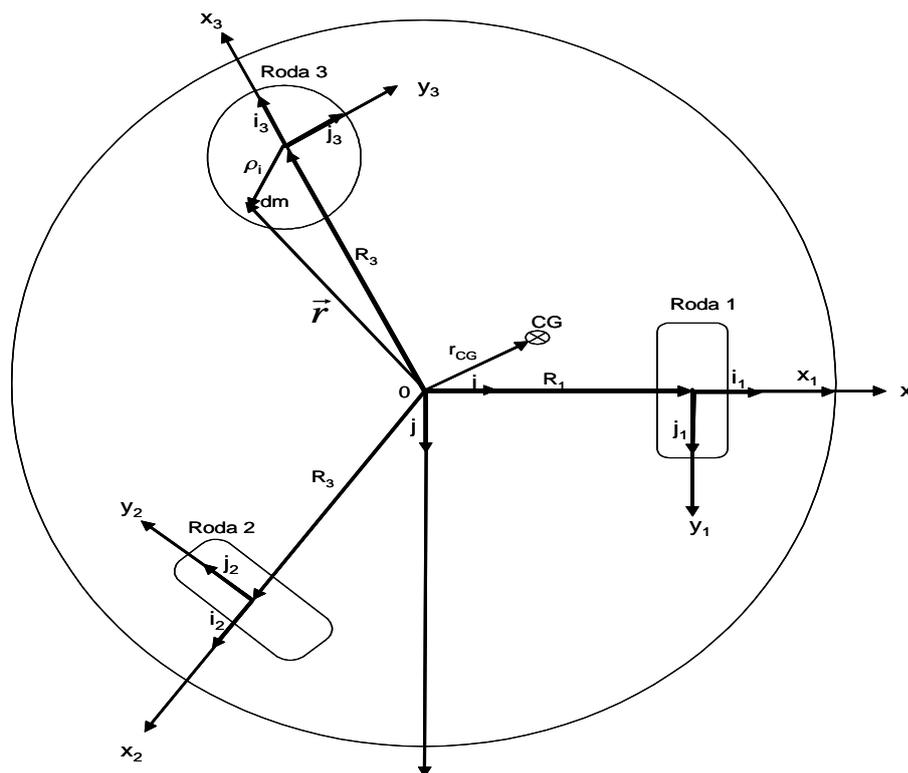


Figura 2.9 - Plataforma simplificada - Simplificação da Figura (2.1)

Como o simulador é animado apenas de movimento de rotação, para obter-se seu modelo basta trabalhar apenas com seu momento angular, lembrando-se

que o simulador é um sistema composto pela plataforma (denominada também de base ou simplesmente B no sinal de integração de (2.1), (2.2) e (2.3)) e de três rodas de reação. O momento angular deste sistema é dado pela expressão a seguir.

$$\vec{H} = \vec{h} + \sum_{i=1}^3 \vec{h}_i \quad \text{com } i=1,2,3 \quad (2.1)$$

$$\vec{H} = \int_{\text{base}} (\vec{r} \times \vec{v}_B) dm + \sum_{i=1}^3 \int_{\text{roda}} (\vec{\rho}_i \times \vec{v}_R) dm \quad (2.2)$$

em que \vec{v}_B , velocidade absoluta dos elementos de massa da base, dm , é dada por $\vec{v}_B = \vec{\Omega} \times \vec{r}$ e \vec{v}_R , velocidade absoluta dos elementos de massa das rodas, é dada por $\vec{v}_R = \vec{\Omega} \times \vec{R}_i + \vec{\omega}_i \times \vec{\rho}_i$; substituindo estes valores de \vec{v}_B e \vec{v}_R equação (2.1), tem-se

$$\vec{H} = \int_{\text{Base}} \vec{r} \times (\vec{\Omega} \times \vec{r}) dm + \sum_{i=1}^3 \int_{\text{Roda}} \vec{\rho}_i \times (\vec{\Omega} \times \vec{R}_i + \vec{\omega}_i \times \vec{\rho}_i) dm \quad (2.3)$$

onde \vec{R}_i , com $i=1,2,3$ é a distância do centro da plataforma às rodas de reação, $\vec{\Omega}$ é a velocidade de rotação da base ou plataforma e $\vec{\omega}_i$, com $i=1,2,3$, é a velocidade de rotação de cada roda de reação. Desenvolvendo-se a segunda parcela do segundo membro de (2.2), obtém-se a expressão a seguir:

$$\vec{H} = \int_{\text{Base}} \vec{r} \times (\vec{\Omega} \times \vec{r}) dm + \sum_{i=1}^3 \int_{\text{Roda}} \vec{\rho}_i \times (\vec{\Omega} \times \vec{R}_i) dm + \sum_{i=1}^3 \int_{\text{Roda}} \vec{\rho}_i \times (\omega_i \times \vec{\rho}_i) dm \quad (2.4)$$

onde a primeira parcela do segundo membro de (2.4) é o momento angular da plataforma ou base, a segunda parcela é o momento angular de cada roda de reação em função da rotação apenas da plataforma e a terceira parcela é o momento angular de cada roda em função da rotação da própria roda de reação.

Os raios vetores \vec{r} e $\vec{\rho}_i$ e as rotações $\vec{\Omega}$ e $\vec{\omega}_i$ são dadas pelas expressões a seguir:

$$\vec{r} = x\vec{\hat{i}} + y\vec{\hat{j}} + z\vec{\hat{k}} \quad (2.5)$$

$$\vec{\rho}_i = x_i\vec{\hat{i}} + y_i\vec{\hat{j}} + z_i\vec{\hat{k}} \quad \text{onde } i=1,2,3 \quad (2.6)$$

$$\vec{\Omega} = p\vec{\hat{i}} + q\vec{\hat{j}} + r\vec{\hat{k}} \quad (2.7)$$

$$\vec{\omega}_i = \omega_x\vec{\hat{i}}_i + \omega_y\vec{\hat{j}}_i + \omega_z\vec{\hat{k}}_i \quad (2.8)$$

Chama-se a atenção para o fato de, neste trabalho, usar-se o vetor \vec{r} como raio vetor da plataforma, ver Figura 2.9, e o escalar r como coeficiente de $r\vec{\hat{k}}$ na expressão (2.7), são entidades distintas. As entidades $\vec{\hat{i}}, \vec{\hat{j}}, \vec{\hat{k}}$ e $\vec{\hat{i}}_i, \vec{\hat{j}}_i, \vec{\hat{k}}_i$ são vetores unitários componentes dos triedros básicos que neste trabalho, para simplificar as notações, serão escritos sem símbolo de vetor unitário ($\hat{\cdot}$).

Substituindo as expressões de (2.5) a (2.8) em (2.3) e desenvolvendo-se os produtos vetoriais e as necessárias simplificações, obtém-se:

$$\vec{r} \times (\vec{\Omega} \times \vec{r}) = [(y^2 + z^2)p - xyq - xzr]\vec{\hat{i}} + [-xyp + (x^2 + y^2)q - yzr]\vec{\hat{j}} + [-xzp - yzq + (y^2 + x^2)r]\vec{\hat{k}} \quad (2.9)$$

$$\vec{\rho}_i \times (\vec{\omega}_i \times \vec{\rho}_i) = [(y^2 + x^2)\omega_x - xy\omega_y - xz\omega_z]\vec{\hat{i}} + [-yx\omega_x + (y^2 + z^2)\omega_y - yz\omega_z]\vec{\hat{j}} + [-zx\omega_x - zy\omega_y + (x^2 + y^2)\omega_z]\vec{\hat{k}} \quad (2.10)$$

As integrais destas parcelas definem os momentos e produtos de inércia dados a seguir:

$$\begin{aligned}
I_{xx} &= \int (y^2 + z^2) dm & I_{xy} &= I_{yx} = -\int xy dm \\
I_{yy} &= \int (x^2 + z^2) dm & I_{xz} + I_{zx} &= -\int xz dm \\
I_{zz} &= \int (x^2 + y^2) dm & I_{yz} + I_{zy} &= -\int yz dm
\end{aligned} \tag{2.11}$$

Quando a origem do sistema de coordenadas coincide com o centro de massa do sistema, os produtos cruzados de inércia se anulam; isto ocorre com as rodas de reação, resultando em simplificações. Com a plataforma esta simplificação não ocorre. Assim tem-se:

$$\vec{h} = (I_{xx}p + I_{xy}q + I_{xz}r)\vec{i} + (I_{yx}p + I_{yy}q + I_{yz}r)\vec{j} + (I_{zx}p + I_{zy}q + I_{zz}r)\vec{k} \tag{2.12}$$

Para as rodas de reação tem-se:

$$\begin{aligned}
\vec{h}_1 &= I_{xx}\omega_x\vec{i}_1 = I_1\omega_1\vec{i}_1 \\
\vec{h}_2 &= I_{yy}\omega_y\vec{i}_2 = I_2\omega_2\vec{i}_2 \\
\vec{h}_3 &= I_{zz}\omega_z\vec{i}_3 = I_3\omega_3\vec{i}_3
\end{aligned} \tag{2.13}$$

Caso não existam torques externos agindo sobre o sistema, o momento angular do sistema é conservado. Se existirem torques externos atuando sobre referido sistema, o teorema de Euler mostra que estes torques equivalem à derivada temporal do momento angular do sistema, dados pela expressão a seguir:

$$\dot{\vec{H}} = \dot{\vec{h}} + \sum_{i=1}^3 \dot{\vec{h}}_i \tag{2.14}$$

Em que $\dot{\vec{h}}$ é dado pelo desenvolvimento a seguir:

$$\begin{aligned} \dot{\vec{h}} = & (I_{xx}\dot{p} + I_{xy}\dot{q} + I_{xz}\dot{r})\vec{i} + (I_{yx}\dot{p} + I_{yy}\dot{q} + I_{yz}\dot{r})\vec{j} + (I_{zx}\dot{p} + I_{zy}\dot{q} + I_{zz}\dot{r})\vec{k} + \\ & + (I_{xx}p + I_{xy}q + I_{xz}r)(\vec{\Omega} \times \vec{i}) + (I_{yx}p + I_{yy}q + I_{yz}r)(\vec{\Omega} \times \vec{j}) + (I_{zx}p + I_{zy}q + I_{zz}r)(\vec{\Omega} \times \vec{k}) \end{aligned} \quad (2.15)$$

$$\begin{aligned} \dot{\vec{h}} = & (I_{xx}\dot{p} + I_{xy}\dot{q} + I_{xz}\dot{r})\vec{i} + (I_{yx}\dot{p} + I_{yy}\dot{q} + I_{yz}\dot{r})\vec{j} + (I_{zx}\dot{p} + I_{zy}\dot{q} + I_{zz}\dot{r})\vec{k} + \\ & + \vec{\Omega} \times [(I_{xx}p + I_{xy}q + I_{xz}r)\vec{i} + (I_{yx}p + I_{yy}q + I_{yz}r)\vec{j} + (I_{zx}p + I_{zy}q + I_{zz}r)\vec{k}] \end{aligned} \quad (2.16)$$

$$\dot{\vec{h}} = (\dot{\vec{h}})_R + \vec{\Omega} \times \vec{h} \quad (2.17)$$

$\sum_{i=1}^3 \dot{\vec{h}}_i$ é dado pelo desenvolvimento a seguir:

$$\sum_{i=1}^3 \dot{\vec{h}}_i = I_1\dot{\omega}_1\vec{i}_1 + I_1\omega_1\dot{\vec{i}}_1 + I_2\dot{\omega}_2\vec{i}_2 + I_2\omega_2\dot{\vec{i}}_2 + I_3\dot{\omega}_3\vec{i}_3 + I_3\omega_3\dot{\vec{i}}_3 \quad (2.18)$$

$$\sum_{i=1}^3 \dot{\vec{h}}_i = I_1\dot{\omega}_1\vec{i}_1 + I_1\omega_1(\vec{\Omega} \times \vec{i}_1) + I_2\dot{\omega}_2\vec{i}_2 + I_2\omega_2(\vec{\Omega} \times \vec{i}_2) + I_3\dot{\omega}_3\vec{i}_3 + I_3\omega_3(\vec{\Omega} \times \vec{i}_3) \quad (2.19)$$

$$\sum_{i=1}^3 \dot{\vec{h}}_i = I_1\dot{\omega}_1\vec{i}_1 + I_2\dot{\omega}_2\vec{i}_2 + I_3\dot{\omega}_3\vec{i}_3 + \vec{\Omega} \times (I_1\omega_1\vec{i}_1 + I_2\omega_2\vec{i}_2 + I_3\omega_3\vec{i}_3) \quad (2.20)$$

$$\sum_{i=1}^3 \dot{\vec{h}}_i = \sum_{i=1}^3 (\dot{\vec{h}}_i)_R + \vec{\Omega} \times \left(\sum_{i=1}^3 \vec{h}_i \right) \quad (2.21)$$

A equação do movimento da plataforma é obtida pela aplicação do teorema de Euler expresso pela equação:

$$\vec{T} = \dot{\vec{H}} \quad (2.22)$$

ou seja:

$$m(r_{cg} \times \vec{g}) = (\dot{\vec{h}})_R + \vec{\Omega} \times \vec{h} + \sum_{i=1}^3 (\dot{\vec{h}}_i)_R + \vec{\Omega} \times \left(\sum_{i=1}^3 \vec{h}_i \right) \quad (2.23)$$

Os termos de (2.23) devem estar na mesma base ortogonal, escolheu-se a base do sistema fixo no corpo $[\vec{i}, \vec{j}, \vec{k}]$, portanto, usam-se as substituições feitas a seguir:

$\vec{i}_1 = \vec{i}$, $\vec{i}_2 = \vec{j}$ e $\vec{i}_3 = \vec{k}$ e expressa-se a aceleração da gravidade \vec{g} em função do sistema fixo no corpo, conforme segue:

$$\vec{g} = |\vec{g}|[-\sin(\theta)\vec{i} + \sin(\phi)\cos(\theta)\vec{j} + \cos(\theta)\cos(\phi)\vec{k}] \quad (2.24)$$

onde ϕ , θ e ψ são os ângulos de Euler que mostram a posição ou atitude do simulador em relação ao sistema inercial F_i , ilustrada pela Figura 2.10; nesta Figura esta representada o centro de massa do simulador.

Para as rodas de reação, a aplicação do teorema de Euler fornece os momentos angulares dados pelas expressões a seguir:

$$\begin{aligned} Q_1 &= I_1(\vec{\omega}_1 + \vec{\Omega}) \cdot \vec{i}_1 = I_1(\omega_1 + p) \\ Q_2 &= I_2(\vec{\omega}_2 + \vec{\Omega}) \cdot \vec{i}_2 = I_2(\omega_2 + q) \\ Q_3 &= I_3(\vec{\omega}_3 + \vec{\Omega}) \cdot \vec{i}_3 = I_3(\omega_3 + r) \end{aligned} \quad (2.25)$$

As derivadas temporais das expressões (2.25) fornecem as equações dos torques das rodas de reação, conforme segue:

$$\begin{aligned} \dot{Q}_1 &= I_1[\dot{\omega}_1 + \dot{p}] = T_1 \\ \dot{Q}_2 &= I_2[\dot{\omega}_2 + \dot{q}] = T_2 \\ \dot{Q}_3 &= I_3[\dot{\omega}_3 + \dot{r}] = T_3 \end{aligned} \quad (2.26)$$

\vec{T}_1 , \vec{T}_2 e \vec{T}_3 , dados por (2.26), são torque internos ao simulador (plataforma e rodas de reação), não alterando o momento angular total \vec{H} .

Para descrever a atitude da plataforma em relação ao sistema inercial (X, Y, Z) foi escolhida a sequência 3-2-1 das rotações dos ângulos de Euler na qual a matriz de rotação, usando os cossenos diretores, transforma os vetores do sistema inercial F_i para o sistema fixo na plataforma F_b ; essa matriz é dada a seguir:

$$R = \begin{bmatrix} c(\theta)c(\psi) & c(\theta)s(\psi) & -s(\theta) \\ s(\phi)s(\theta)c(\psi) - c(\phi)s(\psi) & s(\phi)s(\theta)s(\psi) - c(\phi)c(\psi) & s(\phi)c(\theta) \\ c(\phi)s(\theta)c(\psi) - s(\phi)s(\psi) & c(\phi)s(\theta)s(\psi) - s(\phi)c(\psi) & c(\phi)c(\theta) \end{bmatrix} \quad (2.27)$$

Em que, s, c são respectivamente abreviações de seno e cosseno e ϕ , θ e ψ são os ângulos de atitude da plataforma conforme mostrados na Figura 2.8, e na Figura 2.10, na qual está representado o centro de massa CM

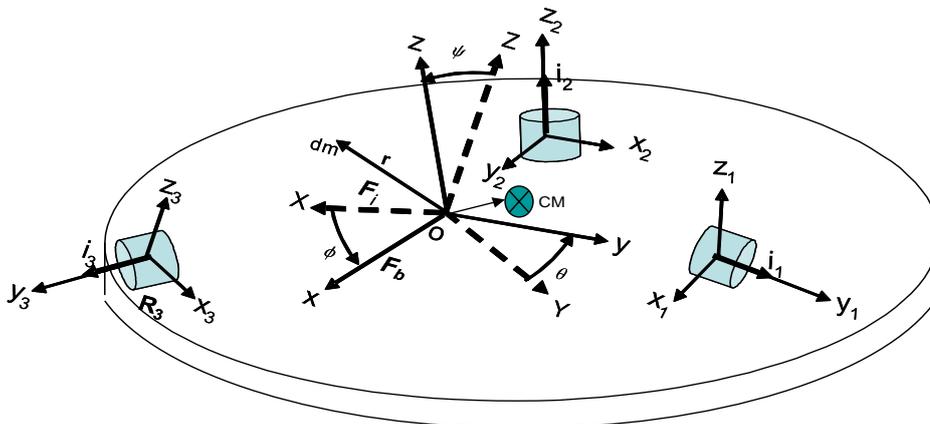


Figura - 2.10 – Reapresentação da Figura 2.5, mostrando F_i e F_b e o CM

As equações cinemáticas para sequência 3-2-1 que descrevem a atitude da plataforma são dadas por:

$$\begin{aligned}\dot{\phi} &= p + \tan(\theta)[q\text{sen}(\phi) + r\cos(\phi)] \\ \dot{\theta} &= q\cos(\phi) - r\text{sen}(\phi)\end{aligned}\tag{2.28}$$

$$\dot{\psi} = \frac{1}{\cos(\theta)} + [q\text{sen}(\phi) + r\cos(\phi)]$$

Organizando todas as equações desenvolvidas anteriormente, com os termos de derivadas de primeira ordem no primeiro membro da igualdade e os termos de derivada de ordem zero no segundo membro desta igualdade, obtém-se a equação matricial dada a seguir:

$$\dot{X} = M^{-1}F(X)\tag{2.29}$$

onde \dot{X} , M e $F(X)$ são matrizes obtidas, dadas por (2.30) e (2.31).

Este sistema pode ser integrado pelo método Runge Kuta, desde que seja dado as condições iniciais $X_0 = [p_o \ q_o \ r_o \ \phi_o \ \theta_o \ \psi_o \ \omega_{10} \ \omega_{20} \ \omega_{30}]^T$

$$\dot{X} = \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{bmatrix} \quad M = \begin{bmatrix} I_{xx} & I_{yx} & I_{zx} & 0 & 0 & 0 & I_1 & 0 & 0 \\ I_{xy} & I_{yy} & I_{zy} & 0 & 0 & 0 & 0 & I_2 & 0 \\ I_{xz} & I_{yz} & I_{zz} & 0 & 0 & 0 & 0 & 0 & I_3 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}\tag{2.30}$$

$$F(X) = \begin{bmatrix}
(I_{xx} - I_{zz})qr + I_{xy}pr - I_{xz}pq + I_{yz}(r^2 - q^2) + I_2\omega_2r - I_3\omega_3q + mgr_y \cos(\theta) \sin(\phi) - \\
\qquad \qquad \qquad - mgr_z \cos(\theta) \sin(\phi) \\
(I_{zz} - I_{xx})pr + I_{yz}pq - I_{xy}qr + I_{xz}(p^2 - r^2) + I_1\omega_1r - I_3\omega_3p + mgr_x \cos(\theta) \cos(\phi) - \\
\qquad \qquad \qquad - mgr_z \sin(\theta) \\
(I_{xx} - I_{yy})pq + I_{xz}qr - I_{yz}pr + I_{xy}(q^2 - p^2) + I_1\omega_1q - I_2\omega_2p + mgr_x \cos(\theta) \sin(\theta) - \\
\qquad \qquad \qquad - mgr_y \sin(\theta) \\
p + tg(\theta)[q \sin(\phi) + r \cos(\phi)] \\
q \cos(\phi) + \sin(\phi) \\
\frac{1}{\cos(\theta)} [q \sin(\phi) + r \cos(\phi)] \\
\frac{T_1}{I_1} \\
\frac{T_2}{I_2} \\
\frac{T_3}{I_3}
\end{bmatrix} \quad (2.31)$$

2.8 Modelo Linearizado do Simulador

O modelo do simulador, obtido na seção anterior, é naturalmente um modelo não linear; para que se possa aplicar um controlador linear sobre o simulador é necessário linearizar o seu modelo não linear original. Esta linearização é obtida assumindo-se pequenas perturbações, ou seja, adotando-se pequenos ângulos ϕ , θ e ψ (valores próximos de zero, dependendo do grau de precisão da missão) para desvios de atitude do simulador; esta hipótese fornece produtos desprezíveis de duas velocidades quando comparados com outros termos, além de que os senos de pequenos ângulos podem ser aproximados para zero e seus cossenos podem ser aproximados para a unidade. Outro fator de linearizante adotado foi fazer $r_{cg} = 0$, que equivale a considerar o centro de massa da plataforma coincidente com a origem dos sistemas de coordenadas; isto anula todos os produtos cruzados de inércia. Da adoção destas simplificações, obtém-se o desenvolvimento abaixo:

$$I_{xx}\dot{p} + I_1\dot{\omega}_1 = 0$$

$$I_{yy}\dot{q} + I_2\dot{\omega}_2 = 0 \quad (2.32)$$

$$I_{zz}\dot{r} + I_3\dot{\omega}_3 = 0$$

$$\dot{\phi} = p, \quad \dot{\theta} = q \quad \text{e} \quad \dot{\psi} = r \quad (2.33)$$

$$\dot{\omega}_1 = \frac{T_1}{I_1} - \dot{p}$$

$$\dot{\omega}_2 = \frac{T_2}{I_2} - \dot{q} \quad (2.34)$$

$$\dot{\omega}_3 = \frac{T_3}{I_3} - \dot{r}$$

Levando as equações (2.34) para (2.32) e considerando-se igualdades (2.33), obtém-se o conjunto de equações dado por (2.35), a seguir:

$$(I_{xx} - I_1)\dot{p} = -T_1$$

$$(I_{yy} - I_2)\dot{q} = -T_2$$

$$(I_{zz} - I_3)\dot{r} = -T_3$$

(2.35)

$$\dot{\phi} = p$$

$$\dot{\theta} = q$$

$$\dot{\psi} = r$$

Organizando (2.35) na forma matricial $\dot{x} = Ax + Bu$, tem-se o sistema linearizado (2.36) onde $x = [\phi \ \theta \ \psi \ p \ q \ r]^T$ é o vetor de estado do sistema linearizado:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \\ \phi \\ \theta \\ \psi \end{bmatrix} + \begin{bmatrix} \frac{1}{I_1 - I_{xx}} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{I_2 - I_{yy}} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{I_3 - I_{zz}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad (2.36)$$

2.9 Satélite rígido-flexível

Nesta seção é desenvolvida a modelagem do sistema, aqui denominado de sistema rígido-flexível, que consiste de uma haste de alumínio de seção retangular, engastada em um bloco cilíndrico (cubo), também metálico, que pode girar em torno de seu eixo vertical coincidente com o eixo \vec{Z} do sistema de coordenadas inerciais F_i ; este sistema é representado pelo modelo *SRV02*, fornecido pela Quanser.[®] mostrado na Figura 2.5. A Figura 2.11 apresenta o esboço da estrutura deste sistema e os sistemas de referência usados: \vec{F}_i e \vec{F}_b , sistema inercial e fixo no corpo respectivamente. São apresentados na Figura 2.12 os símbolos ρ , I , E e L que representam a densidade linear de massa da haste flexível, a inércia da haste em relação a sua linha neutra, o módulo de Young e comprimento da haste, respectivamente. Os eixos \vec{Z} e \vec{z} são coincidentes e perpendiculares ao plano do desenho. O conjugado $\vec{\tau}$ tem mesma direção e sentido dos eixos \vec{Z} e \vec{z} .

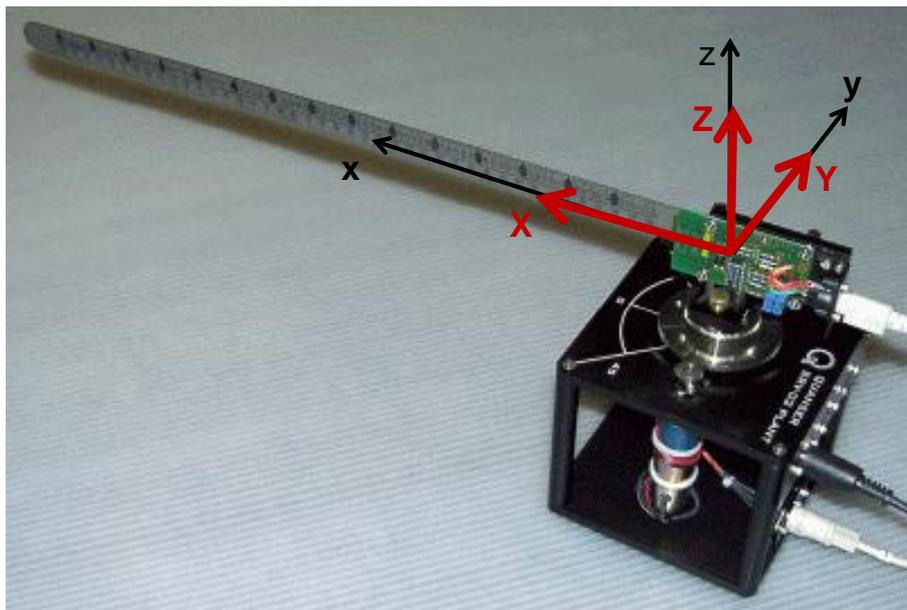


Figura 2.11 – Localização dos sistemas de coordenadas no dispositivo SRV2

Adaptado do Manual da Quanser EE-479

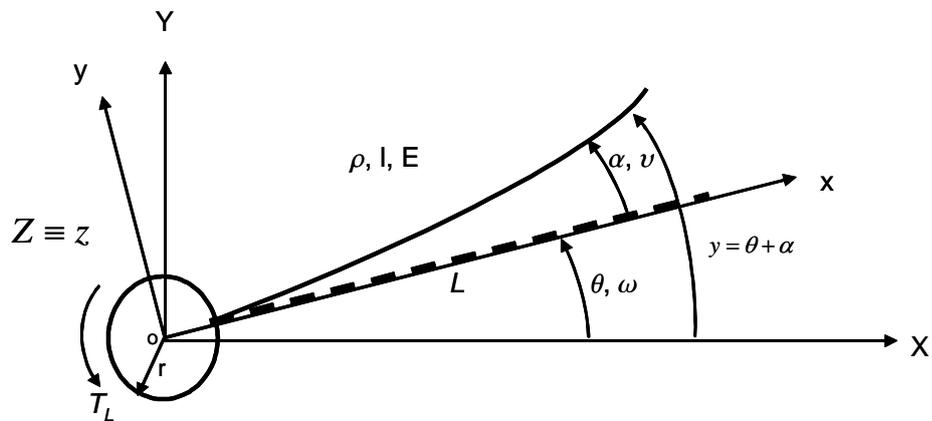


Figura 2.12-Sistema rígido-flexível

Fonte: Extraído Quanser® EE-479

2.10 Modelagem do sistema rígido-flexível

A seguir são apresentados todos os parâmetros da haste e alguns parâmetros do SRV2 (servo-motor), necessários para a modelagem. Do servo-motor só precisaremos do torque e do momento de inércia do sistema formado pelo rotor

do motor e engrenagens, já calculados pelo fabricante. Os parâmetros da haste fornecidos pelo fabricante são dados a seguir:

$\theta(t)$	Deslocamento angular do servo-motor;	
$\omega(t)$	Velocidade angular do servo-motor;	
$\alpha(t)$	Deflexão elástica da haste;	
$v(t)$	Velocidade da deflexão elástica da haste;	
$\gamma(t)$	Deflexão elástica total da haste;	
L	Comprimento da haste (L= 0,381m);	(2.37)
D	Comprimento do arco de deflexão (D= αL);	
m	Massa da haste (m=65 g);	
J_h	Momento de inércia da haste ($J_h = \frac{1}{3} mL^3 = 0.003145 Nm$);	
ω_{FL}	Frequência natural amortecida da haste ($\omega_{FL} = 6\pi$ rad);	
K_{stiff}	Coeficiente de rigidez da haste ($k_{stiff} = \omega_{FL}^2 J_{FL} = 1.1175$);	
K_{Gage}	Fator de calibração do strain gage (1 volt/polegada) e	
B_{eq}	Coeficiente de atrito viscoso.	
ρ	Densidade linear de massa;	
I	Momento de inércia e	
E	Módulo de Young	

Nas derivações desta secção, para fins de simplificação da simbologia, serão omitidos os símbolos de domínio temporal das variáveis dependentes do tempo, (t). Assim, considerando a Figura 2.11 e as definições (2.37) dadas acima, tem-se:

$$\begin{aligned}\dot{\theta} &= \omega \\ \dot{\alpha} &= \nu\end{aligned}\tag{2.38}$$

Considerando que J_h é o momento de inércia da haste flexível, o torque devido á sua aceleração é dado por Ogata (1978):

$$T_{J_h} = J_h \ddot{\gamma} = J_h (\ddot{\theta} + \ddot{\alpha}) = J_h (\dot{\omega} + \dot{\nu})\tag{2.39}$$

Nas derivações que se seguem, o fabricante (QUANSER, 2009) usou o valor numérico do coeficiente de rigidez de torsão, obtido em laboratório, como uma aproximação para o coeficiente de rigidez de flexão, K_{Stiff} , e assumiu que esta rigidez é proporcional à deflexão angular total γ , ou seja:

$$\begin{aligned}T_{K_{Stiff}} &= K_{Stiff} \alpha \\ T_{J_h} + T_{K_{Stiff}} &= 0\end{aligned}\tag{2.40}$$

ou

$$J_h (\dot{\omega} + \dot{\nu}) + K_{Stiff} \alpha = 0\tag{2.41}$$

O torque de saída do servo-motor, além de vencer inércia devido a J_{eq} , (J_{eq} é o momento de inércia equivalente do conjunto servo-motor e da haste flexível) e o torque causado pelo atrito B_{eq} é assumido vencer também o torque devido a aceleração da haste, isto é:

$$J_{eq} \dot{\omega} + B_{eq} \omega + J_h (\dot{\omega} + \dot{\nu}) = T_L\tag{2.42}$$

onde T_L é o torque da carga.

Substituindo $J_h (\dot{\omega} + \dot{\nu})$ da expressão (2.41) na expressão (2.42), tem-se:

$$\dot{\omega} = \frac{K_{Stiff}}{J_{eq}} \alpha + \frac{1}{J_{eq}} T_L - \frac{B_{eq}}{J_{eq}} \omega \quad (2.43)$$

Substituindo T_L da expressão (2.43) pela expressão de T_L fornecido pelo modelamento do servo-motor $\left(T_L(t) = \frac{\eta K_m K_g}{R_a} v_i(t) - \frac{K_m^2 K_g^2}{R_a} \dot{\theta}(t) \right)$, não derivado neste trabalho e sim fornecido manual EE-479 da Quanser[®] (Rotary Servo Base Unit SVR2), tem-se

$$\dot{\omega} = \frac{K_{Stiff}}{J_{eq}} \alpha - \frac{\eta K_m^2 K_g^2 + B_{eq} R_a}{J_{eq} R_a} \omega + \frac{\eta K_m K_g}{J_{eq} R_a} v_i \quad (2.44)$$

Substituindo o valor de $\dot{\omega}$ fornecido por (2.44) na expressão (2.42), obtém-se:

$$\dot{\nu} = \frac{K_{Stiff} (J_{eq} + J_h)}{J_{eq} J_h} \alpha + \frac{\eta K_m^2 K_g^2 + B_{eq} R_a}{J_{eq} R_a} \omega - \frac{\eta K_m K_g}{J_{eq} R_a} v_i \quad (2.45)$$

Para expressar o desenvolvimento acima na forma de espaço de estado dado pelas expressões (2.46), a seguir:

$$\begin{aligned} x(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \quad (2.46)$$

onde $x(t) = [\theta \ \alpha \ \omega \ \nu]^T$ e θ , α , ω , ν são respectivamente deslocamento angular do cubo (parte rígida do sistema rígido-flexível), deslocamento elástico da extremidade da haste, velocidade angular do cubo e velocidade de

deslocamento elástico da haste; colocamos as expressões de θ , α , ω e ν dadas pelas expressões (2.38), (2.44) e (2.45) respectivamente, organizadas de forma a obter todas as matrizes de (2.46), conforme segue:

$$\begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \\ \dot{\omega} \\ \dot{\nu} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{K_{Stiff}}{J_{eq}} & -\frac{\eta K_m^2 K_g^2 + B_{eq} R_a}{J_{eq} R_a} & 0 \\ 0 & -\frac{K_{Stiff} (J_{eq} + J_h)}{J_{eq} J_h} & \frac{\eta K_m^2 K_g^2 + B_{eq} R_a}{J_{eq} R_a} & 0 \end{bmatrix}}_A \begin{bmatrix} \theta \\ \alpha \\ \omega \\ \nu \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{\eta K_m K_g}{J_{eq} R_a} & 0 \\ \frac{\eta K_m K_g}{J_{eq} R_a} & 0 \end{bmatrix}}_B \nu_i \quad (2.47)$$

Substituindo na equação 2.47 acima os valores dos parâmetros fornecidos pela Tabela (1) dada no Apêndice (B), obtém-se o sistema dado em (2.48), abaixo:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 591.9697 & -31.9744 & 0 \\ 0 & -947.2755 & 31.9744 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 56.2361 & 0 \\ -56.2361 & 0 \end{bmatrix} \quad (2.48)$$

em que as matrizes $\mathbf{A} \in \mathbb{R}^{4 \times 4}$ e $\mathbf{B} \in \mathbb{R}^{4 \times 2}$ são respectivamente as matrizes de estado e de controle do sistema rígido-flexível.

3 REVISÃO DAS TÉCNICAS DE CONTROLE USADAS NESTE TRABALHO

Neste Capítulo são revistos os conceitos básicos, definições e formalismo de Lógica *Fuzzy*, Algoritmos Genéticos, da técnica de controle multi-variável Hinfinito e Controle SDRE, que serão utilizados neste trabalho.

3.1 Lógica *Fuzzy* e elementos de controle *Fuzzy*

Nas teorias de controle clássicas e moderna, o primeiro passo para a implementação do controle de um sistema é a obtenção do modelo matemático do sistema; isto requer que se conheça detalhadamente o sistema a ser controlado, o que nem sempre é possível, principalmente se o sistema for muito complexo. Em diversos casos um volume considerável de informações sobre o sistema são desconhecidos e algumas características deste sistema só são conhecidos a priori, de forma empírica, qualitativa e expressas de forma literal (não determinística). Este panorama leva à imprecisões e falta de exatidão que inviabilizam a aplicação das técnicas de controles estocásticos ou determinísticos até agora conhecidos, e abrem espaço para a aplicação de um controle não determinístico baseado em lógica nebulosa e controle nebuloso, mais conhecido na literatura de controle de sistemas por lógica *Fuzzy* e controle *Fuzzy*. Durante todo este trabalho estaremos usando as denominações mais usuais que são lógica *Fuzzy* e controle *Fuzzy*. Usa-se o vocábulo inglês *Fuzzy* e seus derivados *Fuzzyficação* e *deFuzzyficação* adaptados para língua portuguesa por não existir neste idioma vocábulos correspondentes aos vocábulos ingleses *Fuzzyfication* e *deFuzzyfication* (SHAW e SIMÕES, 1999), (LINDEN, 2012). A seguir é dado os significados deste vocábulos:

- ***Fuzzyficação***: É a operação realizada pelo controle *Fuzzy* que converte dados discretos em dados difusos, tratáveis pela lógica *Fuzzy*.

- **DeFuzzyficação:** É a operação inversa de *Fuzzyficação*, ou seja, o controle *Fuzzy* converte dados difusos em dados discretos a serem aplicados como ações de controle em controle clássico

Modelagem e controle *Fuzzy* (GOMIDE, 1989; ZADEH, 1968), são técnicas usadas para manipular informações linguísticas ou verbais de um determinado sistema de forma rigorosa. A abordagem diferenciada e a grande simplicidade de implementação de controle *Fuzzy* reduz a complexidade de um projeto de controle de tais sistemas a tal ponto que problemas anteriormente mencionados apresentam soluções simples, conforme Shaw e Simões (1999).

3.1.1 Sistema de controle *Fuzzy*

Nesta seção serão revistos os conceitos básicos de lógica *Fuzzy* (TANAKA, 1997; GOMIDE, 1995; ZADEH, 1968); esta técnica é também chamada de lógica nebulosa e seu emprego em sistema de controle, atualmente, está bem difundido (ROSS 2010; SHAW; SIMÕES, 1999; TANAKA, 1997). Apesar de existir um complexo formalismo matemático justificando a aplicação da lógica *Fuzzy* em sistemas de controle, (TANAKA, 1997; KLIR, 1995), serão tratados neste trabalho somente os conceitos necessários para o desenvolvimento e entendimento deste trabalho.

Na teoria de conjunto clássica, um elemento pertence ou não pertence a um determinado conjunto; é o caso discreto classicamente conhecido. (ZADEH, 1968), o criador da lógica *Fuzzy*, propôs um tratamento mais amplo, quando afirmou que um determinado elemento pode pertencer a mais de um conjunto, com um grau de pertinência específico para cada conjunto. O parâmetro que caracteriza o grau de pertinência é denominado de fator de pertinência, μ_i que pode assumir valores no intervalo fechado $[0, 1]$, sendo que o valor zero indica a completa exclusão e valor um indica uma completa pertinência do elemento a um determinado conjunto. Em lógica *Fuzzy* é usual expressar o grau de pertinência de um elemento a um ou mais conjuntos através de figuras geométricas simples, tais como triângulos, trapézios, curvas gaussianas, etc.;

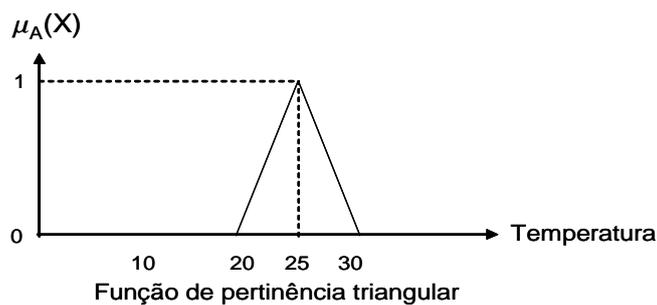


Figura 3.1 - Função de pertinência triangular

A Figura 3.1 apresenta um exemplo, usando um triângulo para expressar um conjunto *Fuzzy* de temperatura e o texto a seguir exemplifica como a pertinência é representada.

Como exemplo, para expressar a idéia de que uma temperatura está por volta de 25°C, pode-se usar uma função de pertinência triangular, conforme Figura 3.1, com o pico em 25°C, para sugerir a idéia de que quanto mais perto o valor da temperatura estiver do pico, mais este valor se identifica com a expressão linguística usada para expressar a temperatura. Genericamente **U** é uma coleção de elementos designados por {u} aqui representando temperatura e **U** é chamado de universo de discurso ou o universo das temperaturas, podendo ser contínuo ou discreto. Um conjunto *Fuzzy* **A** em um universo de discurso **U** é expresso por uma função de pertinência μ_A que assume valores no intervalo [0, 1], ou seja: $\mu_A: U \rightarrow [0,1]$ (GOMIDE, 1995).

3.1.2 Classificação dos raciocínios *Fuzzy*

Para a realização do raciocínio *Fuzzy* são necessárias regras denominadas de regras de inferência *Fuzzy*. Estas regras são dadas por expressões linguísticas da forma "Se...Então" ("if...Then") captadas da lógica usual do ser humano. Existem dois tipos de raciocínio *Fuzzy*; os dois tipos usam a mesma regra "se - então". Temos o tipo direto que apresenta três modalidades denominadas de método direto de Mandani, método de Sugeno e método Simplificado (TANAKA, 1997). Neste trabalho é usado o método de Mandani, por ser mais usual e menos complexo.

3.1.3 Operações com conjunto *Fuzzy*

Basicamente as operações com conjuntos *Fuzzy* são as mesmas operações definidas em conjuntos clássicos; para o presente trabalho é suficiente rever apenas as operações de união e interseção, assim, sendo **A** e **B** dois conjuntos *Fuzzy* contidos no conjunto universo **U**, com funções de pertinência μ_A e μ_B respectivamente, as operações de união **A**∪**B** e interseção **A**∩**B** são definidas pelas expressões (3.1) e (3.2), respectivamente:

$$\mu_{A \cup B}(u) = \mu_A(u) \cup \mu_B(u) \quad (3.1)$$

$$\mu_{A \cap B}(u) = \mu_A(u) \cap \mu_B(u) \quad (3.2)$$

Para se atribuir um significado determinístico aos termos linguísticos, ou seja, expressar em linguagem tipo baixa, média, alta, pequeno, grande, etc.; associa-se cada um desses termos linguísticos a um conjunto *Fuzzy* definido em um universo de discurso comum, conforme exemplo mostrado na Figura 3.2, em que são usadas as variáveis linguísticas baixa, média e alta para expressar sensação de temperatura; neste caso o conjunto universo é a temperatura, baixa, média e alta são conjuntos *Fuzzy* representados por trapézios e triângulo que compõem a citada figura, conforme segue

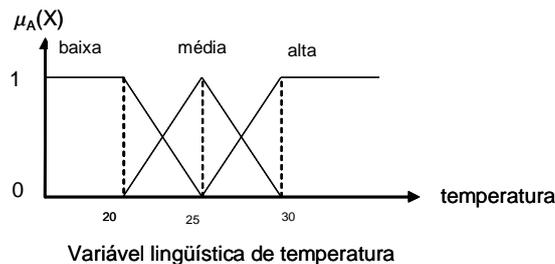


Figura 3.2-Variável linguística de temperatura

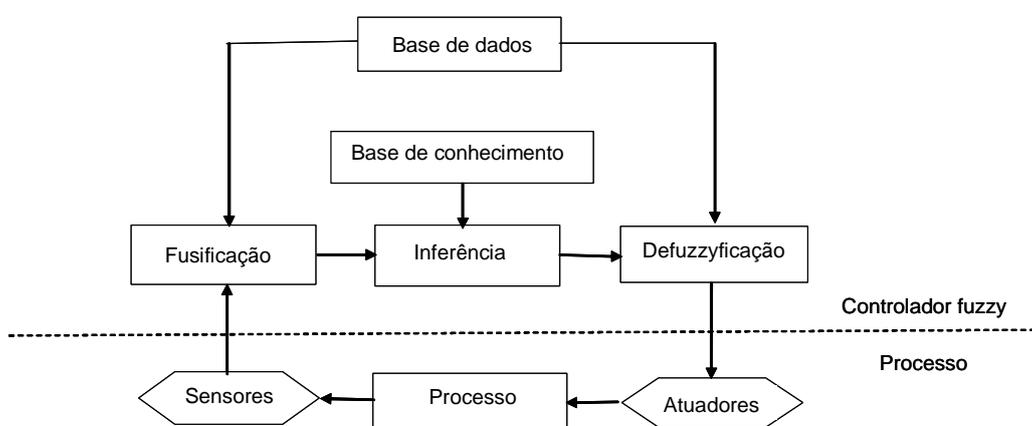
Fonte: Gomide et al (1994)

A ideia básica em controle *Fuzzy* é modelar as ações a partir de conhecimento de um especialista humano, ao invés do processo determinístico clássico, ou

seja, por exemplo: obtém-se de um operador humano, responsável pelo comando de uma determinada válvula de uma tubulação, de um sistema genérico o grau de atuação manual por ele exercida sobre a válvula, expressa em termos linguísticos tais como pequena abertura, média abertura ou grande abertura para exercer o necessário controle sobre o processo; estas informações são *Fuzzyficadas* pelo controlador *Fuzzy* como uma primeira etapa de controle.

3.1.4 Estrutura básica de um controlador *Fuzzy*

A estrutura básica de um controlador *Fuzzy* é ilustrada na Figura 3.3, em que são apresentados, na forma de blocos, os componentes básicos do controlador *Fuzzy* e do sistema a ser controlado; a função de cada componente é descrita a seguir:



Estrutura Básica de Um Sistema de Controle Fuzzy

Figura 3.3 - Estrutura básica de um sistema de controle *Fuzzy*

Fonte: Gomide et al. (1995)

A interface de *Fuzzyficação* é o componente do controlador que recebe as informações de entrada vinda do sensor, escalona seus valores para condicioná-los a um universo de discursos normalizados e *Fuzzyfica*-os, transformando-os, em conjuntos *Fuzzy*. A base de conhecimento consiste de

uma base de regras, que caracteriza a estratégia de controle e suas metas. A base de dados armazena as definições necessárias para realizar as seguintes operações: discretizações, normalização de universos de discurso, partições dos espaços de entrada e de saída e as definições de função de pertinência. O procedimento de inferência processa os dados *Fuzzy* de entrada, junto com as regras, de modo inferir as ações de controle *Fuzzy*. A interface de *deFuzzyização* transforma as ações de controle *Fuzzy* já inferidas em ação de controle não *Fuzzy*; em seguida executa o necessário condicionamento (volta para as grandezas físicas do sistema que está sendo controlado) enviando finalmente as informações de controle para os atuadores.

A forma usual, de se expressar conhecimento especialista é por meio de regras do tipo de condição-ação expressa usualmente pela forma linguística se-então, "If- then" , conhecida por regra de controle *Fuzzy*. (GOMIDE, 1995; ZADEH, 1962).

3.1.5 Exemplo da aplicação do método direto de Mandani

O método direto de Mandani é o mais usado em controle por lógica *Fuzzy* (GOMIDE, 1995; KLIR; YOUAN, 1995; TANAKA, 1997); deste último autor utiliza-se o exemplo apresentado a seguir:

Dados os conjuntos *Fuzzy* $A_1, A_2, B_1, B_2, C_1, C_2$ e as variáveis x, y e z , usando a expressão " se...então" constroem-se as duas regras a seguir:

$$\text{Regra 1: SE } x \text{ é } A_1 \text{ e } y \text{ é } B_1 \text{ ENTÃO } z \text{ é } C_1 \quad (3.3)$$

$$\text{Regra 2: SE } x \text{ é } A_2 \text{ e } y \text{ é } B_2 \text{ ENTÃO } z \text{ é } C_2 \quad (3.4)$$

A parte de cada regra à esquerda de ENTÃO é chamada de premissa e a parte à direita é chamada de consequência.

Supondo-se que $x=x_0$ e $y=y_0$ sejam entradas particulares de cada regra, o processo de raciocínio *Fuzzy* é expresso pelas relações (3.5) e (3.6), dadas a seguir:

$$c_1 = \mu_{A1}(x_0) \cap \mu_{B1}(y_0) \quad (3.5)$$

$$c_2 = \mu_{A2}(x_0) \cap \mu_{B2}(y_0) \quad (3.6)$$

As operações (3.5) e (3.6) são correspondentes às operações de interseção do conjunto clássico. Os conjuntos *Fuzzy* c_1 e c_2 correspondem às áreas hachuradas dos triângulos C_1 e C_2 da Figura 3.4.

A operação final é realizada pela expressão (3.7) e graficamente dada pela Figura 3.4 formada por funções de pertinências triangulares, dada seguir:

$$\mu_C(z) = C_1 \cup C_2 \quad (3.7)$$

Exemplo gráfico do processo de inferência fuzzy

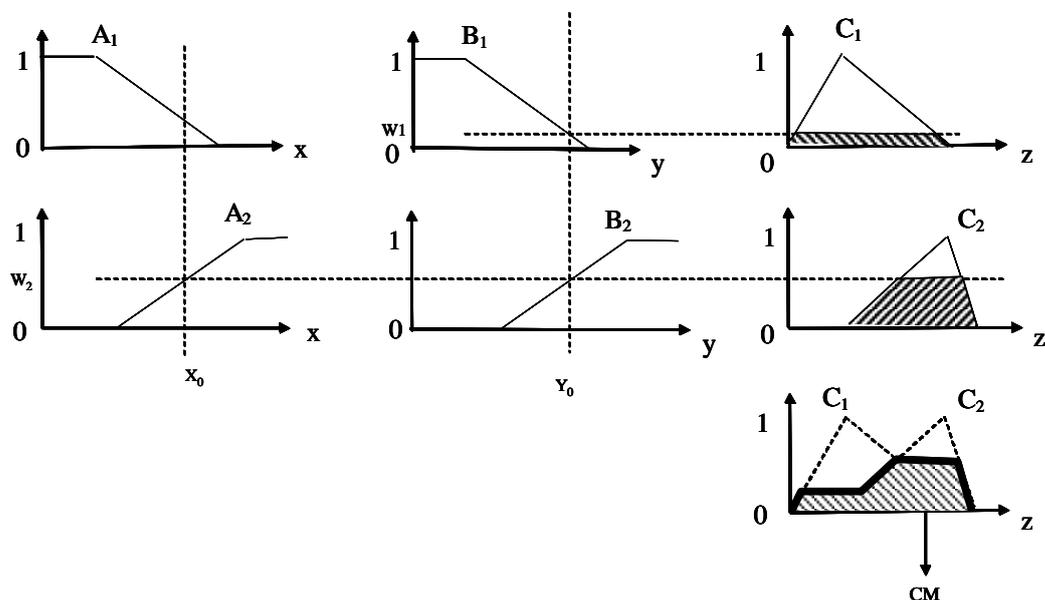


Figura 3.4-Processo de inferência f

Fonte: Tanaka (1996)

O subconjunto *Fuzzy* $\mu_C(z)$ corresponde graficamente à área da figura composta C_1C_2 da Figura 3.4; da área desta figura composta usa-se a centróide para expressar a grandeza *defuzzificada* a ser enviada para comando, (GOMIDE, 1995). A expressão deste centróide é dada pela expressão (3.7), que é o método da média ponderada, considerado um dos métodos computacionais mais eficientes, ver (ROSS, 2010). Na Figura 3.4 é apresentado o exemplo do processo de inferência *Fuzzy* quando existem duas regras; conforme se apresentou anteriormente, neste exemplo são regras de duas entradas e uma saída. Neste exemplo foram usadas funções de pertinência triangulares para o processo de *Fuzzyficação*. Após a inferência do controlador *Fuzzy*, é necessária a transformação da ação de controle *Fuzzy* para uma ação de controle não *Fuzzy* (*defuzzificação*) que represente a decisão *Fuzzy* a ser enviada ao processo; aqui o vocábulo processo é usado em um sentido muito amplo, podendo significar, por exemplo, desde componentes físicos, como atuadores usados em controle industrial, até

grandezas numéricas como parâmetros usados em programa de computador. Não existe, atualmente, nenhum processo sistemático para a escolha da estratégia de *deFuzzyficação*, (TANAKA 1997); o método mais usual é o critério do centro de massa, também denominado de centróide, do conjunto *Fuzzy* de saída, conforme mostrado na Figura 3.4 e cuja expressão matemática é dada pela expressão (3.8); esta expressão, portanto, calcula o centróide composto, para o qual contribuem as funções de pertinências de cada regra da parte antecedente.

$$c = \frac{\sum_{i=1}^n \mu_i \mu_{saída}(\mu_i)}{\sum_{i=1}^n \mu_{saída}(\mu_i)} \quad (3.8)$$

onde $\mu_{saída}(\mu_i)$ é a área de uma função de pertinência (como, por exemplo, \mathbf{A}_1 ou \mathbf{A}_2) modificada pelo resultado da inferência *Fuzzy* (como, por exemplo, 0.2 ou 0.8, respectivamente) e μ_i é a posição do centróide da função de pertinência individual, \mathbf{A}_1 e \mathbf{A}_2 respectivamente. A expressão (3.8) calcula o centróide composto, para o qual contribuem as funções de pertinência da parte antecedente de cada regra. O valor numérico deste centróide é a solução do problema tratado pelo controle *Fuzzy*.

3.1.6 Controlador por lógica *Fuzzy*

Nesta seção, usando-se os conceitos de lógica *Fuzzy* e de controle *Fuzzy*, cujos fundamentos e exemplos de funcionamentos, necessários a este trabalho, foram dados nas seções anteriores, implementa-se o controlador *Fuzzy*, usando funções de pertinência triangulares e trapezoidais, que será aplicado em toda esta proposta de tese.

As regras *Fuzzy* usadas para a construção do controle deste estudo são dadas em função de erro e e da variação do erro ep . Neste trabalho, o erro é obtido pela expressão $e = abs(\min(1 - y(:,3)))$, onde a componente de saída y_3 foi escolhida por ser a saída menos esparsa (tem maior densidade numérica); a

variação do erro foi obtida pela expressão $\dot{e} = \text{abs}(\min(\text{diff}(1 - y(:,3))))$. A Figura 3.5 apresenta a regra de controle *Fuzzy* com funções de pertinências triangulares e trapezoidais, usada neste trabalho, onde o erro e e sua variação ep são as entradas (estão na abscissa dos gráficos). É usado apenas o controlador cuja saída q formar a matriz diagonal de ponderação de estado, Q , tendo em vista que neste trabalho considera-se a matriz R uma matriz diagonal unitária; nada impede que esta matriz R possa ter alguns componentes diferentes da unidade em sua diagonal principal, para ponderar o esforço de forma diferenciada sobre determinados estados.

As ações de controle *fuzzy* são realizadas pelo algoritmo *FIS* (algoritmo existente no *Fuzzy Logic Toolbox* do Matlab). O programa fonte é obtido das sentenças lógicas dadas abaixo, que também correspondem à regra gráfica da Figura 3.5.

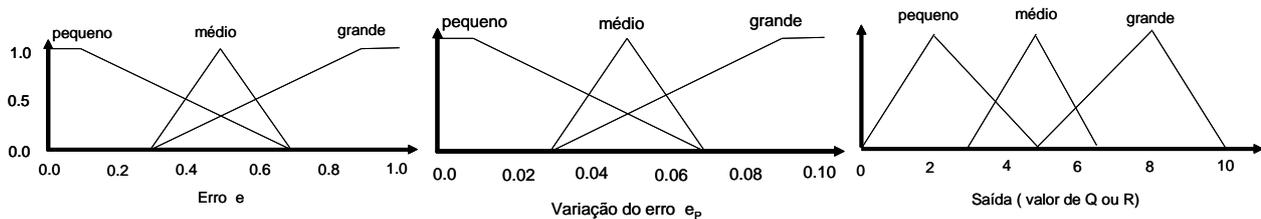


Figura 3.5 - Regra gráfica formada por funções de pertinências triangulares e trapezoidais

A Figura 3.5 usa funções de pertinências triangulares e trapezoidais para expressar o controlador *Fuzzy* fornecidas pelas sentenças lógicas dadas a seguir:

- 1) Se e é grande e ep é grande então Q é grande;
- 2) Se e é médio e ep é pequeno então Q é médio;
- 3) Se e é pequeno e ep é médio então Q é pequeno.

A Figura 3.5 juntamente com as regras servem de orientação para a construção da Tabela 3.1, dada a abaixo:

Tabela 3. 1 - Controle *Fuzzy*

Erro e Variação do erro ep	Pequeno	Médio	Grande
Pequeno	nf (nada a fazer)	médio	grande
Médio	pequeno	médio	médio
Grande	médio	pequeno	grande

A tabela de controle *Fuzzy* ou as sentenças lógicas dadas anteriormente, fornecem informações para o desenvolvimento do controlador *Fuzzy* usando a função *fis* existente no *Fuzzy Logic Toolbox* do Matlab.

3.2 Elementos de algoritmos genéticos

Os algoritmos genéticos (GA) são algoritmos de busca baseados no processo de seleção natural existente na natureza e no mecanismo da genética estudado e codificado pela primeira vez pelo inglês Charles Darwin. Como mecanismo de busca para fins gerais os algoritmos genéticos foram inicialmente proposto por John Holland em seu trabalho publicado por volta de 1975, com os objetivos de:

- Abstrair e explicar de forma rigorosa os processos evolutivos e adaptativos verificados em sistemas biológicos e
- Desenvolver programas de computador para simular os mecanismos naturais encontrados nestes sistemas.

Atualmente, a principal aplicação destes algoritmos é em problemas de busca generalizada: dado um conjunto de elementos, também denominados de conjunto de indivíduos, procura-se encontrar aquele ou aqueles que melhor se adaptem a certas condições especificadas; no caso deste trabalho estes algoritmos estão trabalhando como coadjuvante na busca de otimização.

Os algoritmos genéticos são metodologias heurísticas de otimização global, portanto se diferenciam das técnicas determinísticas que aplicam a derivada da função para encontrar o máximo ou o mínimo da função, desde que esta função seja uni nodal. É sabido que existem técnicas determinísticas mais complexas que garantem a solução ótima global, mesmo para funções multi nodais, contudo, o custo computacional do emprego destas técnicas é elevado, sendo esta uma das principais razões que justificam a utilização de heurísticas.

Os algoritmos genéticos, através de operadores genéticos e dos critérios de adaptabilidade, transformam uma população em uma nova população adaptada a uma nova realidade, atendendo a uma função denominada de função de aptidão.

Em uma dada população, cada indivíduo é uma possível solução que o GA procura melhorar para melhor atender o problema em estudo.

3.2.1 Definições básicas

Algoritmos genéticos (AG) é uma ferramenta atualmente muito usada em pesquisa e otimização de metodologias, que já está perfeitamente consolidada. Ela tem sido aplicada em muitas áreas da ciência e da engenharia e a principal razão para esse sucesso é, sem dúvida, devido ao desenvolvimento e difusão do computador e das técnicas computacionais. Da sua origem herdou toda sua nomenclatura, da qual algumas necessárias ao desenvolvimento deste trabalho, são dadas a seguir:

- **Indivíduo ou cromossomo:** É uma cadeia de caracteres numérico ou alfabético que

representa uma possível solução do problema proposto;

- **Gene ou genes:** É a unidade básica do cromossomo; É uma entidade que caracteriza o cromossomo;

- **População:** É o conjunto de todos os indivíduos (candidatos a solução);

- **Geração:** É o número de iteração executada pelo algoritmo que corresponde ao ciclo do computador que está executando o programa de GA;

- **Operações genéticas:** São as diversas operações que o GA executa sobre os cromossomos (indivíduos);

- **Espaço de Busca:** É o subconjunto do universo de possíveis soluções do problema a ser otimizado; e

- **Função de avaliação, ou função custo:** É a função utilizada pelo GA para determinar a qualidade de um indivíduo como solução do problema em estudo. ou seja, determina um valor numérico que expressa quão bem os parâmetros representados pelo indivíduo ou cromossomo solucionam o problema.

3.2.2 Diferenças entre os algoritmos genéticos e os métodos determinísticos

O método determinístico de otimização aplica-se a funções contínuas e bem comportadas e considera apenas um ponto maximizante destas funções; usando derivadas, determina para qual direção deve-se evoluir na busca do primeiro ponto de máximo ou de mínimo absoluto. O maior problema deste processo é que não existe nenhuma garantia de obtenção de um ponto extremo global para funções que apresentam mais de um máximo, ou seja, o processo determinístico convergirá para o extremo local mais próximo na

direção da busca; como dito anteriormente, existem métodos que garantem a solução ótima, porém nem sempre em tempo aceitável. Por esta razão tais processos funcionam corretamente apenas para funções uni modais, ou seja, para funções que apresentam apenas um extremo dentro do intervalo considerado.

Já os algoritmos genéticos apresentam uma estrutura versátil e robusta que é utilizada nos problemas de otimização, embora os GAs possam executar funções mais complexas e gerais, isto é, os GAs têm utilização mais abrangente, não servindo apenas para otimizar funções. Quando aplicados apenas em otimização de funções, os GAs diferem basicamente dos métodos determinísticos pelos seguintes aspectos (GOLDBERG, 1989):

- Trabalham com uma população de indivíduos.
- Operam com os códigos dos indivíduos e não com os próprios indivíduos.
- Operam com regras de transição probabilísticas.
- Não requerem dados iniciais do problema a ser otimizado.

3.2.3 Estrutura dos algoritmos genéticos

Existem numerosas estruturas de algoritmos genéticos, (LINDEN, 2012) com variações de procedimentos e operadores empregados, entretanto, neste trabalho foi utilizado o GA genérico cujo fluxograma é dado pela Figura 3.6 dada a seguir:

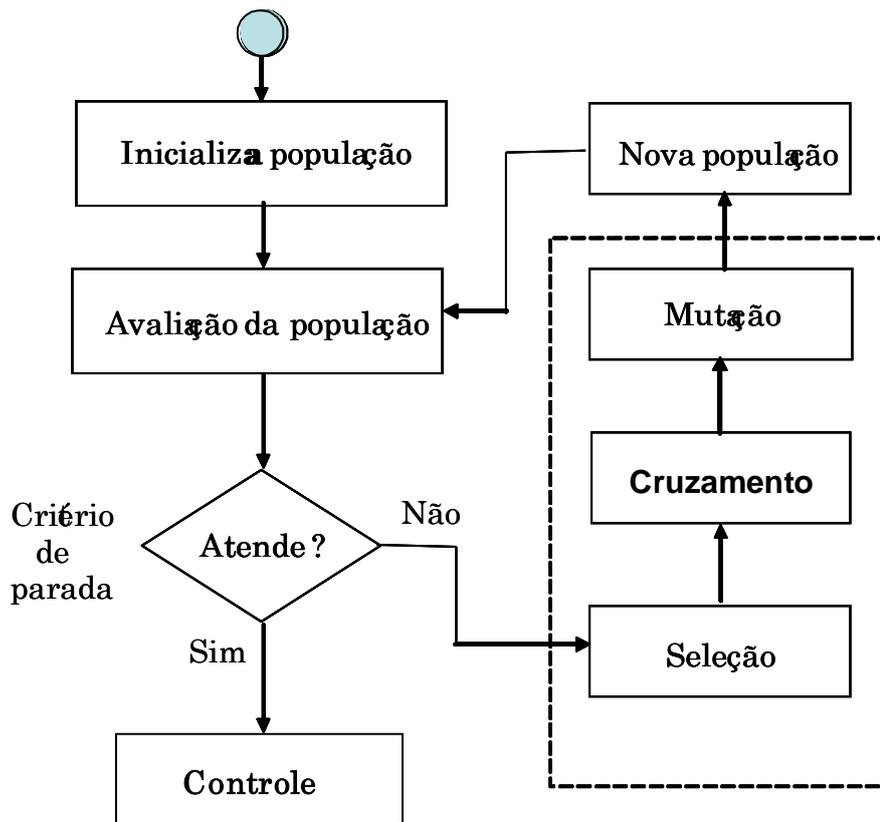


Figura 3.6 - Fluxograma do funcionamento do algoritmo genético
 Fonte: Basgalupp (2010)

3.2.4 Representação das variáveis em algoritmos genéticos

A representação das variáveis de um projeto usando GA tem influência decisiva no processo de busca; estas variáveis devem ser: as mais simples possíveis sem perder, contudo, as características de representatividade do problema em estudo.

Existem várias maneiras de representação destas variáveis, tais como: números binários, números inteiros ou números complexos. A grande maioria dos trabalhos empregando GA usa codificação binária (LINDEN, 2012) por ser uma codificação mais simples e uma das primeiras originalmente usadas em

codificação; nesta codificação o indivíduo é representado por um vetor composto por zeros e uns.

3.2.5 Geração da população inicial

A população inicial de indivíduos ou cromossomos é realizada de forma aleatória, embora em certas ocasiões, seja mais adequada uma seleção heurística desta população, ou seja, logo de início introduz-se um ou mais componentes nos indivíduos iniciais consideradas como relevantes, tais como: algumas soluções aproximadas que contenha algum tipo de informação que possa acelerar a busca. Vários trabalhos existentes (GOLDBERG, 1989; CASTRO, 2001) comprovam que população inicial não é crítica, desde que esta população contenha cromossomos suficientemente variados.

3.2.6 Avaliação da população

A avaliação da população é realizada pela função aptidão, que deve quantificar a qualidade de cada indivíduo da população. Em problemas de otimização a função de aptidão está intimamente relacionada com a função objetivo que se deseja maximizar ou minimizar.

Dada a generalidade dos algoritmos genéticos simples (GAs), a função de avaliação, em muitos casos, é a única ligação real do programa com o problema real (LINDEN, 2012). O conceito mais importante a ter em mente é que a função de avaliação deve refletir os objetivos a serem alcançados na resolução de um problema, visto que derivam das condições imposta deste problema. O resultado da função de avaliação é sempre é um número escalar.

3.2.7 Mecanismo de seleção

O processo ou mecanismo de seleção, como uma emulação do processo de seleção natural da biologia, baseia-se no princípio da sobrevivência dos indivíduos mais bem adaptados; estes indivíduos por terem maior aptidão são mais prováveis de serem copiados para dar origem a um novo conjunto de indivíduos, denominado de população intermediária, de onde serão aleatoriamente escolhidos novos indivíduos para nova reprodução. Os indivíduos com baixa aptidão têm menores chances de serem selecionados, conforme a pressão da seleção; vários mecanismos de seleção (BARBOSA, 1996) já foram propostos e implementados na literatura sobre algoritmos genéticos. O mecanismo de seleção do GA clássico é proporcional à adaptação do elemento a ser selecionado; esta adaptação é quantificada por uma função denominada de função objetiva ou função de avaliação do indivíduo. A seleção final dos indivíduos em função de seu grau de adaptação é geralmente realizada através de um algoritmo denominado de roleta; esta roleta virtual é constituída por um programa de computador que simula o funcionamento de uma roleta mecânica. A seguir é apresentado um exemplo de funcionamento da seleção usando a roleta:

Exemplo:

N	Cromossomo	Adaptação	Graus
1	0001100101010	6,0	180
2	0101001010101	3,0	90
3	1011110100101	1,5	45
4	1010010101001	1,5	45

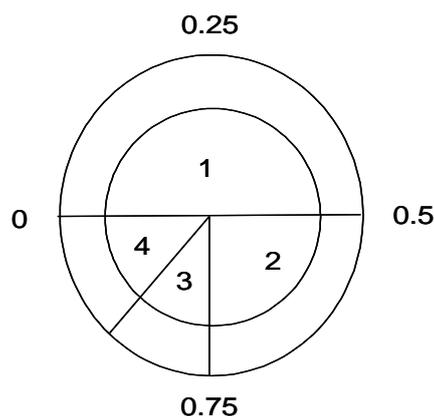


Figura 3.7 - Ilustração do funcionamento da roleta

Fonte: Castro e Von Zuben (2002)

Pelo exemplo, nota-se que o cromossomo ou indivíduo mais bem adaptado é expresso por um maior setor circular no diagrama simbolizando a roleta, apresentado à direita, na Figura 3.7.

Outro mecanismo de seleção muito usado a seleção por torneio: trata-se de sistema simples que apresenta bons resultados e é muito usado. Consiste em promover um torneio entre um grupo de N , $N \geq 2$, indivíduos aleatoriamente extraídos da população; o indivíduo com maior aptidão é selecionado e os menos aptos são descartados. Este mecanismo apresenta as vantagens listadas a seguir:

- Não gera convergência prematura;
- É dinâmico, combatendo a estagnação;
- Não requer esforço computacional extra;
- Apresenta mecanismo funcionamento muito próximo do processo natural biológico

Ainda dentre os mecanismos de seleção tem-se um mecanismo denominado de elitista, normalmente trabalhando juntos com outros métodos com o objetivo de acelerar a convergência do algoritmo. Este processo copia os indivíduos mais bem adaptados da população para formar a população seguinte, assegurando que estes bem qualificados não sejam destruídos nas etapas de mutação e recombinação. A desvantagem deste mecanismo é forçar o processo de busca otimizando de forma artificial a qualidade do indivíduo, na direção de algum ponto ótimo local que tenha sido encontrado antes do ponto ótimo global.

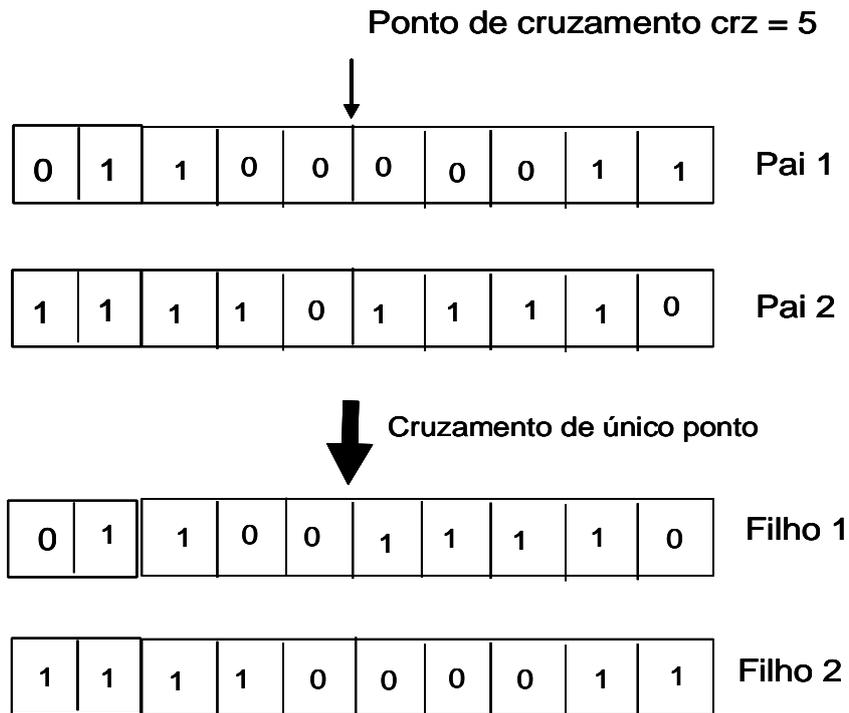
3.2.8 Mecanismo de reprodução

O mecanismo de reprodução é formado por operadores genéticos. Estes operadores são responsáveis pela introdução de novos indivíduos na população e pela manutenção da diversidade destes indivíduos. Nas seções seguintes são apresentados, de forma resumida, os principais e mais usuais operadores.

3.2.9 Reprodução por cruzamento (crossover simples)

Este tipo de reprodução não introduz novos indivíduos na população, apenas os indivíduos denominados pais geram novos indivíduos, denominados filhos por permuta de genes entre si, a partir de um ponto, denominado de ponto de cruzamento ou (crossover). Na fase de reprodução, o algoritmo procura criar indivíduos mais adaptados (melhores solução) pelo processo de cruzamento conforme descrito a seguir:

Inicialmente toda a população é agrupada aleatoriamente aos pares, formando $N/2$ genitores. O cruzamento é iniciado por um processo estocástico de decisão. Assim de cada par, dois novos indivíduos são gerados, conforme ilustrado no diagrama da Figura 3.8; nessa figura observa-se que do ponto de cruzamento é ponto $r = 5$, que é válido para o pai 1 e pai 2, estes genitores geraram o filho 1 e o filho 2 apenas combinando de forma cruzada seus genes alocados a sua direita e a sua esquerda com referência ao ponto de cruzamento, como dito acima, válido para os dois pais.



Crossover de um único ponto para um indivíduo de comprimento $l = 10$

Figura 3.8 - Ilustração do funcionamento da reprodução por cruzamento

Fonte: Castro (2001)

A Figura 3.8 ilustra a reprodução por cruzamento em um único ponto, que é usado neste trabalho, contudo, o GA pode fazer reprodução usando múltiplos pontos de cruzamentos que é uma generalização da ideia de troca de material genético, em que muitos pontos de cruzamentos são utilizados.

3.2.10 Reprodução por mutação

A mutação em GA é realizada pelo operador de mutação; mutação é um tipo de reprodução necessária para introduzir e manter a diversidade genética da população. Na mutação são alteradas aleatoriamente as características de um ou mais genes do indivíduo, conforme ilustrado na Figura 3.9. Conforme

ilustrado nesta Figura, o último gene do indivíduo original era zero e, por ação do operador de mutação, mudou para a unidade.

A mutação oferece meios para a introdução de novos elementos na população, assegurando uma probabilidade não nula ao término do processo de busca (BARBOSA, 1996). A taxa de mutação geralmente é baixa, em torno de 0,001 mutação por bit do indivíduo, ou seja, considerando um indivíduo ou cromossomo contendo 100 genes (chamado algumas vezes de bits), a taxa de mutação permitida para este indivíduo é de 0,1%; é próximo disto que ocorre na genética natural; A mutação atualmente é considerada uma operação secundária, embora existam alguns pesquisadores que acham que ela tem maior importância, ver (LINDEN, 2012).

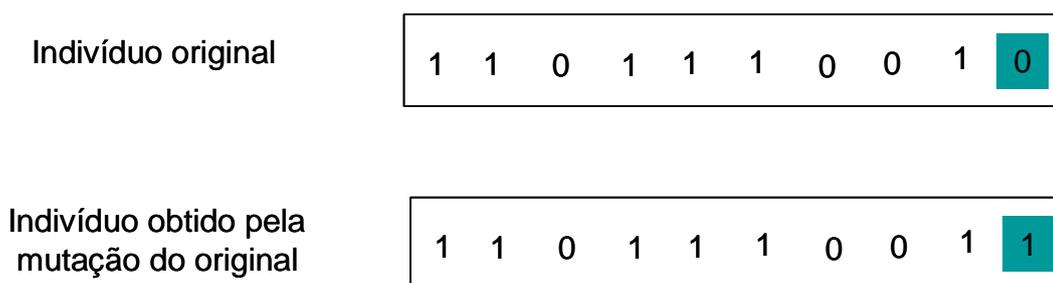


Figura 3.9 - Ilustração da operação de mutação
Fonte: Linden (2012)

3.2.11 Eficiência de um algoritmo genético

A eficiência de um GA é fundamentalmente dependente de seus parâmetros; a seguir é feita uma revisão destes parâmetros:

3.2.12 Tamanho da população – N ou L

O tamanho da população tem influência direta na diversidade de soluções: deve haver, portanto um compromisso ao se adotar o tamanho deste parâmetro de forma a se ter diversidade de soluções compatível com problema em estudo, sem, contudo, sobrecarregar o sistema computacional. Usualmente este parâmetro é designado por N ou L e a faixa de valores sugerido pela literatura é 50 a 100 indivíduos (MITCHELL, 1999).

3.2.13 Taxa de probabilidade de cruzamento – pc

Este parâmetro indica a probabilidade com a qual ocorrerá o cruzamento entre os indivíduos da população; quanto maior for a taxa de probabilidade de cruzamento, mais rapidamente novos indivíduos serão introduzidos na população. Deve ser observado um compromisso com relação ao valor de pc: se a probabilidade for muito alta, indivíduos com boas aptidões poderão ser descartadas mais rapidamente da população; valores muito baixos de pc pode tornar a convergência muito lenta. Valor usual para a taxa de pc está em torno de 0,7 (MITCHELL, 1999).

3.2.14 Taxa de probabilidade de mutação – pm

Este parâmetro determina a probabilidade com a qual ocorrerá a mutação nos cromossomos da população durante sua evolução; conforme dito anteriormente, a mutação é a responsável por introduzir novos indivíduos à população, aumentando sua diversidade, possibilitando maior varredura no espaço de busca. A taxa de mutação deve ser baixa, da ordem de 0,001 por bit (MITCHELL, 1999; JONG, 2002) para a busca não se tornar excessivamente aleatória.

3.3 Conceitos básicos de controle H_{∞}

O objetivo do controle H_{∞} é sintetizar um controlador que, quando o sistema a ser controlado estiver em malha fechada, mantenha as variáveis a serem controladas dentro de limites especificados, mesmo na presença de incertezas paramétricas ou não estruturadas. Estas incertezas abrangem uma vasta gama de desvios, destacando-se como casos mais importantes o erro, quase sempre desconhecido entre o modelo e sistema físico, grandezas de sinais e ruídos, entre outros parâmetros. Esta técnica de controle é uma técnica multivariável, visto que trata com sistemas com mais de uma variável, e é também robusta porque mantém as variáveis dentro de limites especificados, mesmo na presença de incertezas.

O conceito de robustez de um sistema de controle, do ponto de vista físico conforme tratado pela literatura de controle robusto, é a manutenção da estabilidade do sistema mesmo na presença de determinada faixa de distúrbios. Do ponto de vista matemático, o conceito de robustez de um sistema cuja função de transferência seja $\mathbf{G}(s)$, com uma incerteza $\Delta(s)$ é expresso considerando-se um número ϵ suficientemente pequeno ($0 < \epsilon < 1$), de forma a atender às relações $\|\Delta\|_{\infty} \leq 1/\epsilon \Leftrightarrow \|\mathbf{G}(s)\|_{\infty} < \epsilon$ e $\|\Delta\|_{\infty} < 1/\epsilon \Leftrightarrow \|\mathbf{G}(s)\|_{\infty} \leq \epsilon$ (ZHOU, 1995). Muitos dos conceitos básicos deste controle derivaram do controle clássico e do controle moderno.

Nesta seção é revisto alguns conceitos básicos de sistemas multivariáveis e controle H_{∞} necessários para este trabalho. Em alguns trechos deste trabalho, com o objetivo de simplificar as notações as funções de transferência estão em negrito porem sem presença do "s" entre parênteses "(s)", conforme é usual na literatura usada (SKOGESTAD, 2007; MACIEJOWISKY, 1989).

3.3.1 Diagrama típico de controle clássico

Um sistema de controle clássico pode ser representado pela Figura 3.10 abaixo:

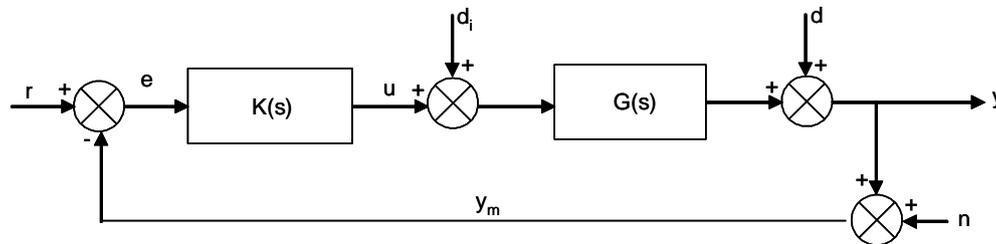


Figura 3.10 - Configuração de controle com um grau de liberdade

Fonte: Skogestad (2007)

onde **G** é a função de transferência da planta e **K** é o controlador, **r**, **e** e **y** são os sinais de referência de entrada, sinal de erro e sinal de saída, respectivamente.

A Figura 3.10 representa um sistema mais próximo de um sistema real, visto que considera ruídos de sensores dado por **n**, distúrbio de entrada **d_i** e distúrbios de saída da planta dados por **d**. Além destes ruídos podem estar presentes incertezas de modelo, devido à imprecisão de modelamento, não representados na Figura em questão, conforme já referido anteriormente, e/ou variações de parâmetros durante o funcionamento do sistema planta-controlador. É desejável, portanto, projetar um controlador que garanta a estabilidade do sistema mesmo na presença destes ruídos e incertezas. A planta da Figura 3.10 pode ser representada também por matrizes numa configuração denominada por espaço de estado conforme dada pelas expressões a seguir:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad \text{com} \quad x(t_0) = x_0 \quad (3.9)$$

$$y(t) = Cx(t) + Du(t) \quad (3.10)$$

em que **A** é a matriz de estado que, no caso mais geral em que o sistema é de ordem **n**, é de dimensão $n \times n$, **B** é a matriz de entradas de dimensão $n \times r$, **C** é a matriz de saídas de dimensão $n \times m$ e **D** é a matriz de entradas diretas cuja dimensão é $m \times r$; para o caso linear todas estas matrizes são consideradas constantes.

Aplicando-se a transformada de Laplace às equações (3.9) e (3.10) pode-se obter a função de transferência matricial do sistema $G(s)$ de $Y(s)$ em função de $U(s)$ dada a seguir:

$$G(s) = C(sI - A)^{-1}B(s) + D(s) \quad (3.11)$$

No caso de a matriz **D** ser nula, a função de transferência na forma numerador/denominador é estritamente própria.

3.3.2 Relações fundamentais

Da figura 3.10, ver Pagina 50, tiram-se as expressões a seguir:

$$y = d + GK(r - n - y) \quad (3.12)$$

rearranjando a expressão (3.12), obtém-se:

$$(I - GK)y = d + GK(r - n) \quad (3.13)$$

Ainda da Figura 3.10 obtém-se as expressões;

$$S = (I - GK)^{-1} \quad (3.14)$$

$$T = (I - GK)^{-1} GK = SGK \quad (2.15)$$

em que S e T são conhecidas por função sensibilidade e função complementar respectivamente. T é conhecida também por função de transferência de malha fechada e é dita complementar devido à expressão resultante da soma dos segundos membros das expressões (3.14) e (3.15), ou seja:

$$S + T \equiv I \quad (3.16)$$

A expressão (3.16) mostra que se aumentando \mathbf{S} , diminui-se \mathbf{T} e vice versa; esta identidade aponta para uma solução de compromisso, que é fundamental, no projeto de sistema de controle multivariável, uma vez que \mathbf{S} e \mathbf{T} não podem ser simultaneamente grandes ou pequenos.

Para mensurar estas matrizes é usado o conceito de valor singular máximo de uma matriz, que fornece o ganho máximo da planta para qualquer direção do sinal de entrada; o valor singular máximo é representado por $\bar{\sigma}(\cdot)$ e o valor singular mínimo por $\underline{\sigma}(\cdot)$ (SKOGESTAD, 2007; ZHOU, 1995).

Dada uma matriz complexa \mathbf{G} , valores singulares de \mathbf{G} são raízes quadradas positivas dos autovalores de $\mathbf{G}^H \mathbf{G}$, onde \mathbf{G}^H é a transposta conjugada de \mathbf{G} , ou seja,

$$\sigma_i = \sqrt{\lambda_i \mathbf{G}^H \mathbf{G}} \quad (3.17)$$

em que λ_i são os autovalores de \mathbf{G} (SKOGESTAD, 2007).

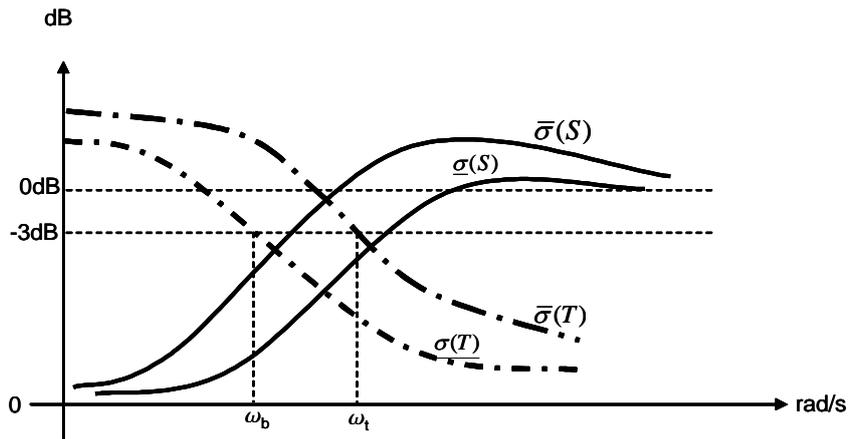
3.3.3 Compromisso de projeto entre s e t em função de seus valores singulares

Lembrando-se que \mathbf{S} é a função sensibilidade (3.14) e que \mathbf{T} é a função sensibilidade complementar (3.15) e tendo em mente (3.16), para obter-se um bom projeto de controle multivariável deve respeitar os requisitos a seguir (SKOGESTAD 2007; MACIEJOWISKI, 1989; SOUZA, 2006; ZHOU, 1989).

- a) **Sensibilidade:** Manter \mathbf{S} pequena, ou seja, manter $\bar{\sigma}(S)$ pequeno;
- b) **Rejeição de ruídos:** Manter \mathbf{T} pequena, ou seja, manter $\bar{\sigma}(T)$ pequeno;
- c) **Rastrear a referência:** Manter \mathbf{T} grande, ou seja, manter $\underline{\sigma}(T)$ grande e
- d) **Minimizar energia de controle:** Fazer \mathbf{SK} pequena, ou seja, fazer $\bar{\sigma}(SK)$ pequeno.

Devido à (3.16) é inevitável o conflito entre alguns itens da lista anterior, assim: o item (a) conflita com item (b); o item (b) conflita com o item (c) e o item (a) conflita com os itens (c) e (d). Contorna-se este problema moldando-se os ganhos (valores singulares) de \mathbf{S} e \mathbf{T} convenientemente, de forma que \mathbf{S} e \mathbf{T} tenham tamanhos adequados em faixas de frequências pré-fixada, sempre respeitando expressão (3.16).

A moldagem de \mathbf{S} e \mathbf{T} aproveita o fato prático observado de que os sinais de perturbação \mathbf{d} e sinal de referência \mathbf{r} ocorrem na faixa de baixa frequência, sendo conveniente, portanto, adotar \mathbf{S} pequena e \mathbf{T} grande na faixa de baixa frequência. Os sinais de ruído, em geral, ocorrem na faixa de alta frequência, sendo conveniente que \mathbf{T} seja pequeno nesta faixa. A Figura 3.11 ilustra o que foi explanado no texto acima.



Formatação desejada para S e T

Figura 3.11 - Representação das formas desejadas para **S** e **T**

3.3.4 Definição de norma H_{∞} de uma função de transferência

Dada uma função de transferência $G(s)$, sua norma H_{∞} é simplesmente o valor de pico da resposta de $G(s)$ no domínio da frequência, ou seja:

$$\|G(s)\|_{\infty} = \max_{\omega} |G(j\omega)| \quad (3.18)$$

Uma forma simbólica usual de expressar $\|G\|_{\infty}$, ver (Maciejowski 1989) é por:

$$\|G\|_{\infty} = \sup_{\omega} \bar{\sigma}(G) \quad (3.19)$$

A expressão (3.19) mostra que a norma H_{∞} é o valor supremo do maior valor singular de $\mathbf{G}(s)$ e expressa o máximo ganho de energia que pode ocorrer entre a entrada e a saída do sistema. Este fato está ilustrado na Figura 3.12, a seguir:

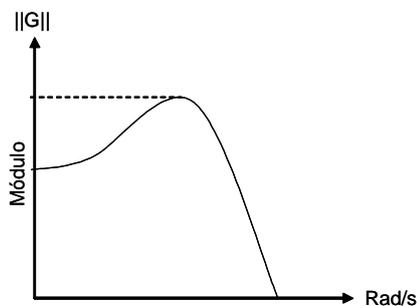


Figura 3.12 - Norma H_{∞} de $G(s)$

3.3.5 Formulação geral do problema de controle H_{∞}

A Figura 3.10 expressa sistemas com um grau de liberdade; sistemas com n graus de liberdade são representados pela Figura 3.13, conhecida como planta generalizada. Nesta Figura, P é planta generalizada, K é controlador, w são entradas exógenas, tais como ruídos, comandos e distúrbios, z são sinais de saídas exógenas como erro ou outros sinais a serem minimizados, v é a entrada do controlador (sinal medido vindo da planta) e u é o sinal de controle (MACIEJOWSKI, 1989; SOUZA, 2006).

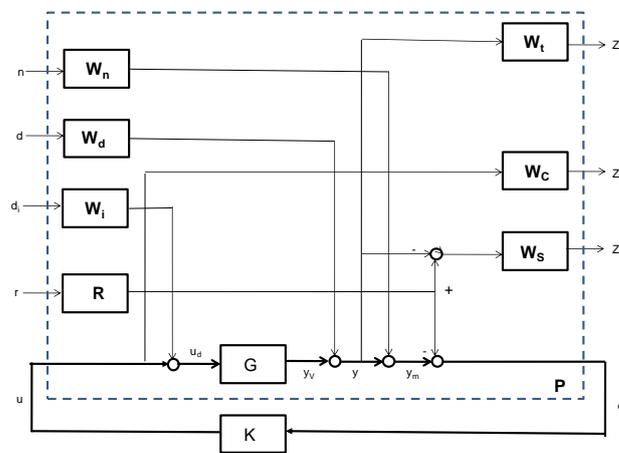


Figura 3.13 - Sistema multivariável

Fonte: Cardenas (2005)

Na Figura 3.13, temos as variáveis de entrada exógenas n , d , d_i , e r , ponderadas pelas matrizes W_n , W_d , W_{di} , e R respectivamente e as variáveis de saída z_1 , z_2 e z_3 , ponderadas pelas matrizes W_t , W_c e W_s respectivamente.

A simplificação da Figura 3.13, geralmente usada na literatura, é apresentada pela Figura 3.14, a seguir:

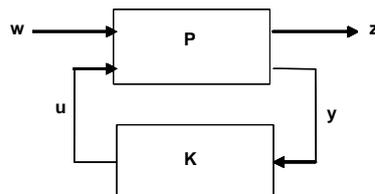


Figura 3.14 - Planta generalizada

3.3.6 Descrição matemática do sistema

A descrição matemática do sistema acima é apresentada pela equação matricial dada a seguir, na qual a matriz \mathbf{P} do sistema apresentado na Figura 3.14 é particionada conforme (3.20):

$$\begin{bmatrix} z \\ v \end{bmatrix} = P(s) \begin{bmatrix} w \\ u \end{bmatrix} = \begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} \quad (3.20)$$

A notação compacta da planta generalizada \mathbf{P} é dada pela matriz (3.21), a seguir:

$$P = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} \quad (3.21)$$

O diagrama da Figura 3.14, quando em malha fechada pode ser representado pela Figura 3.15 (SKOGESTAD, 2007).

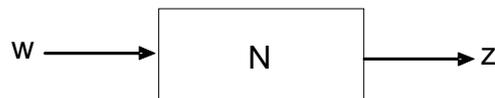


Figura 3.15 - Diagrama bloco geral

O erro z deve ser minimizado para que sejam alcançados os objetivos de controle. A função de transferência de malha fechada entre o erro z e entrada w é dada pela função transformação fracional linear, dada por:

$$z = F_1(P, K)w \quad (3.22)$$

em que a função F_1 é expressa por:

$$F_1(P, K) = P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21} \quad (3.23)$$

Supondo que sejam verificadas as relações a seguir em relação às matrizes de $\mathbf{G}(s)$ (SKOGESTAD, 2007):

$$(A1) \quad (A, B_2, C_2) \text{ é estabilizável e detectável;} \quad (3.24)$$

$$(A2) \quad D_{12} \text{ e } D_{21} \text{ tenha posto pleno;} \quad (3.25)$$

$$(A3) \quad \begin{bmatrix} A - j\omega I & B_2 \\ C_1 & D_{12} \end{bmatrix} \text{ tenha posto pleno de coluna para todo } \omega \quad (3.26)$$

$$(A4) \quad \begin{bmatrix} A - j\omega I & B_1 \\ C_2 & D_{21} \end{bmatrix} \text{ tenha posta pleno de linha para todo } \omega \quad (3.27)$$

$$(A5) \quad D_{11} = 0 \text{ e } D_{22} = 0; \quad (3.28)$$

$$(A6) \quad D_{12} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ e } D_{21} = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (3.29)$$

$$(A7) \quad D_{12}^T C_1 = 0 \text{ e } B_1 D_{21}^T = 0 \quad (3.30)$$

$$(A8) \quad (A, B_1) \text{ é estabilizável e } (A, C_1) \text{ é detectável.} \quad (3.31)$$

O problema de controle H_{∞} ótimo é determinar todos os controles estabilizantes K que minimize a função transformação fracional linear dada por (3.32), a seguir:

$$\|F_1(P, K)\|_{\infty} = \max_{\omega} \overline{\sigma}(F_1(P, K)(j\omega)) \quad (3.32)$$

Isto, também tem uma interpretação no domínio do tempo como a norma dois induzida; Seja $e = F_1(P, K)\omega$; então: $\|F_1(P, K)\|_{\infty} = \max_{\omega} \overline{\sigma}(F_1(P, K)(\omega))$

$$\|F_1(P, K)\| = \max_{\omega(t)=0} \frac{\|z(t)\|_2}{\|\omega(t)\|_2} \quad (3.33)$$

$$\text{em que } \|z(t)\|_2 = \sqrt{\int_0^x \sum_i |z_i(t)|^2 dt} \text{ é a norma dois do vetor.} \quad (3.34)$$

3.3.7 Sensibilidade mista

Sensibilidade mista é a denominação dada a um tipo de modelamento da forma da função de transferência do sistema a ser projetado, em que a função sensibilidade **S** é moldada juntamente com uma ou mais funções de transferência em malha fechada como, por exemplo: **KS**, **KT**, onde **K** é, como visto anteriormente, a função de transferência do controlador (SKOGESTAD, 2007), obtemos uma estrutura denominada de planta aumentada mostrada na Figura 3.16, que através das funções de ponderações $w_s(s)$, $w_u(s)$ e $w_T(s)$, procura-se alcançar os objetivos de projeto, tais como desempenho e robustez (ZHOU; DOYLE 1995).

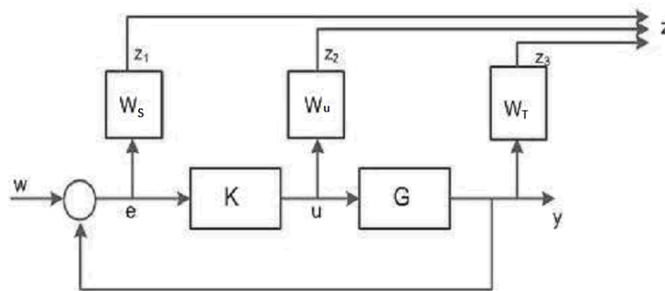


Figura 3.16 - Diagrama da planta aumentada

Fonte: Cardenas (2005)

Pode-se através desta metodologia, sintonizar controladores de acordo com os requisitos impostos, como exemplo, para atenuação de distúrbios do sistema, projeta-se um controlador que minimize $\|w_s S\|_\infty$, se bem que, na prática, é mais comum a minimização da norma H_{INFINITO} da função mista dada a seguir:

$$\begin{bmatrix} w_s S \\ w_u KS \end{bmatrix}_\infty \quad (3.35)$$

onde é minimizado tanto a função sensibilidade quanto a energia de controle.

A técnica da sensibilidade mista permite a ponderação simultânea das malhas **S** e **T** do sistema a ser controlado, possibilitando obter a largura de faixa do sistema a ser controlado, bem como a limitação das malhas **S** e **T** dentro de contornos pré-estabelecidos. A Figura 3.17 mostra graficamente os requisitos

de desempenho, robustez e redução de energia obtidos pelo uso da técnica de sensibilidade mista.

Projeto de compromisso para a função de transferência multivariável GK

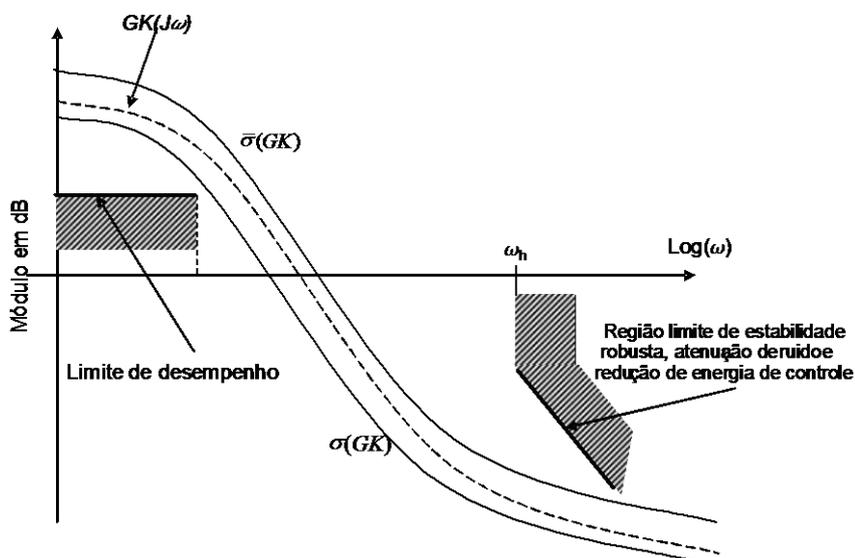


Figura 3.17 - Representação gráfica da função de transferência GK

Fonte: Skogestad e Postlethwaite (2005)

A largura de faixa do sistema é definida como a faixa de frequência $[\omega_1, \omega_2]$, dentro da qual o controle deve atuar. Usualmente a largura de faixa de malha fechada ω_b é definida pela frequência onde $|S(j\omega)|$ cruza pela primeira vez o nível de -3dB, quando em sua subida. A frequência de cruzamento ω_{CR} é definida como a frequência onde $|L(j\omega_c)|$ cruza pela primeira vez o nível de 1 dB, vindo de cima. As Figuras 3.18 e 3.19 ilustram as explicações acima; a Figura 3.20 mostra a largura de faixa de frequência gerada pelas curvas das funções sensibilidade **S** e função complementar **T**.

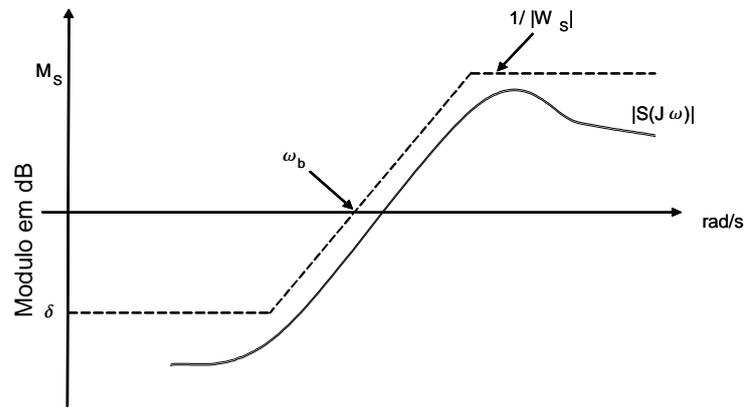


Figura 3.18 - Função de ponderação W_s

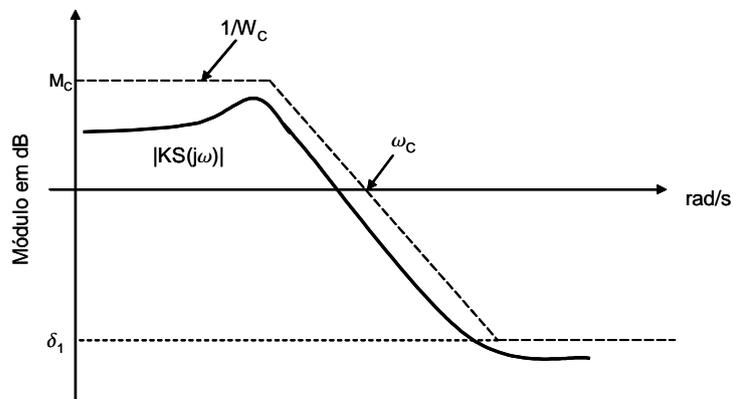


Figura 3.19 - Função de ponderação W_c

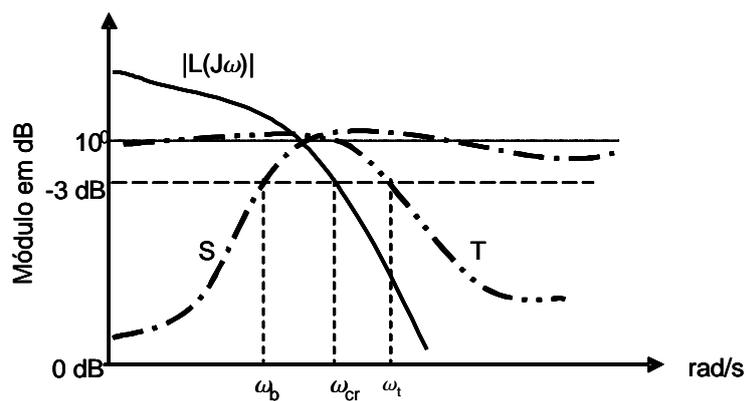


Figura 3.20 - Função sensibilidade S e função Complementar T

Geralmente, a relação $\omega_b < \omega_{cr}$ coloca um limite inferior na largura de faixa do sistema; para determinar superior para esta faixa de frequência, usa-se a

função de sensibilidade complementar $T(J\omega)$. Ainda, nas Figuras 3.18 e 3.19, vê-se que as inversas das funções de ponderação W_S e W_C comportam-se como filtros passa alta e passa baixa respectivamente. A Figura 3.20 mostra a faixa de passagem com seu limite inferior ω_b e superior ω_t , bem como a frequência de corte ω_{cr} .

Considerando que $T=I-S=GKS$, a determinação dos limites inferior e superior da largura de faixa, a formatação e a taxa de decaimento podem ser obtidas formatando-se **GKS**. Esta formatação é obtida com função de ponderação grande em altas frequências. A ponderação $\|W_T T\|_\infty < 1$ e $\|W_C C\|_\infty < 1$ pode ser obtida combinando estas relações em uma matriz **N**, dada por:

$$N = \begin{bmatrix} W_S S \\ W_T T \end{bmatrix} \quad (3.36)$$

Esta configuração garante robustez relativa às perturbações aditivas, boa rejeição de distúrbio e limite de faixa de passagem do controlador.

3.3.8 Escolha das funções de ponderação W_S E W_C

Os conceitos apresentados nesta seção foram baseados em Skogestad (2007) e Zhou e Doyle (1995). Assim, considerando as especificações de projeto: largura de faixa ω_b , pico máximo de **S** menor que M_s e erro de rastreamento menor que ϵ , em ϵ é o erro de regime permanente, usa-se a expressão (3.37), dada a seguir para se obter a ponderação W_S , conforme Skogestad e Zhou, citados acima

$$W_S = \frac{\frac{s}{M_s} + \omega_b}{s + \delta_s \omega_b} \quad (3.37)$$

Para se obter uma ponderação mais severa, embora aumentando a ordem do controlador, pode-se usar a expressão (3.38), dada a seguir:

$$W_s = \left(\frac{\frac{s}{\sqrt{M_s}} + \omega_b}{s + \sqrt{\epsilon\omega_b}} \right)^2 \quad (3.38)$$

Para a função de ponderação do controle, usa-se a expressão $u = \mathbf{KS}(r-n-d)\text{-Td}$; a magnitude de **KS** em baixas frequências é limitada principalmente pelo custo de controle e pelo efeito de saturação dos atuadores, como é o caso das rodas de reação (CARDENAS, 2005; SKOGESTAD, 2007). Uma função de ponderação de controle, \mathbf{W}_c , usual adotadas pelos autores Skogestad e Zhou, citados acima, é dada por (3.39), a seguir:

$$W_c = \frac{s + \frac{\omega_{bc}}{M_c}}{\delta_c s + \omega_{bc}} \quad (3.39)$$

com ϵ muito pequeno, geralmente menor que 0,02, tendo em vista este parâmetro é o erro de regime permanente (ZHOU; DOYLE, 1995).

3.3.9 Controlador Hinfinito sub-ótimo

Devido às dificuldades teórica e numérica encontradas no projeto de controlador H_{INFINITO} ótimo e considerando que nem sempre é necessário um controlador ótimo, na prática é mais simples e suficiente projetar um controlador sub-ótimo conforme Zhou (1995) e Konstantinov (1989). Para isto, especificado um valor γ , procura-se um controlador tal $\|F_i(P, K)\|_{\infty} < \gamma$. Para se aproximar de controlador ótimo reduz-se iterativamente o valor de γ até que um mínimo satisfatório, especificado pelo projetista, seja alcançado. Portanto, encontrar um controle H_{INFINITO} sub-ótimo resume-se em: dado um $\gamma < \gamma_{\text{min}}$, achar todos os controladores estáveis **K** que satisfaça desigualdade a seguir:

$$\|F_i(P, K)\|_{\infty} < \gamma \quad (3.40)$$

3.3.10 Procedimentos básicos para projeto de um controlador H_{∞}

O projeto de um controlador robusto H_{∞} , como visto anteriormente, requer que as matrizes do sistema a ser controlado atendam a um conjunto de condições expressas de (3.24) a (3.31); assim, para o caso particular de um sistema com apenas um grau de liberdade, dado pela Figura 3.10 e repetida a seguir pela Figura 3.21 para facilitar a consulta, cujas matrizes são $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$ e $\mathbf{C} \in \mathbb{R}^{m \times n}$, respectivamente matrizes de estado do sistema, matriz de controle e matriz de saídas, que atendem às condições acima referidas (SKGESTAD, 2005; ZHOU; GLOVER, 1995), tem-se a planta generalizada mostrada pela Figura 3.22, a seguir:

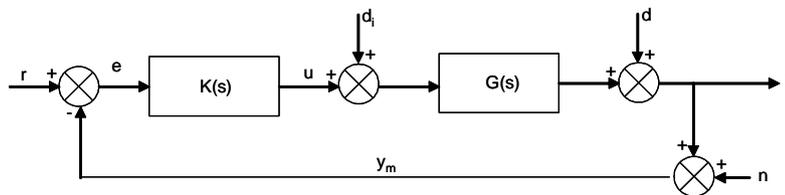


Figura 3.21 - Sistema com um grau de liberdade

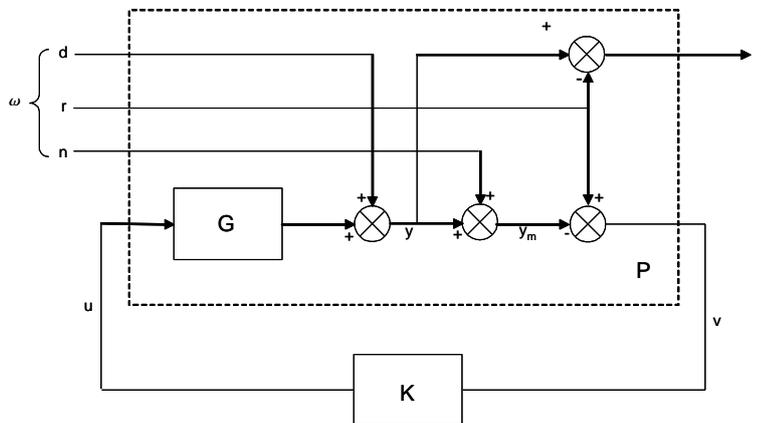


Figura 3.22 - Representação generalizada do sistema mostrado na Figura 3.21

A matriz generalizada \mathbf{P} é determinada, inicialmente, identificando-se na Figura 3.22 os sinais presentes, que são:

$$\omega = [d \ r \ n]^T = [\omega_1 \ \omega_2 \ \omega_3]; \quad z = e = y - r; \quad v = r - y_m = r - y - n \quad (3.41)$$

em que \mathbf{d} , \mathbf{r} , \mathbf{n} são respectivamente o distúrbio, sinal de referência e ruído de medidas; \mathbf{z} e \mathbf{y} são sinais de saídas. O diagrama da Figura 3.22 produz as expressões a seguir:

$$\begin{aligned} z = y - r &= Gu + d - r = I\omega_1 - I\omega_2 + 0\omega_3 + Gu \\ v = r - y_m &= r - Gu - d - n = -I\omega_1 + I\omega_2 - I\omega_3 - Gu \end{aligned} \quad (3.42)$$

\mathbf{P} representa a função de transferência matricial de $[\omega \ u]^T$ para $[z \ v]^T$ é dado pela matriz a seguir:

$$P = \begin{bmatrix} I & -I & 0 & G \\ -I & I & -I & -G \end{bmatrix} \quad (3.43)$$

Deve ser notado que \mathbf{P} não contém o controlador e pode ser obtido por inspeção da Figura 3.22; a obtenção desta matriz por inspeção é tediosa e, quando desenvolvendo cálculos numéricos, pode-se usar um programa para sua geração. Skogestad usa o *Sysic* que é um programa pacote "Robust Control toolbox do Matilab[®]".

Para sintetizar controlador H_{INFINITO} na modalidade de sensibilidade mista através do Matlab, utiliza-se o programa denominado de *Mixsyn* também contido no pacote suprarreferido; este programa será usado no exemplo da próxima seção.

3.3.11 Exemplo de controle H_{∞} de um sistema rígido flexível

Para complementar esta revisão resumida do controlador H_{∞} , apresenta-se nesta seção a aplicação desta técnica, usando o programa *Mixsyn* existente no Matlab para controlar o sistema rígido-flexível modelado na seção (2.2.2); será usada a matriz de estado e de controle do sistema dado por (2.48), que, como visto anteriormente, foi obtida pela substituição dos valores dados pela Tabela 1 do Anexo B na expressão (2.47). Estas matrizes atendem às condições dadas por (3.24) a (3.31).

Para ter acesso a todas as saídas, que são o ângulo de atitude do cubo θ , e deflexão elástica α da haste, que serão controladas, usaremos uma matriz de saída $\mathbf{C}_{2 \times 4}$ e uma matriz de transferência direta $\mathbf{D}_{2 \times 2}$ nula. O sistema, agora na forma numérica, é mostrado em (3.44) e (3.45), dados a seguir:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 591.9697 & -31.9744 & 0 \\ 0 & -947.2755 & 31.9744 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 56.2361 & 0 \\ -56.2361 & 0 \end{bmatrix} \quad (3.44)$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad \mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (3.45)$$

em que $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{m \times n}$, $\mathbf{C} \in \mathbb{R}^{n \times m}$, e $\mathbf{D} = 0$, são respectivamente matrizes de estado, controle, saída e transferência direta, neste caso nula. As condições iniciais usadas são dadas por:

$$x(0) = [0 \ 0.05 \ 0.1 \ 0.2]. \quad (3.46)$$

Considerando que a frequência de ressonância do corpo rígido-flexível em estudo, (3.44), está em torno de 26 rad/s, conforme mostra a Figura 3.23, para projetarmos as funções de ponderação W_S e W_C escolhemos o limite inferior $\omega_b = 1$ rad/s para W_S e o limite superior $\omega_{bc} = 3$ rad/s para W_C . Adotamos uma sensibilidade de pico, tanto para W_S quanto para W_C de 2 (dois), ou seja, $M_S = M_C = 2$ (SKOGESTAD, 2007).

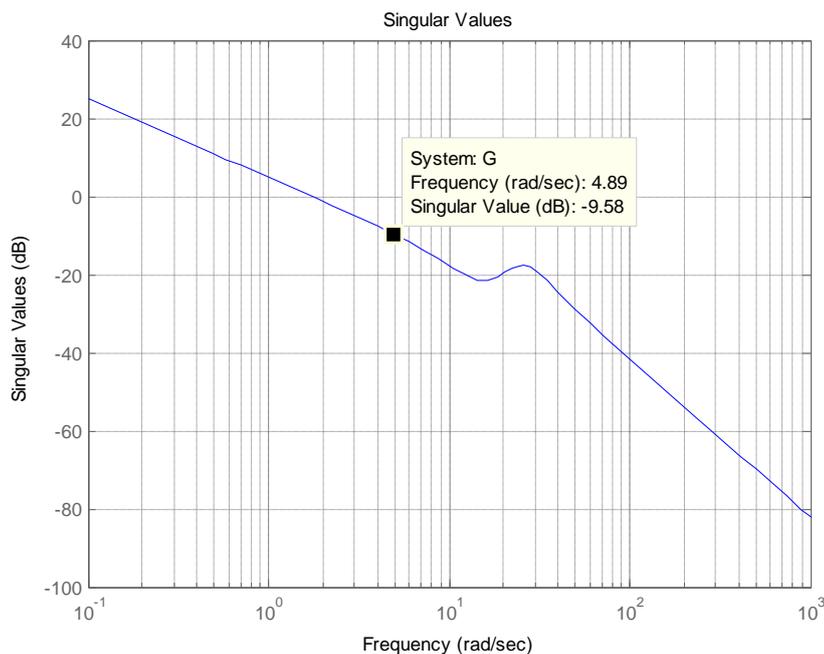


Figura 3.23 - Frequência de ressonância do sistema rígido-flexível

Para obter a função de ponderação $W_S(s)$ relativa à função sensibilidade S usou-se a expressão (3.37), que para facilitar a consulta é repetida a seguir:

$$W_S(s) = \frac{\frac{s}{M_s} + \omega_b}{s + \delta_s \omega_b} \quad (3.47)$$

Para a (3.47) adotou-se os parâmetros abaixo:

$\omega_b = 1$ rad/s: Frequência inferior inicial da faixa do sistema rígido-flexível;

$\delta_S = 0,02$: Erro de regime permanente (ZHOU; DOYLE, 1995);

$M_S = 2,0$: Sensibilidade de pico de W_S .

Para obter-se a função de ponderação de controle $W_C(s)$, usou-se a expressão (3.39), repetida a seguir (ZHOU; DOYLE, 1995):

$$W_C(s) = \frac{s + \frac{\omega_{bc}}{M_C}}{\delta s + \omega_{bc}} \quad (3.48)$$

para a qual adotou-se os parâmetros abaixo:

$\omega_{bs} = 3$ rad/s: Frequência superior inicial da faixa do sistema rígido-flexível;

$\delta = 0,05$: Erro de regime permanente e

$M_C = 2,0$; Sensibilidade de pico de W_C .

Com os filtros acima e usando a função *Mixsyn* do Matlab, foi projetado um controlador robusto H_{INFINITO} cujo programa encontra-se no Anexo A1. A realização numérica deste exemplo foi obtida usando o diagrama Simulink, dado pela Figura 3.24, a seguir:

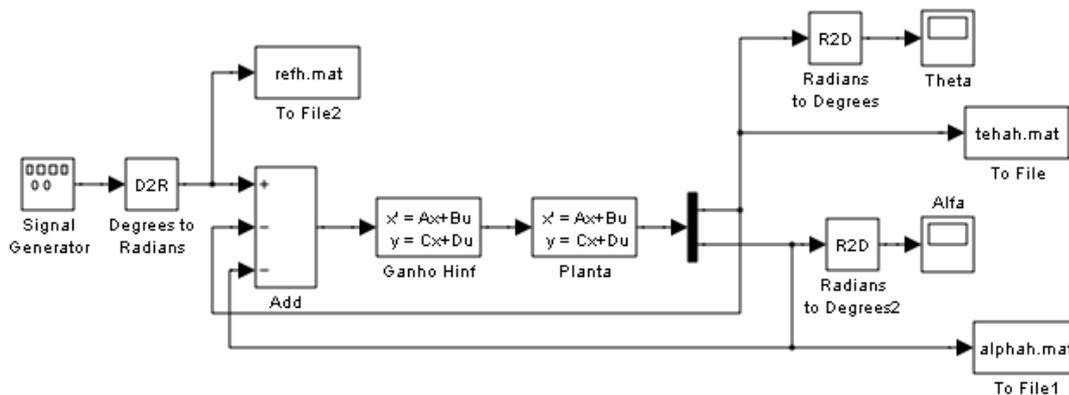
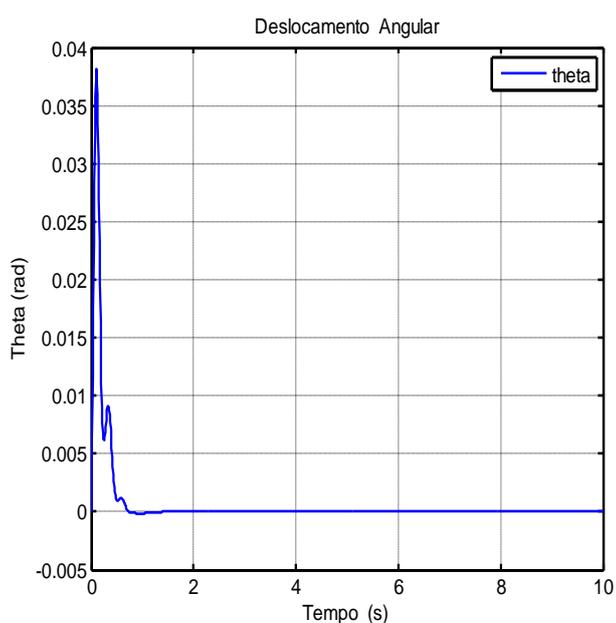


Figura 3.24 - Diagrama Simulink do Controlador H_{INFINITO}

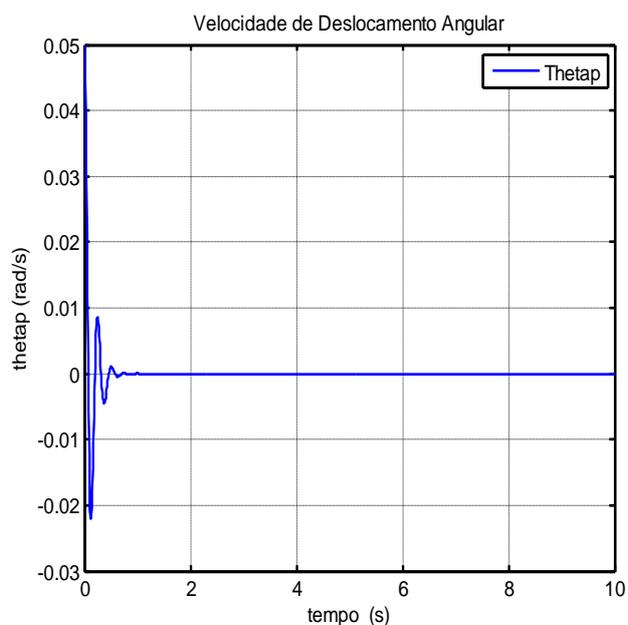
No diagrama Simulink da Figura 3.24 acima, no bloco *Ganho* H_{INF} está contido os resultados do controlador $H_{INFINITO}$ obtido, usando a função *Mixsyn* do Matlab, dado no Apêndice A2. No bloco *Planta* estão contidas as matrizes da planta (3.44) e (3.45).

Os resultados deste controlador serão usados também para comparar com os resultados a serem obtidos pela versão do controlador $H_{INFINITO}$ auxiliado por algoritmos genético.

As Figuras 3.25a, 3.25b, 3.26a e 3.26b apresentam as curvas respectivamente para deslocamento angular, velocidade de deslocamento angular, deslocamento elástico e velocidade de deslocamento elástico do sistema rígido-flexível.

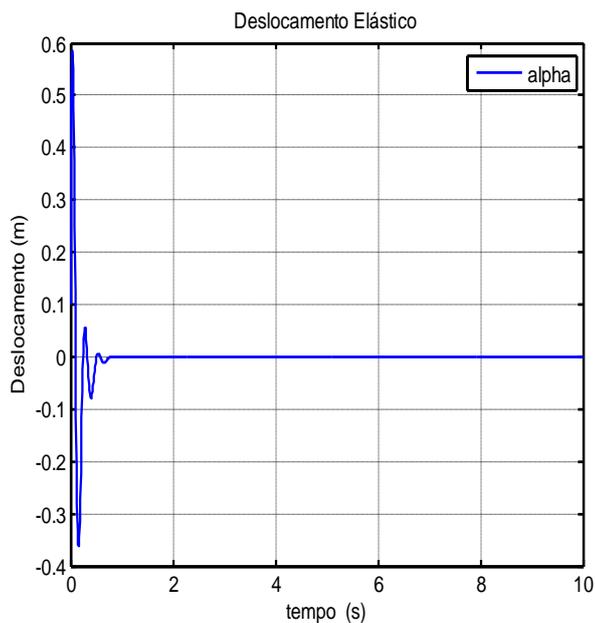


(a)

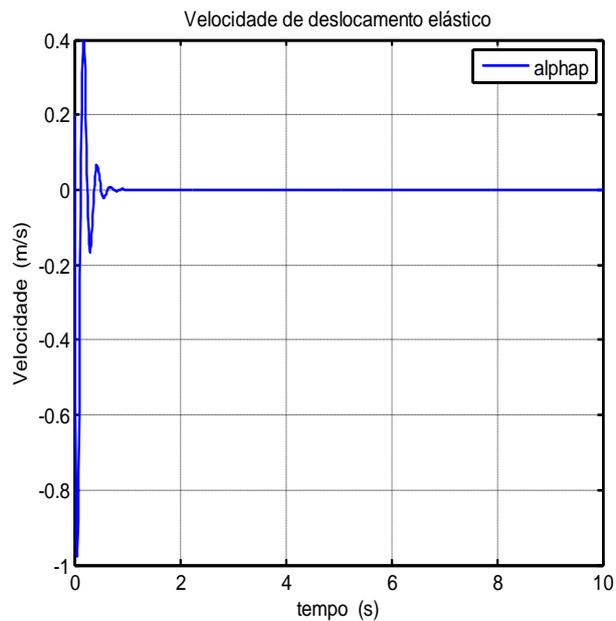


(b)

Figura 3.25 - a) Deslocamento angular e b) Velocidade de deslocamento angular



(a)



(b)

Figura 3.26 - a) Deslocamento elástico e b) Velocidade de deslocamento elástico

3.3.12 Desempenho do *Mixsyn*

As curvas de controle do sistema rígido-flexível foram apresentadas na seção 3.3.12; esta seção apresenta as curvas das funções de ponderação e desempenho do exemplo que estamos analisando, através da função *Mixsyn* do Matlab®. A Figura 3.27 mostra

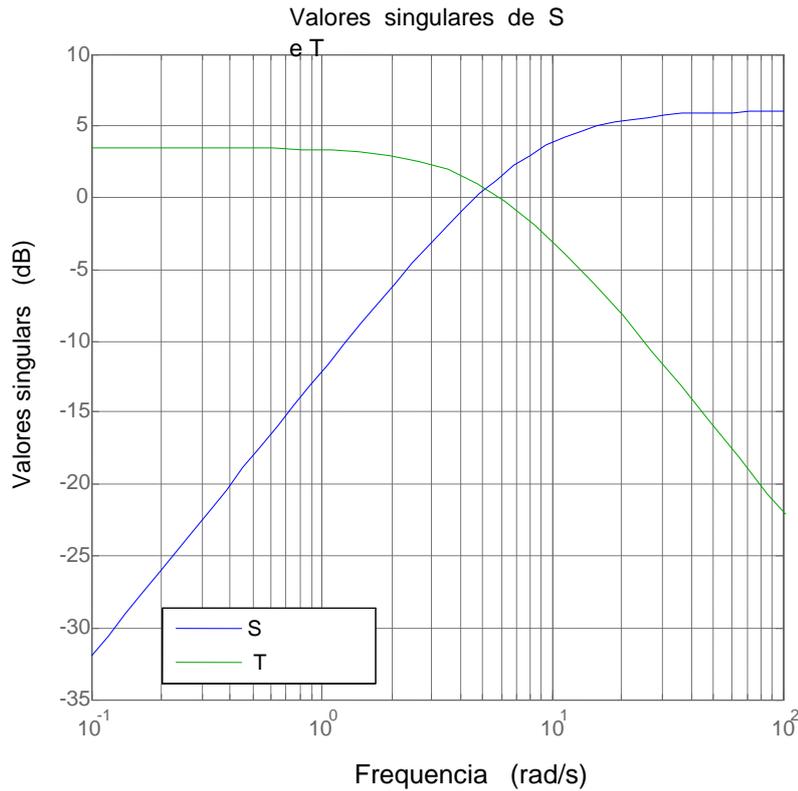


Figura 3.27 - Valores singulares de **S** e **T**

as curvas de sensibilidade e robustez **S** e **T** respectivamente; por esta Figura vê-se que a banda passante é de aproximadamente 8 radianos, com o início da faixa em $\omega_s \approx 3\text{rad/s}$ e fim da faixa em $\omega_b \approx 11\text{rad/s}$.

A Figura 3.28 mostra o gráfico da função sensibilidade **S** e do inverso da função de ponderação W_s . De acordo com Skogestad e Postlethwaite (2007) e Zhou e Doyle (1995), para que haja robustez os valores singulares máximos da função sensibilidade **S** e da função sensibilidade complementar **T** devem atender às relações (3.49) e (3.50) respectivamente, dadas a seguir:

$$\bar{\sigma}(S(j\omega)) < \frac{1}{|W_s(j\omega)|} \quad (3.49)$$

$$\bar{\sigma}(T(j\omega)) < \frac{1}{|W_T(j\omega)|} \quad (3.50)$$

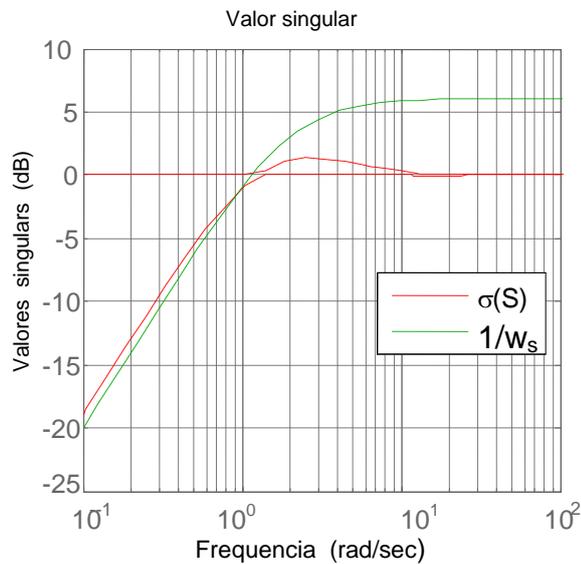


Figura 3.28 - Curvas de $\sigma(\mathbf{S})$ e $1/|W_s|$

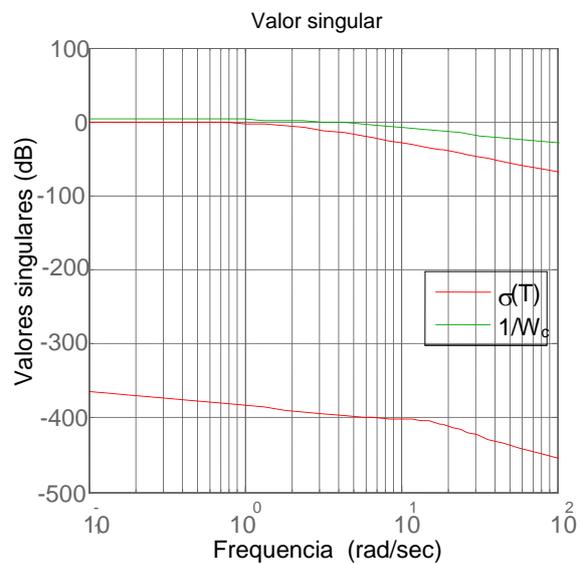


Figura 3.29 - Curvas de $\sigma(\mathbf{T})$ e $1/|W_c|$

Analisando-se as curvas da Figura 3.28, observa-se que a relação (3.49) é atendida a partir da frequência de 0,95 rad/s. Na Figura 3.29, observa-se que a relação (3.50) é praticamente atendida em toda a faixa escolhida que vai de 1 a 4 rad/s.

3.4 Fundamentos de um controlador SDRE

O controle de sistemas lineares, atualmente está bem desenvolvido, apresentando várias técnicas eficientes para controle e estabilização de sistemas; afirmativa similar não pode ser feita em relação ao controle não linear cuja metodologia ainda não está completamente dominada, pelo menos para os casos mais gerais. Várias técnicas existem que funcionam razoavelmente bem para uma faixa muito estreita de sistemas simples; estas técnicas linearizam o sistema a ser controlado em um determinado ponto, considerado o ponto de trabalho, e aplicam no entorno deste ponto uma técnica de controle linear. Para controlar sistemas não lineares de forma mais eficiente e natural, é

necessário um controle não linear que melhor se adapte ao sistema e às limitações computacionais.

Atualmente, existem várias técnicas de controle não lineares tais como: linearização usando o jacobiano do sistema a ser controlado, linearização por realimentação e controle adaptativo. As técnicas que fazem uso do jacobiano do sistema e linearização por realimentação são eficientes apenas para um pequeno conjunto de sistemas, visto que estas técnicas consideram que o sistema permaneça na região linearizada durante toda a evolução do processo de controle, o que nem sempre acontece na prática (KATSEV, 2006).

Controle adaptativo é uma boa técnica para controle linear, mas não está suficientemente desenvolvida para controle não linear e, além disso, seu custo computacional é elevado. Os principais problemas dos controles não lineares atuais são:

- Aplicáveis a um conjunto limitado de sistemas;
- Em muitos casos consideram o sistema a ser controlado quase linear e
- E apresentam elevada carga computacional.

Uma técnica mais recente, aplicada a sistemas de controle não lineares, denominada controle por Equação de Riccati Dependente do Tempo (SDRE) evita os problemas acima; o controlador SDRE usa procedimentos simples, apresenta baixa carga computacional e é aplicável a uma larga faixa de sistemas não lineares. O controlador SDRE é, basicamente, uma generalização do controlador LQR, tornando o LQR dependente do estado e, portanto, tornando-o um controlador não linear; por esta razão o controlador SDRE herda do LQR a problemática da determinação das matrizes de ponderação de estado \mathbf{Q} e custo de controle \mathbf{R} que são determinadas por tentativa e erro.

Faz parte dos objetivos deste trabalho, usando o Simulink[®], determinar, de forma automática, as matrizes de ponderação \mathbf{Q} e \mathbf{R} . No caso mais geral, as matrizes de ponderação de estado e de custo de controle podem ser, também, dependentes do estado, fato que confere ao controlador maior grau de

liberdade. Neste trabalho estamos considerando estas matrizes como matrizes constantes.

3.4.1 Controlador SDRE

O problema de um controlador não linear usando a metodologia SDRE é formulado através de uma função custo, conforme Cloutier e Mracek, (1996) e Souza e Gonzales (2012), por:

$$J(x_0, u) = \frac{1}{2} \int_{t_0}^{\infty} (x^T Q(x)x + u^T R(x)u) dt, \quad (3.51)$$

em que $\bar{x} = [x_1, \dots, x_n]^T \in \mathbb{R}^n$ é o vetor de variáveis de estado, $u = [u_1, \dots, u_m]^T \in \mathbb{R}^m$ é o vetor de entradas, $\mathbf{Q} \in \mathbb{R}^{n \times n}$ é uma matriz simétrica, positiva semidefinida e $\mathbf{R} \in \mathbb{R}^{m \times m}$ é uma matriz simétrica positiva definida sujeita à restrição não linear do sistema dada pelo sistema:

$$\begin{aligned} \dot{x} &= f(x) + B(x)u \\ y &= C(x) \\ x(0) &= x_0 \end{aligned} \quad (3.52)$$

em que $\mathbf{B} \in \mathbb{R}^{n \times m}$ é a matriz de controle, $\mathbf{C} \in \mathbb{R}^{r \times n}$ é a matriz de saída, e \mathbf{y} é o vetor de saída do sistema, $y = [y_1, \dots, y_r]^T$. O vetor de condições iniciais é dado por $x_0 = [x_{01}, \dots, x_{0n}]^T$.

Aplicando-se uma adequada parametrização para transformar o sistema não linear em uma representação de Coeficientes Dependentes do Estado (SDC), a equação que expressa a dinâmica do sistema e o seu controle pode ser escrita como;

$$\dot{x} = A(x)x + B(x)u \quad (3.53)$$

em que $f(x) = A(x)x$, sendo $A \in \mathbb{R}^{n \times n}$ a matriz de estado que expressa a dinâmica do sistema e $g(x) = B(x)u$, sendo $B \in \mathbb{R}^{n \times m}$ a matriz de controle, ambas na forma SDC.

É sabido que para o caso multivariável, existe um número infinito de possibilidades para a conversão do sistema não linear para a forma SDC, ou seja, $A(x)$ e $B(x)$ não são únicas. Para que estas matrizes sejam parametrizações válidas para compor um sistema de controlador SDRE, devem ser verificadas as condições a seguir:

- $A(x)$ deve ser uma parametrização observável (detectável), ou seja, o par $[C(x), A(x)]$ deve ser observável (detectável) no sentido do controle linear;
- $A(x)$ deve ser uma parametrização controlável (estabilizável), ou seja, o par $[A(x), B(x)]$ deve ser controlável (estabilizável) no sentido do controle linear.

3.4.2 Obtenção do controlador SDRE

A equação algébrica de Riccati dependente do estado (SDRE) pode ser obtida pela aplicação das condições de optimalidade do cálculo variacional (CLOUTIER, 1997; ARENA; SOUZA, 2013; MENON et al, 2002). Assim, o hamiltoniano para o problema de controle ótimo do sistema dado pelas Equações (3.51) e (3.52) é dado por:

$$H(x, u, \lambda) = \frac{1}{2}(x^T Q(x)x + u^T R(x)u) + \lambda^T (A(x)x + B(x)u) \quad (3.54)$$

em que $\lambda \in \mathbb{R}^n$ é o multiplicador de Lagrange.

Aplicando sobre (3.54) as condições de optimalidade dadas pelas expressões:

$$\dot{x} = \frac{\partial H}{\partial \lambda}, \quad \frac{\partial H}{\partial u} = 0 \quad \text{e} \quad \dot{\lambda} = \frac{\partial H}{\partial x} \quad (3.55)$$

obtem-se:

$$\dot{\lambda} = -Q(x)x - \frac{1}{2}x^T \frac{\partial Q(x)}{\partial x} x - \frac{1}{2}u^T \frac{\partial R(x)}{\partial x} - \left[\frac{\partial(A(x)x)}{\partial x} \right]^T \lambda - \left[\frac{\partial(B(x)u)}{\partial x} \right]^T \quad (3.56)$$

$$\dot{x} = A(x)x + B(x)u \quad (3.57)$$

$$0 = R(x)u + B(x)\lambda \quad (3.58)$$

Admitindo o co-estado da forma $\lambda=P(x)x$, que é dependente do estado, da Equação. (3.58) obtém-se a lei de retro alimentação a seguir:

$$u = -R^{-1}(x)B^T(x)P(x)x \quad (3.59)$$

Substituindo (3.59) em (3.57), obtém-se;

$$\dot{x} = A(x)x - B(x)R^{-1}(x)B^T(x)P(x)x \quad (3.60)$$

$P(x)$ é obtido derivando-se $\lambda=P(x)$ em relação ao tempo , obtendo-se:

$$\dot{\lambda} = \dot{P}(x)x + P(x)\dot{x} = \dot{P}(x)x + P(x)A(x)x - P(x)B(x)R^{-1}(x)B^T(x)P(x)x \quad (3.61)$$

Substituindo a Equação.(3.61) na Equação.(3.56), tem-se:

$$\begin{aligned} \dot{P}(x)x + P(x)A(x)x - P(x)B(x)R^{-1}(x)B^T(x)P(x)x = \\ = -Q(x)x - \frac{1}{2}x^T \frac{\partial Q(x)}{\partial x} x - \frac{1}{2}u^T \frac{\partial R(x)}{\partial x} u - \left[\dot{A}(x) + \frac{\partial A(x)x}{\partial x} \right]^T P(x)x - \left[\frac{\partial(B(x)u)}{\partial x} \right]^T P(x)x \end{aligned} \quad (3.62)$$

Rearranjando os termos da expressão acima, obtém-se:

$$\begin{aligned} \dot{P}(x)x + \frac{1}{2}x^T \frac{\partial Q(x)}{\partial x} x + \frac{1}{2}u^T \frac{\partial R(x)}{\partial x} u + x^T \left[\frac{\partial(A(x))}{\partial x} \right]^T P(x)x + \left[\frac{\partial(B(x)u)}{\partial x} \right]^T P(x)x + \\ + \left[P(x)A(x) + A^T(x)P(x) - P(x)B(x)R^{-1}(x)B^T(x)P(x) + Q(x) = 0 \right] \end{aligned} \quad (3.63)$$

Com objetivo de satisfazer a igualdade da Equação (3.63), obtém-se duas relações importantes, dadas a seguir: a primeira relação é a equação de Riccati dependente do estado (SDRE), cuja solução é $P(x)$, dada por:

$$P(x)A(x) + A^T(x)P(x) - P(x)B(x)R^{-1}(x)B^T(x)P(x) + Q(x) = 0 \quad (3.64)$$

a segunda relação é a condição necessária de optimalidade a ser satisfeita e é dada por:

$$\dot{P}(x)x + \frac{1}{2}x^T \frac{\partial Q(x)}{\partial x} x + \frac{1}{2}u^T \frac{\partial R(x)}{\partial x} u + x^T \left[\frac{\partial(A(x))}{\partial x} \right]^T P(x)x + \left[\frac{\partial(B(x)u)}{\partial x} \right]^T P(x)x = 0 \quad (3.65)$$

Considerando um regulador linear quadrático padrão (LQR), no qual tempo de integração é ilimitado (integral de $T=0$ a $T=\infty$), esta é uma condição que satisfaz as condições de optimalidade de um controle sub-ótimo.

Finalmente, a lei de controle não linear retroalimentada pelos estados é dada a seguir:

$$u = -R^{-1}(x)B^T(x)P(x) \quad (3.66)$$

Para sistemas com poucas variáveis e com pequena dependência dos estados, a Equação (3.64) pode ser resolvida analiticamente; para sistemas mais complexos e com forte dependência dos estados a solução é obtida numericamente por computador, usando uma adequada taxa de amostragem.

Para a aplicação da metodologia SDRE, é necessário que o par formado pelas matrizes $A(x)$ e $B(x)$ sejam controláveis de acordo o critério de controlabilidade de um sistema linear do controle (SKOGESTAD, 2007; CLOUTIER, 1997), conforme detalhado no final da Seção 3.5.2.

Uma propriedade importante do método SDRE é que ele não cancela os benefícios provenientes da não linearidade, pelo fato de não exigir inversões e nem linearização do sistema a ser controlado.

3.5 Conceitos básicos de alocação de pólos

Alocação de pólos é uma técnica que consiste em deslocar os pólos em malha fechada do sistema a ser controlado para locais convenientes do plano complexo, visando obter o controle do sistema. Na literatura de controle estão disponíveis vários processos para obter-se alocação de pólos; para alocação de pólos de sistemas com múltiplas entradas e múltiplas saídas, que é o caso

deste trabalho, é muito empregada a metodologia das matrizes companheiras, cujos conceitos básicos são apresentados na próxima seção.

3.5.1 Alocação de pólos usando o método das matrizes companheiras

Dado o sistema a seguir:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \quad (3.67)$$

em que $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{m \times n}$ e u são respectivamente matriz de estado, matriz de controle, matriz de saída e vetor de controle do sistema; sendo o sistema controlável, mediante algumas operações com as matrizes, sempre se pode colocar estas matrizes na forma:

$$A = \begin{bmatrix} 0 & 1 & \dots & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & \vdots & \vdots & \dots & 1 \\ -a_1 & -a_2 & \dots & \dots & -a_n \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ 1 \end{bmatrix}, \quad C = [c_1 \quad c_2 \quad \dots \quad c_n] \quad (3.68)$$

Os autovalores de malha aberta são os autovalores da matriz companheira A dada em (3.68). Da realização deste sistema por equações diferenciais temporal, conforme a expressão:

$$y^{(n)} + a_n y^{(n-1)} + \dots + a_2 \dot{y} + a_1 y = C_m^{(m-1)} u + \dots + C_2 \dot{u} + C_1 u \quad (\text{VEGTE, 1994}),$$

têm-se que os autovalores são também raízes da equação característica dada por:

$$s^n + a_n s^{n-1} + \dots + a_2 s + a_1 = 0 \quad (3.69)$$

A expressão (3.69) tem os mesmos coeficientes que a última linha da matriz **A** em (3.68), com seus sinais trocados e ordem invertida. Supondo que todos os estados estejam disponíveis para retro alimentação e que a retro alimentação de estado seja dado por:

$$u = -K'x = -[k_1 \quad k_2 \quad \dots \quad k_n]x \quad (3.70)$$

onde a plica indica a transposta da matriz coluna **K**, a matriz de malha fechada do sistema $A-BK'$ é novamente uma matriz companheira dada por:

$$A = \begin{bmatrix} 0 & 1 & \dots & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & 1 \\ -a_1 & -a_2 & \dots & \dots & -a_n \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \dots \\ \dots \\ \dots \\ \dots \\ \dots \\ 1 \end{bmatrix} \quad (3.71)$$

Então por analogia de (3.71) com (3.70), os autovalores de malha fechada devem ser as raízes da equação característica de malha fechada dada por:

$$s^n + (a_n + k_n)s^{n-1} + \dots + (a_2 + k_2)s + (a_1 + k_1) = 0 \quad (3.72)$$

Fazendo os autovalores de malha fechada desejados iguais a $\hat{\lambda}_1 \dots \hat{\lambda}_n$, então a equação característica será dada por:

$$(s - \hat{\lambda}_1)(s - \hat{\lambda}_2) \dots (s - \hat{\lambda}_n) = s^n + a_{dn}s^{n-1} + \dots + a_{d2} + a_{d1} = 0 \quad (3.73)$$

A expressão (3.72) será igual **K'** dada pela expressão (3.73) dada a seguir:

$$K' = [a_{d1} - a_1 \quad a_{d2} - a_2 \quad \dots \quad a_{dn} - a_n] \quad (3.74)$$

Se o modelo de estado já estiver na forma de matriz companheira, esta metodologia é a mais simples para se obter alocação de pólos (VEGTE, 1994).

A eficiência da técnica de alocação de pólos é comprovada pelos resultados do controle obtidos, contudo uma visualização gráfica direta de seus resultados, para o presente trabalho, é obtida para determinação dos pólos e zeros, sobre a malha fechada do sistema rígido

(simulador de satélite) em estudo, cujas matrizes são dadas por (4.2), antes e após o processo de alocação dos pólos, conforme mostrado pelas Figuras 3.30a,b.

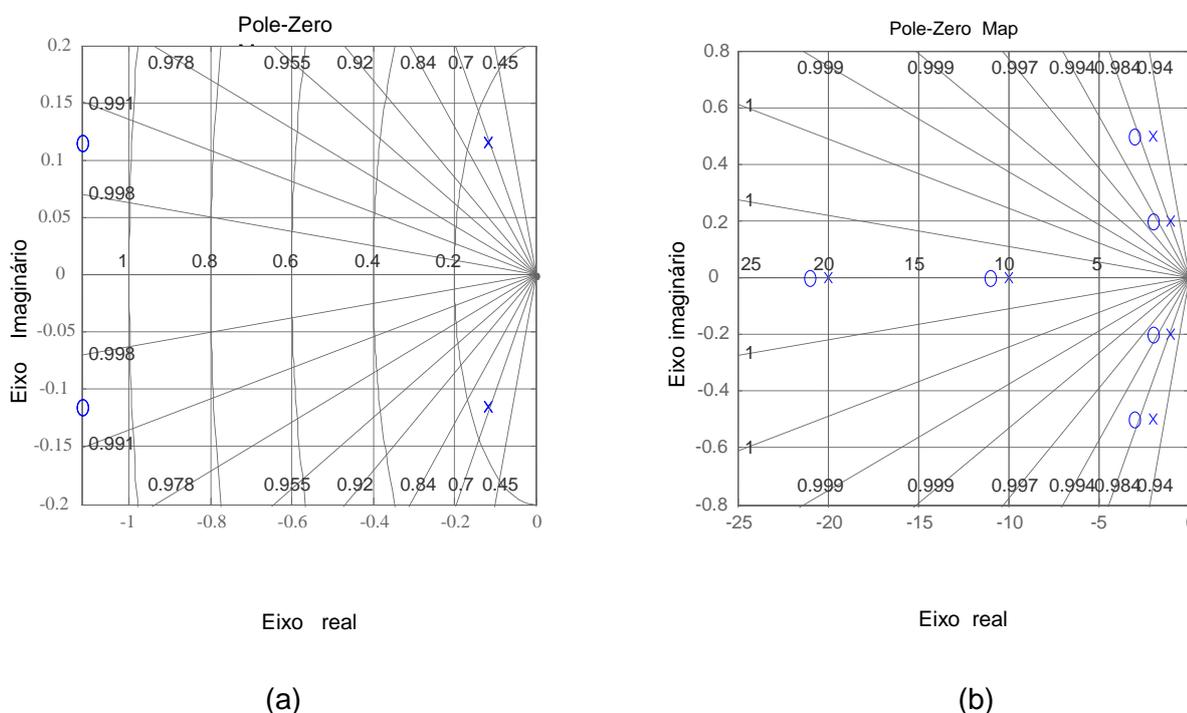


Figura 3.30a,b - Mapas dos pólos do sistema: a) antes e b) após a alocação de pólos

A figura 3.30a apresenta as posições dos pólos do simulador de satélite em malha fechada original conforme listado abaixo:

- 0,1171 + 0,1155i
- 0,1171 - 0,1155i
- 0,1171 + 0,1155i

$$-0,1171 - 0,1155i$$

$$-0,1171 + 0,1155i$$

$$-0,1171 - 0,1155i$$

Apos alocação de pólos, conforme a matriz: $p=[-1+0,2i \ -1-0,2i \ -2+0,5i \ -2-0,5i \ -10 \ -20]$, os pólos foram para as posições dadas abaixo, aumentando a estabilidade relativa do simulador, conforme mostra a Figura 3.30b.

$$-1,0000 + 0,2000i$$

$$-1,0000 - 0,2000i$$

$$-2,0000 + 0,5000i$$

$$-2,0000 - 0,5000i$$

$$-10,0000$$

$$-20,0000$$

4 CONTROLE E SIMULAÇÃO

Neste capítulo foram utilizadas as técnicas de controle revistas no Capítulo 3 para controlar os modelos apresentados no Capítulo 2 e realizar comparações entre os resultados de controladores usando técnicas convencionais e os resultados destes mesmos controladores auxiliados por um controlador *Fuzzy*.

Inicialmente trabalhou-se com o modelo do simulador de satélite linearizado visto que os controladores empregados destinam-se a controle de sistemas lineares.

A seguir, buscando realizar o segundo objetivo deste trabalho, utilizou-se o modelo do referido simulador porém na sua versão não linear original, considerando que neste caso será usado um controlador SDRE convencional, cujos resultados serão comparados com os resultados de um controlador SDRE auxiliado por controlador *Fuzzy*, para a determinação das matrizes de ponderação constante \mathbf{Q} . A matriz \mathbf{R} é considerada unitária.

Finalmente, na terceira e última etapa deste trabalho, utilizou-se o modelo linearizado de um sistema rígido-flexível, sobre o qual será aplicado um controlador H_{∞} , no qual as matrizes de ponderação serão determinadas por algoritmos genético.

4.1 Comparação dos controladores por alocação de pólos e LQR/*Fuzzy*

Como dito no Capítulo 2, o modelo do simulador tratado na seção 2.1.1 originalmente é não linear, contudo, adotando-se as hipóteses de pequenas perturbações como: ângulos de atitudes ϕ , θ e ψ pequenos (ângulos com valores próximos de zero, dependendo dos objetivos da missão) e admitindo-se $r_{cg}=0$, ou seja, o centro de massa da plataforma coincidente com a origem do sistema de coordenada da plataforma, conforme dito no início da seção 2.1.2, obtém-se uma versão linear do modelo do simulador (DORUK, 2008).

Na seção 2.1.2, foi obtida a versão linearizada do simulador, expressão 2.36, que se reinterpreta em 4.1 a seguir para facilidade de consulta.

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \\ \phi \\ \theta \\ \psi \end{bmatrix} + \begin{bmatrix} \frac{1}{I_1 - I_{xx}} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{I_2 - I_{yy}} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{I_3 - I_{zz}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad (4.1)$$

Em 4.1, $\vec{x} = [\phi \ \theta \ \psi \ p \ q \ r]^T$ e $\vec{u} = [T_1 \ T_2 \ T_3]^T$ onde \mathbf{x} e \mathbf{u} são respectivamente as variáveis a serem controladas e os torques fornecidos pelas rodas de reação; estes torques são gerados pelas rotações das respectivas rodas e constituem o controle \mathbf{u} do simulador.

I_1 , I_2 e I_3 são os momentos de inércia das rodas de reação e I_{xx} , I_{yy} e I_{zz} são os momentos de inércia da plataforma do simulador.

Substituindo-se em 4.1, os valores numéricos das inércias das rodas de reação dados por: $I_{xx}=1,1667$, $I_{yy}=1,1671$ e $I_{zz}=2,2129 \text{ Nm}^2$, fornecidos pelo fabricante e dados no Apêndice B e os valores numéricos das inércias da plataforma dados por $I_1=I_2=I_3=1,792 \times 10^{-3} \text{ Nm}^2$, obtêm-se as matrizes numéricas da plataforma dadas por 4.2.

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}}_A \begin{bmatrix} p \\ q \\ r \\ \phi \\ \theta \\ \psi \end{bmatrix} + \underbrace{\begin{bmatrix} -0.8584 & 0 & 0 \\ 0 & -0.8581 & 0 \\ 0 & 0 & -0.4523 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_B \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \quad (4.2)$$

Com as matrizes **A** e **B** que representam a dinâmica da plataforma, dadas por 4.2 (FONSECA, 2011) e usando a ferramenta Simulink do Matlab®, implementou um controle por retro alimentação de estado, $u=-Kx$, em que o ganho **K** foi determinado por alocação de pólos; este controlador está mostrado na Figura 4.1, dado a seguir:

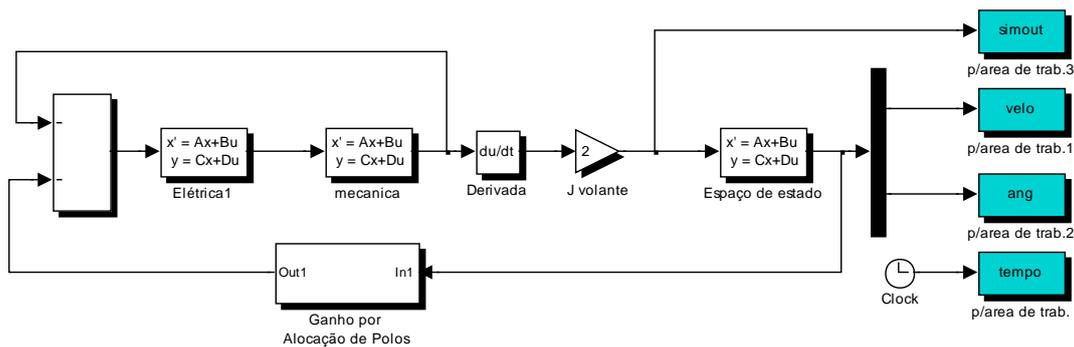


Figura 4.1 - Diagrama Simulink do Controlador por Alocação de Pólos

Para atender o primeiro objetivo deste trabalho, que é a proposta de usar um controlador *Fuzzy* para determinar, de forma automática, a matriz **Q** de um controlador *LQR*, utilizou-se o controlador da Figura 4.1, modificado de forma que o ganho de retro alimentação **K** foi determinado por um controlador *LQR*, cuja matriz de ponderação **Q** foi determinada por um controlador *Fuzzy*; a matriz **R** considerada constante diagonal e unitária. Para desenvolver este controlador *LQR/Fuzzy*, usou-se a função *lqr* para criar a parte *LQR* do

controlador e a função *newfis* para criar a parte referente ao controle *Fuzzy*; ambas são funções do Matlab[®]. O programa completo do controlador LQR/*Fuzzy* encontra-se nos Apêndices A1, A2, A3, A4 e A5. A Figura 4.2 apresenta a versão proposta do controlador da plataforma que, neste trabalho é denominado controlador LQR/*Fuzzy*; o controle *Fuzzy*, na Figura 4.2, está inserido no bloco *Out1-In1* que contém a retro alimentação de estado, cujo ganho foi obtido pelo controlador LQR/ *Fuzzy*.

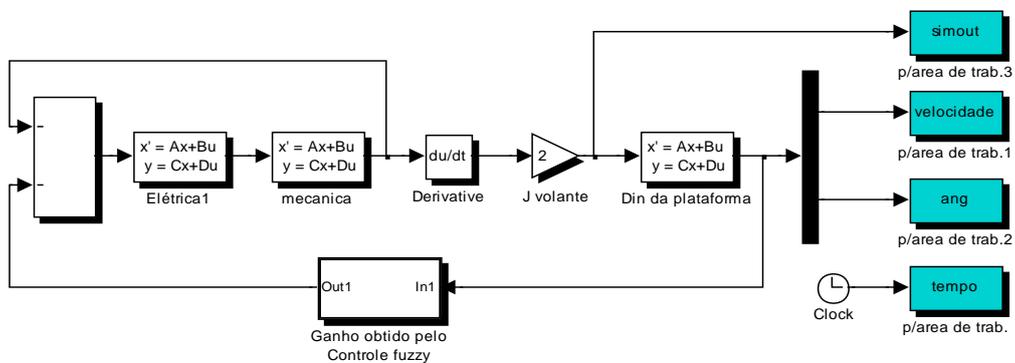
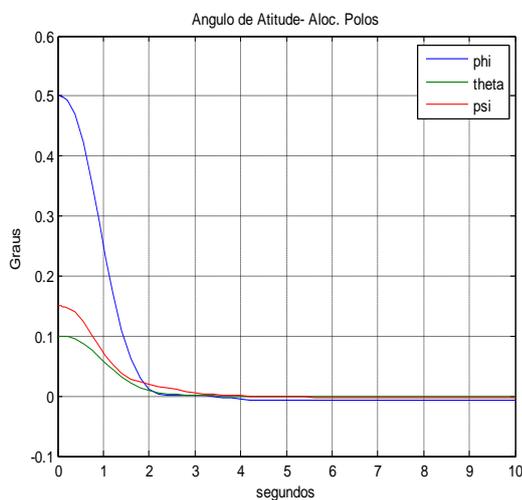


Figura 4.2 - Diagrama Simulink do controlador LQR/*Fuzzy*

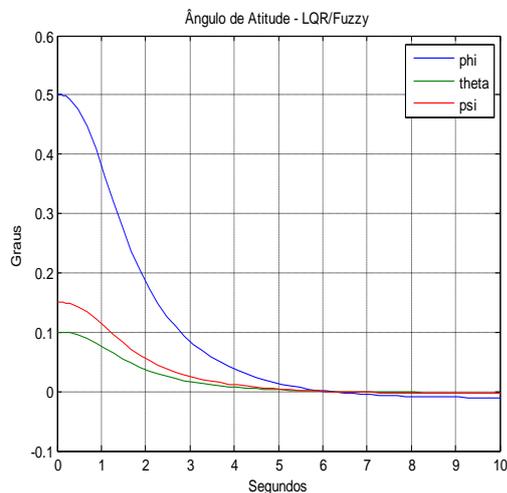
Para projetar um controlador por alocação de pólos para sistemas de múltiplas entradas e múltiplas saídas (MIMO) existem várias técnicas. Neste trabalho foi usada a técnica denominada de matriz companheira, que faz uso das variáveis de fase, cujos conceitos fundamentais foram revisto na seção 3.6; usou-se esta técnica aplicando a função *place* do Matlab para controlar o modelo linearizado do simulador de satélite tratado na seção 2.1.1, cujos resultados são mostrados pelas Figuras 4.3a, 4.4a, 4.5a, 4.6a, 4.7a e 4.8a. Para comparar o desempenho do controlador LQR/*Fuzzy*, que está sendo apresentado neste trabalho, mostrado na Figura 4.2, com o desempenho de outro controlador (neste caso o

controlador por alocação de pólos da Figura 4.1 foi adotado a condição inicial dada por $\phi = 0,5^\circ$, $\theta = 0,1^\circ$ e $\psi = 0,15^\circ$ para os dois controladores.

Para que ocorra o controle, tanto para o controlador por alocação de pólos, como para o controlador LQR/*Fuzzy*, preliminarmente deve ser rodado o programa A1, do Apêndice A, visto que este programa é responsável por passar os valores numéricos dos parâmetros do simulador de satélite e a rotina controlador LQR/*Fuzzy* para o Simulink. Para facilitar a comparação entre os resultados do controlador LQR/*Fuzzy* em relação ao controlador por alocação de pólos, suas curvas de respostas foram agrupadas lado a lado, de acordo com as naturezas das variáveis (ângulo com ângulo, velocidade com velocidade e torque com torque). Inicialmente, para analisar o comportamento transitório dos dois sistemas, foi usado um tempo 10 segundos para respostas de cada variável, usando as condições iniciais dadas acima ($\phi = 0,5^\circ$, $\theta = 0,1^\circ$ e $\psi = 0,15^\circ$). Assim, as Figuras 4.3a,b, 4.4a,b e 4.5a,b mostram as curvas de respostas transitórias de atitude, velocidades do simulador e torques das rodas de reação respectivamente, dos controladores em estudo.

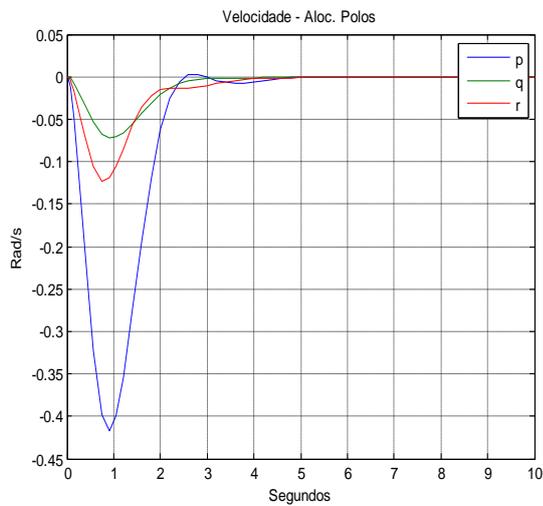


(a)

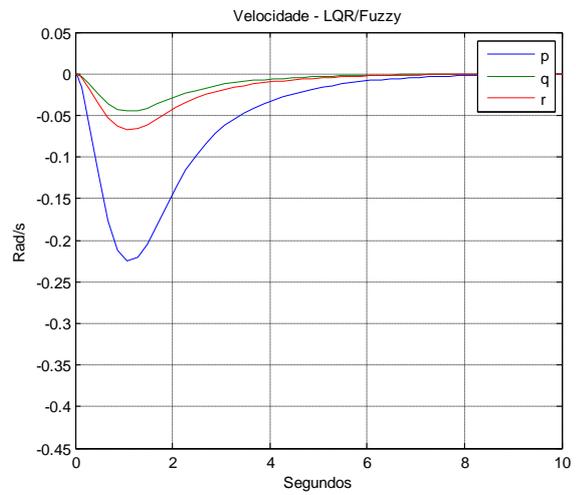


(b)

Figura 4.3a,b - Respostas transitória de atitude - Alloc. de pólos versus LQR/*Fuzzy*

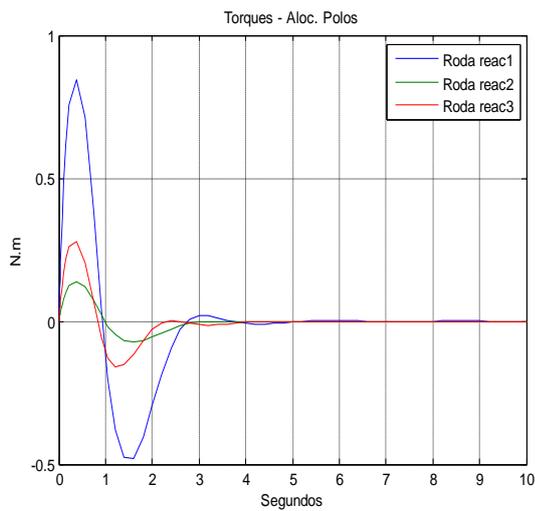


(a)

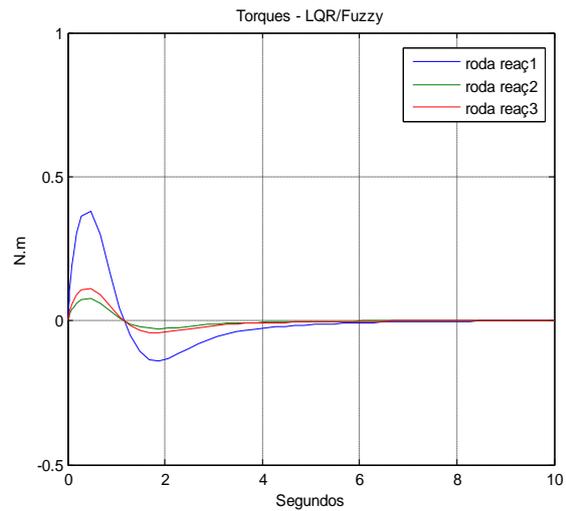


(b)

Figura 4.4a,b - Resposta transitória de velocidade – Alloc. pólos versus LQR/Fuzzy



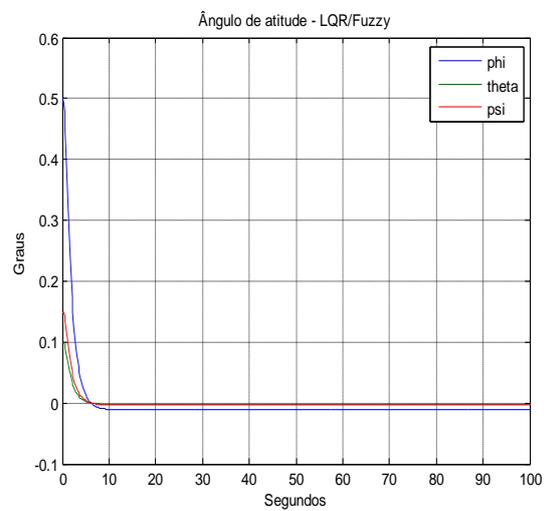
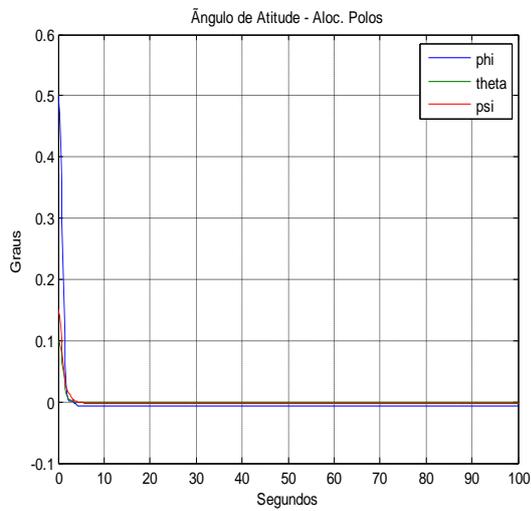
(a)



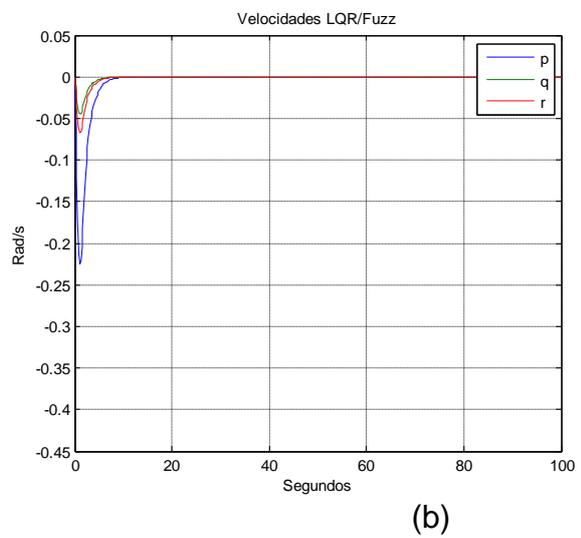
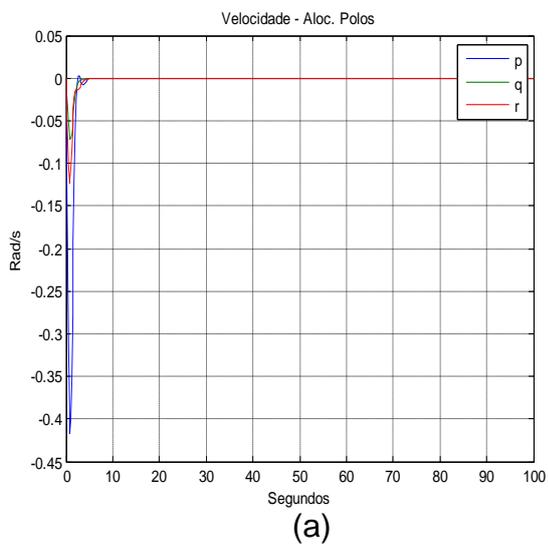
(b)

Figura 4.5a,b - Resposta transitória de torque – Alloc. pólos versus LQR/Fuzzy

As Figuras 4.6a,b, 4.7a,b e 4.8a,b mostram as curvas de respostas de regime permanente da atitude do simulador, velocidades e torques das rodas de reação respectivamente, dos controladores em estudo.



(a) (b)
 Figura 4.6a,b – Resposta de regime permanente de atitude - Alocação de pólos versus LQR/Fuzzy



(a) (b)
 Figura 4.7a,b –Resposta de regime permanente de velocidades- Alocação de pólos versus LQR/Fuzzy

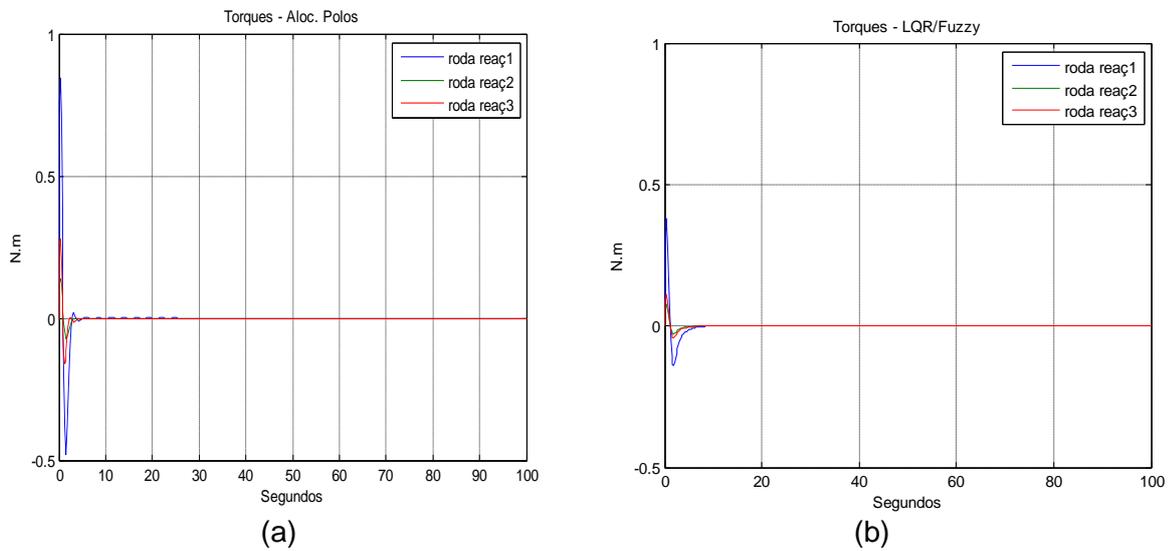


Figura 4.8a,b – Resposta de regime permanente de torque - Alocação pólos versus LQR/Fuzzy

A matriz de ponderação de estado \mathbf{Q} obtida para o controlador acima é da forma diagonal dada por $\mathbf{Q}=\mathbf{Q}*\mathbf{eye}(6)$. A matriz de ponderação de custo de controle, como dito anteriormente, foi considerada escalar $R=0,001$.

4.1.1 Comparação do comportamento das matrizes \mathbf{Q} e \mathbf{R} dos controladores LQR/Fuzzy e SDRE/Fuzzy

Nesta seção é realizado um breve estudo dos comportamentos das matrizes de ponderação \mathbf{Q} e \mathbf{R} de um controlador LQR auxiliado por lógica *Fuzzy*. Foram usadas para este estudo as matrizes do simulador de satélite tratado neste trabalho que são rerepresentadas por 4.3 e 4.4, abaixo. Inicialmente foi usado um controlador LQR clássico, onde as matrizes \mathbf{Q} e \mathbf{R} foram determinadas por tentativas e erro. Tendo em vista que a resposta do controle *Fuzzy* depende do erro \mathbf{e} do sistema e sua variação \mathbf{ep} , para este estudo, sem perda de generalidade, considera-se o erro \mathbf{e} e sua variação \mathbf{ep} constantes.

4.1.2 Matrizes do sistema

$$A = [\text{zeros}(3,6); \\ 1 \ 0 \ 0 \ 0 \ 0 \ 0; \\ 0 \ 1 \ 0 \ 0 \ 0 \ 0; \\ 0 \ 0 \ 1 \ 0 \ 0 \ 0]; \quad (4.3)$$

$$B = [-0,856 \ 0 \ 0; \\ 0 \ -0,856 \ 0; \\ 0 \ 0 \ -0,856; \\ \text{zeros}(3,3)]; \quad (4.4)$$

Em que $A_{6 \times 6}$ e $B_{6 \times 3}$ são respectivamente as matrizes da dinâmica e de controle do simulador de satélite. É considerado que todas as saídas estão disponíveis para controle.

Como este sistema é controlável e observável, para implementar um controlador LQR deste simulador, usou-se a função *lqr* do Matlab, obtendo-se os resultados mostrados na Figura 4.9. Para este controlador, por tentativa e erro, obteve-se uma matriz diagonal de ponderação de estado $Q = \text{eye}(6)$, onde $\text{eye}(6)$ é uma função do Matlab; adotou-se a ponderação de custo de controle $R=1$, para as condições iniciais $x_1=0,5$; $x_2=0,1$; $x_3=0,15$; $x_4=0$; $x_5=0$; $x_6=0$. A Figura 4.10 mostra o consumo de energia deste controlador

4.1.3 Resultados do controlador lqr clássico (matriz q obtida por tentativas)

Matrizes de ponderação $Q = \text{diag}(100,6)$ e $R = \text{diag}(1,6)$

Condições iniciais: $x_1=0,5$; $x_2=0,1$; $x_3=0,15$; $x_4=0$; $x_5=0$; $x_6=0$

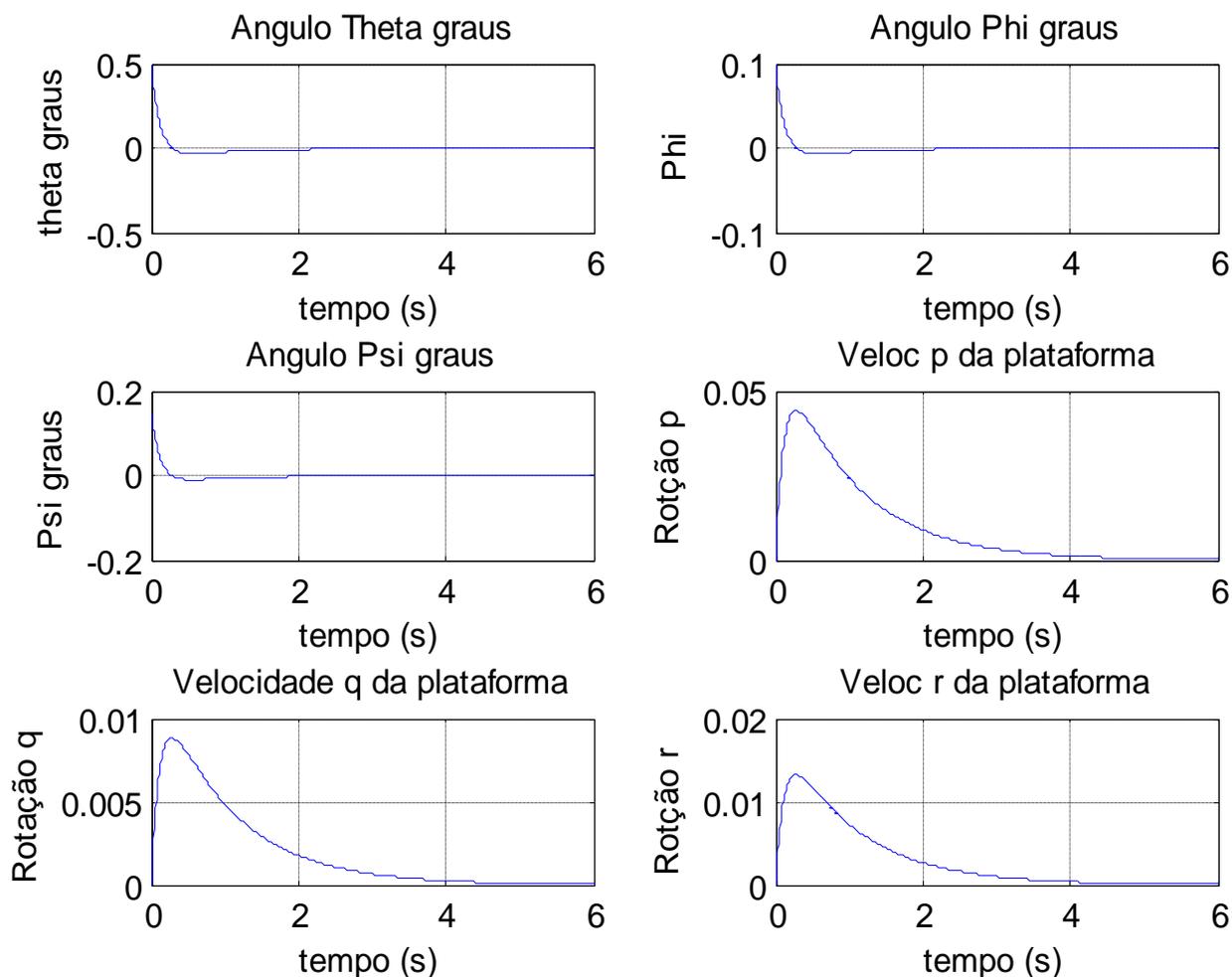


Figura 4.9 - Resposta do controlador LQR clássico

Na Figura 4.9, p , q e r são respectivamente as velocidades de deslocamento da plataforma segundo os eixos X , Y e Z do sistema inercial

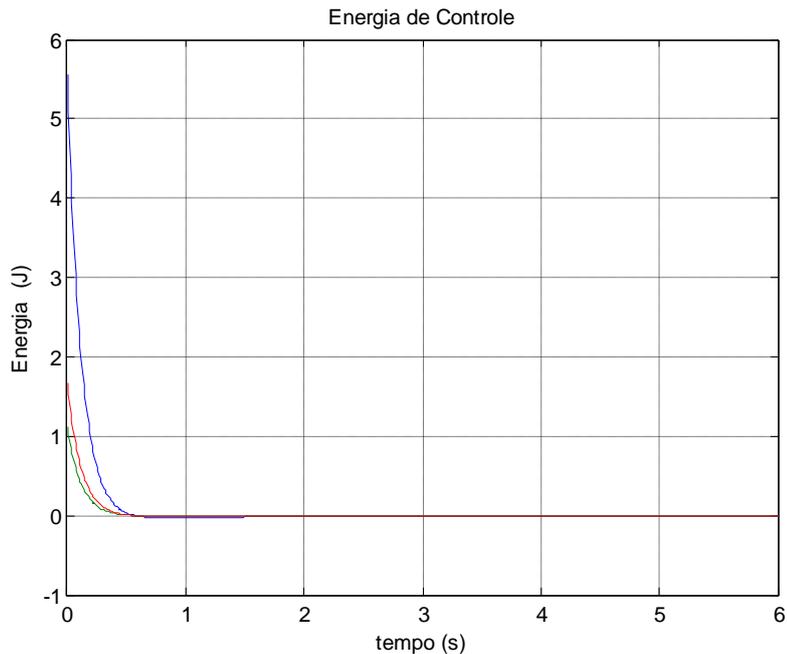


Figura 4.10 - Consumo de energia do controlador LQR clássico $R=1$

A Figura 4.9 mostra que as piores saídas entraram em controle em um tempo próximo de cinco segundos (saídas **p**, **q** e **r**).

Em seguida foi usado no simulador em estudo um controlador LQR/*Fuzzy*, cuja matriz **Q** foi obtida por lógica *Fuzzy*. Foram realizadas diversas simulações, nas quais se manteve um determinado valor de **R** constante e variaram-se as condições das entradas conforme tabelas 1 e 2. Os gráficos a seguir mostram os resultados destes experimentos, onde como entrada usou-se diversos impulsos através da função *initial* do Matlab, com condições iniciais variadas.

4.1.4 Resultados de um controlador LQR/*Fuzzy*

Nesta seção apresenta-se os resultados de um controlador LQR/*Fuzzy*. Inicialmente a Figura 4.11 mostra os resultados deste controlador para a fase em que a técnica *Fuzzy* ainda não atuou, ou seja, a matriz de ponderação de estado **Q** ainda não foi calculada pelo controlador *Fuzzy*. Inicialmente, para que o programa pudesse rodar usou-se um valor arbitrário para a matriz **Q** muito

baixo ($Q=0,0001$); para a ponderação R adotou-se escalar 1, valor que foi mantido durante parte das simulações. A Figura 4.12 mostra a resposta do sistema controlado e a Figura 4.13 mostra o consumo de energia.

Condições iniciais: $x_1=0,5$; $x_2=0,1$; $x_3=0,15$; $x_4=0$; $x_5=0$; $x_6=0$ $R=1$

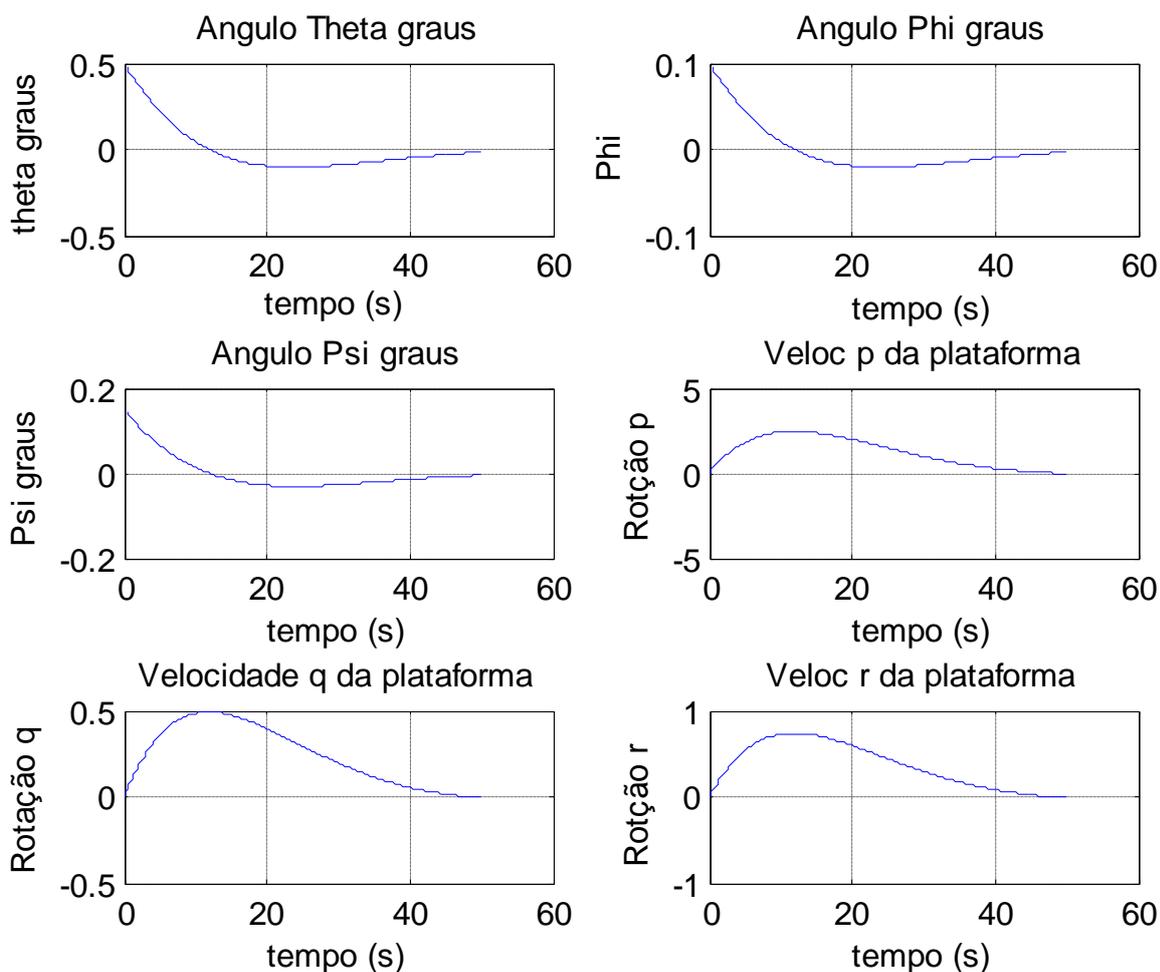


Figura 4.11 - Resultados do controlador LQR/*Fuzzy* inicialmente sem controle

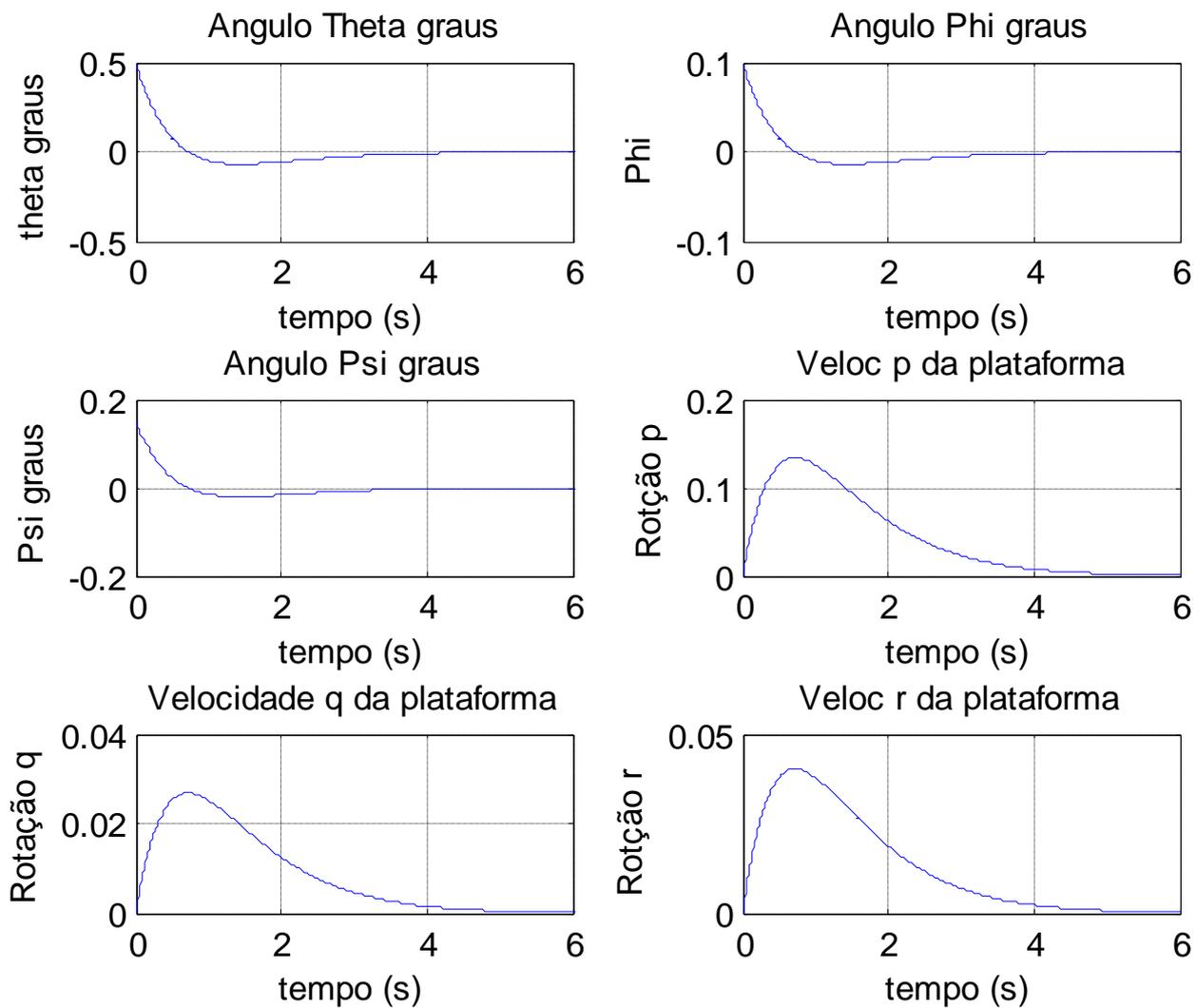


Figura 4.12 - Resultados do controlador LQR/*Fuzzy* com controle

Os resultados do controlador LQR/*Fuzzy* dados pela Figura 4.12 convergem muito mais rápido que os resultados da Figura 4.11, visto que escala de tempo da Figura 4.12 vai de zero a dez segundos enquanto a escala da Figura 4.11 vai de zero a sessenta segundos.

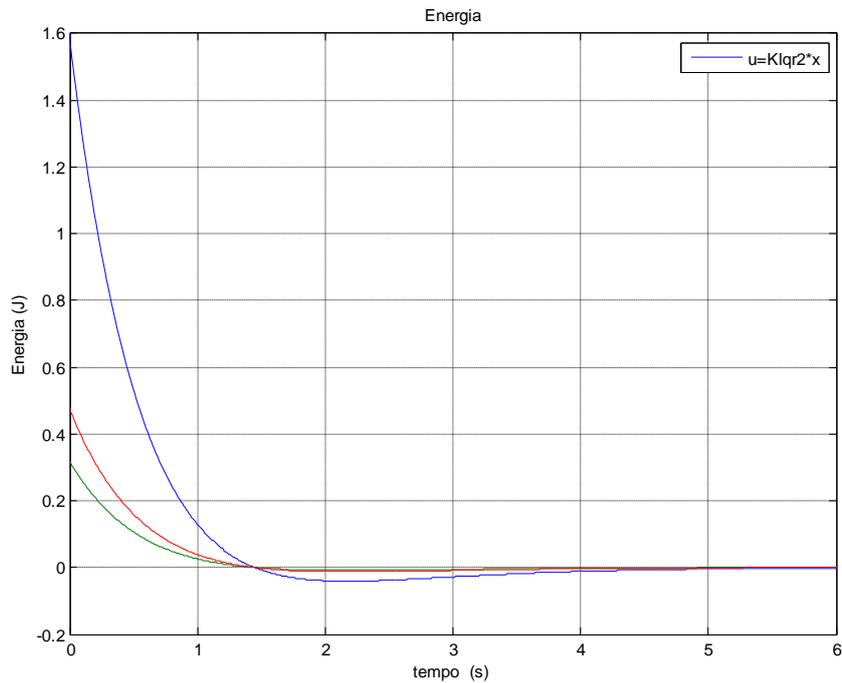


Figura 4.13 - Consumo de energia

4.1.5 Verificação de relação entre Q e R

Com a finalidade de verificar alguma interdependência entre as matrizes **Q** e **R**, novamente utilizou-se o programa usado para fornecer os resultados mostrados pelas Figuras 4.11, 4.12 e 4.13. Foram realizadas diversas simulações, mantendo-se fixo o valor da matriz **R** e variando-se as condições de entrada; atribuiu-se à matriz **R** os valores escalares: $R=1$. $R=10$. Das simulações realizadas apresentam-se duas séries de resultados mostrados pelas tabelas 4.1 e 4.2, dadas a seguir:

1) Para $R=1$

Tabela 4.1- Relações entre Q e R

simulação	X1	X2	X3	R	q
1	0,5	0,1	0,15	1	2,658275
2	1,0	0,2	0,3	1	2,660849
3	2,0	0,4	0,6	1	2,664559
4	4,0	0,8	1,0	1	2,666666
5	5,0	2,0	0,9	1	2,666400
6	5,0	2,0	1,0	1	5,000000

$Q=\text{diag}(q,6)$

2) Para $R=10$

Tabela 4.2 - Relações entre Q e R

Simulação	X1	X2	X3	R	q
1	0,5	0,1	0,15	10	2,658275
2	1,0	0,2	0,3	10	2,660849
3	2,0	0,4	0,6	10	2,664559
4	4,0	0,8	1,0	10	2,666666
5	5,0	2,0	0,9	10	2,666401
6	5,0	2,0	1,0	10	5,000000

$$\mathbf{Q} = \text{diag}(q, 6)$$

Observando-se a coluna \mathbf{q} das Tabelas 4.1 e da Tabela 4.2, nota-se que os valores destas colunas não alteram de uma tabela para outra, embora da Tabela 4.1 para a Tabela 4.2, a ponderação de custo de controle \mathbf{R} tenha variado de 1 para 10; as condições de entradas nas duas tabelas são as mesmas. Isto indica que, para o controle Fuzzy, não existe dependência entre \mathbf{Q} e \mathbf{R} .

As Figuras 4.14, 4.15 e 4.16 mostram respectivamente as respostas dos estados do controlador LQR/*Fuzzy*, para as simulações 2, 4 e 6 da Tabela 4.2.

condições iniciais: $x_1=1$, $x_2=0,2$, $x_3=0,3$, $x_4=0$, $x_5=0$ e $x_6=0$

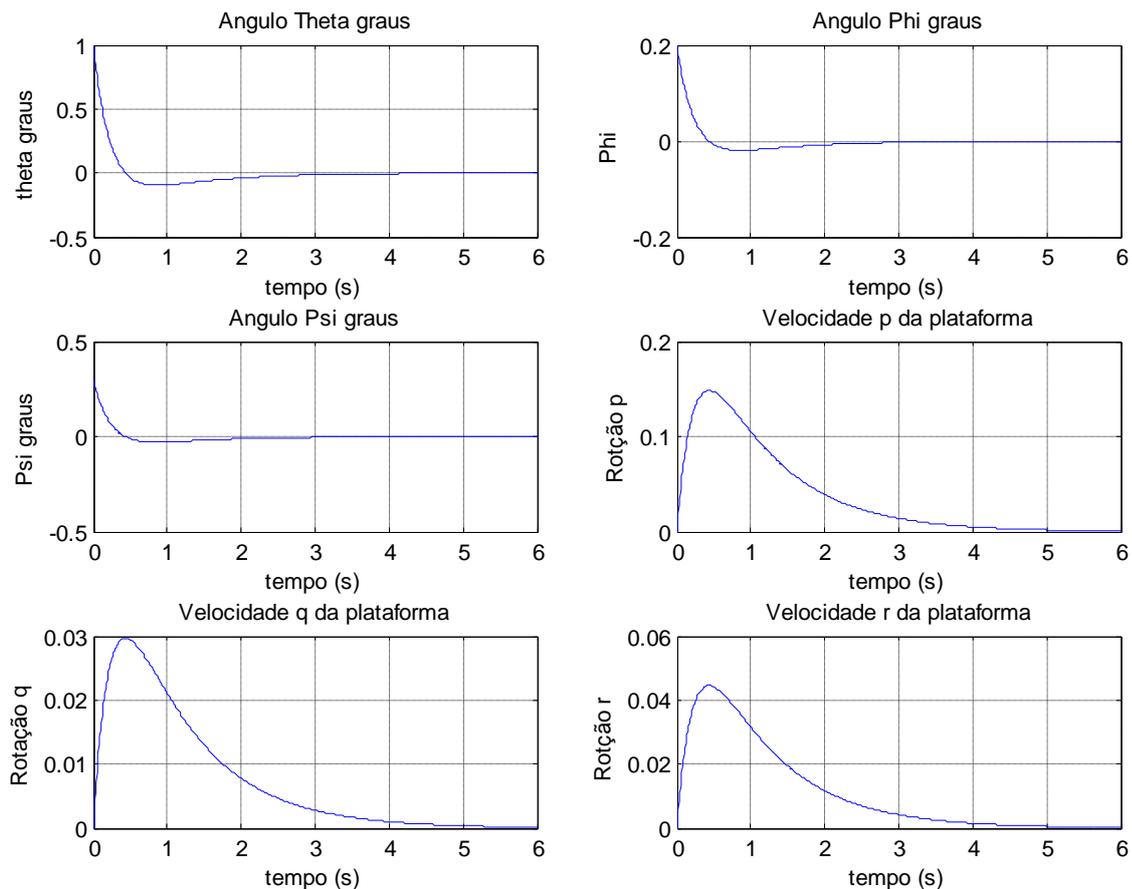


Figura 4.14 - Respostas para a Simulação 2 da Tabela 4.2.

Condições iniciais: $x_1=4$, $x_2=0,8$, $x_3=1,0$, $x_4=0$, $x_5=0$ e $x_6=0$

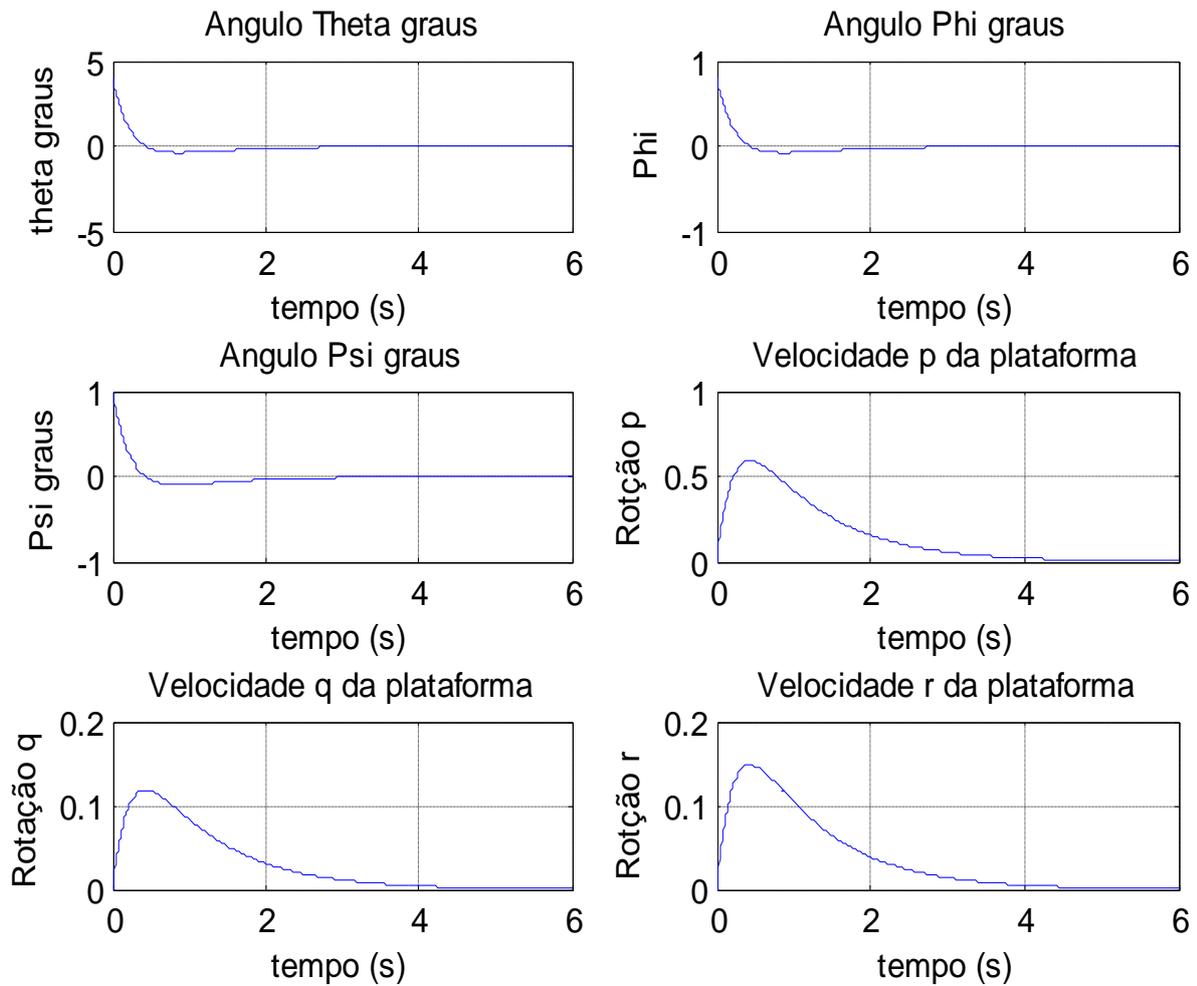


Figura 4.15 - Resposta para a Simulação 4 da Tabela 4.2

Condições iniciais: $x_1=5$, $x_2=2$, $x_3=1$, $x_4=0$, $x_5=0$ e $x_6=0$

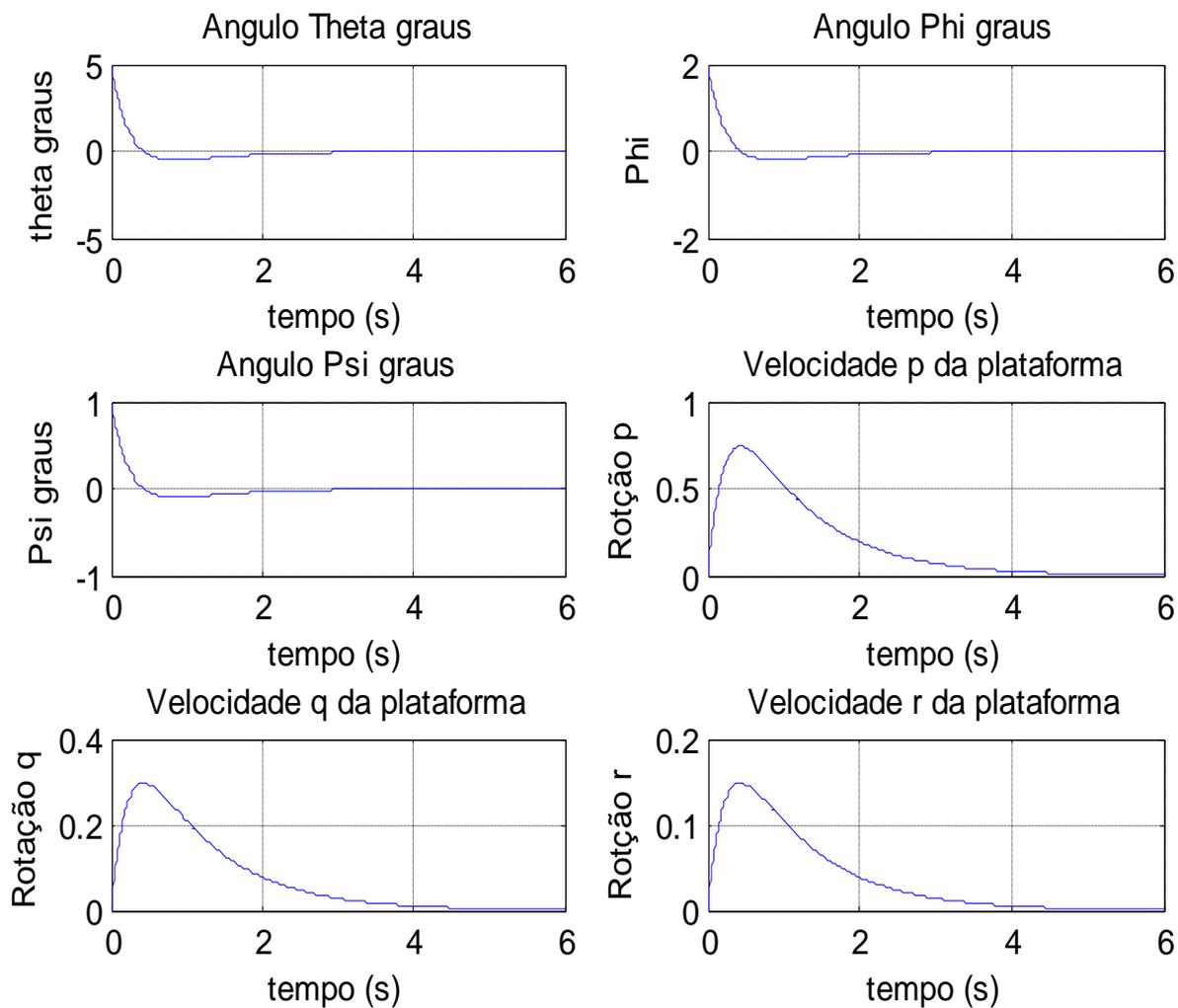


Figura 4.16 - Resposta para a Simulação 6 da Tabela 4.3

As Figuras 4.17, 4.18 e 4.19 mostram as curvas de consumo de energia, respectivamente das simulações 2, 4 e 6 da tabela 4.2

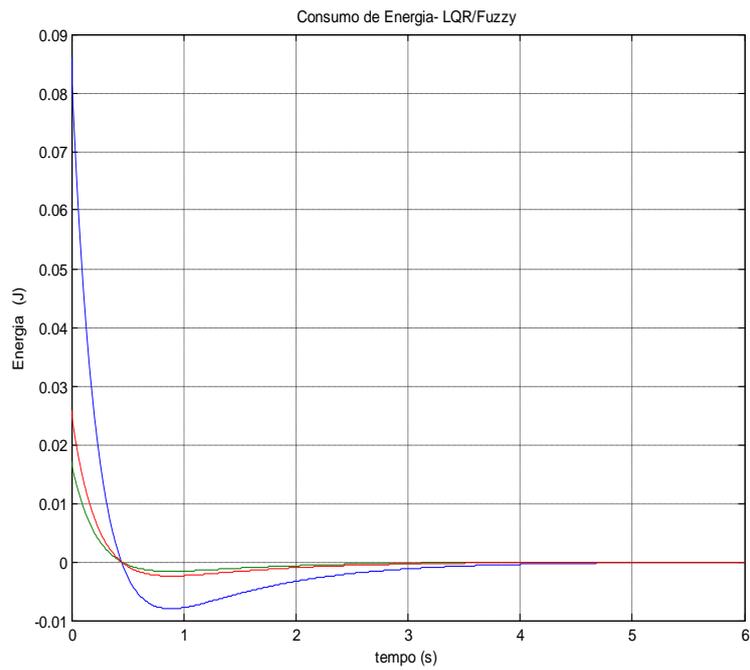


Figura 4.17 - Consumo de Energia da Simulação 2 da Tabela 4.2

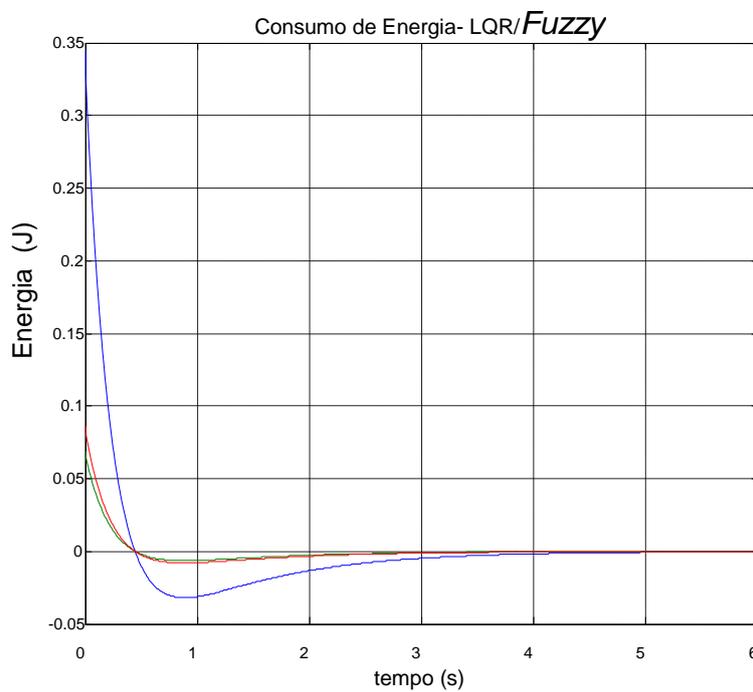


Figura 4.18 - Consumo de Energia da Simulação 4 da Tabela 4.2.

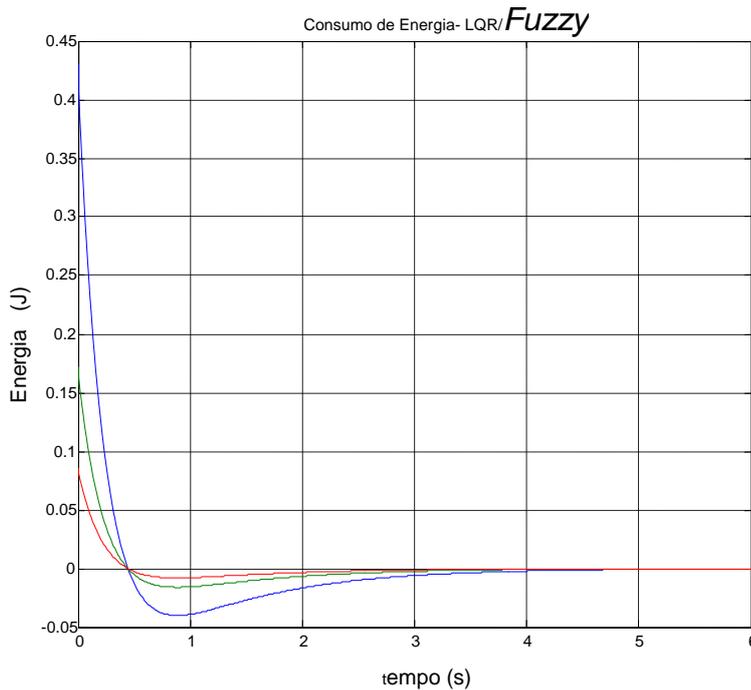


Figura 4.19 - Consumo de Energia da Simulação 6 da Tabela 4.2

4.2 Controlador SDRE clássico e SDRE/*Fuzzy*

Nesta seção é feito um breve estudo sobre o controlador SDRE/*Fuzzy*, similar ao estudo realizado na seção anterior para controlador LQR/*Fuzzy*. O sistema que será usado é o mesmo simulador da seção anterior dados pelas matrizes 4.3 e 4.4 que estão repetidas abaixo sob os números 4.5 e 4.6.

$$A=[\text{zeros}(3,6); \tag{4.5}$$

$$1 \ 0 \ 0 \ 0 \ 0 \ 0;$$

$$0 \ 1 \ 0 \ 0 \ 0 \ 0;$$

$$0 \ 0 \ 1 \ 0 \ 0 \ 0];$$

$$B=[-0,856 \ 0 \ 0; \tag{4.6}$$

$$0 \ -0,856 \ 0;$$

$$0 \ 0 \ -0,856;$$

$$\text{zeros}(3,3)];$$

em que $A_{6 \times 6}$ é matriz da dinâmica do simulador e $B_{3 \times 6}$ é a matriz de controle.

Foram utilizados o programa desenvolvido pela função *fis* do Matlab e os diagramas Simulink apresentados na Figura 4.27 de tese.

Seguindo a metodologia da seção anterior, inicialmente apresentam-se os resultados de um controlador SDRE clássico, em que as matrizes Q e R , com R constante, foram determinadas por tentativa e erro, e inseridas no programa pelo projetista.

4.2.1 Resultados do Controlador SDRE

A Figura 4.20 mostra os resultados originais obtidos por Gonzales (GONZALES, 2009), usando as matrizes de ponderação e entradas abaixo:

$$Q = [1 \ 0 \ 0 \ 0 \ 0 \ 0; 0 \ 1 \ 0 \ 0 \ 0 \ 0; 0 \ 0 \ 1 \ 0 \ 0 \ 0; 0 \ 0 \ 0 \ 100 \ 0 \ 0; 0 \ 0 \ 0 \ 0 \ 100 \ 0; 0 \ 0 \ 0 \ 0 \ 0 \ 100]$$

$$R = [0.0001 \ 0 \ 0; 0 \ 0.0001 \ 0; 0 \ 0 \ 0.0001]$$

Entrada degrau (graus): [20 50 -30]

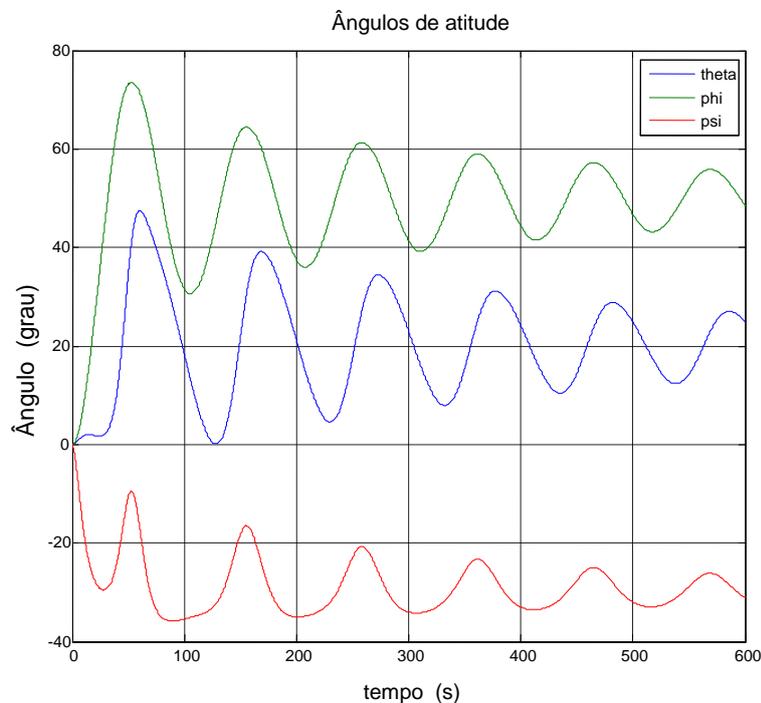


Figura 4.20 - Ângulos de atitude do controlador SDRE clássico.

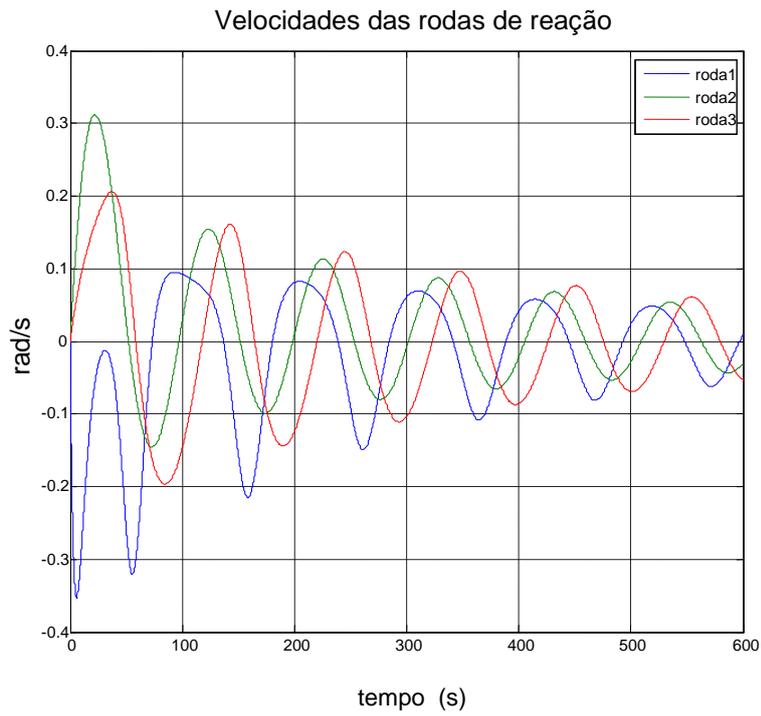


Figura 4.21 - Velocidades das rodas de reação Controlador SDRE clássico

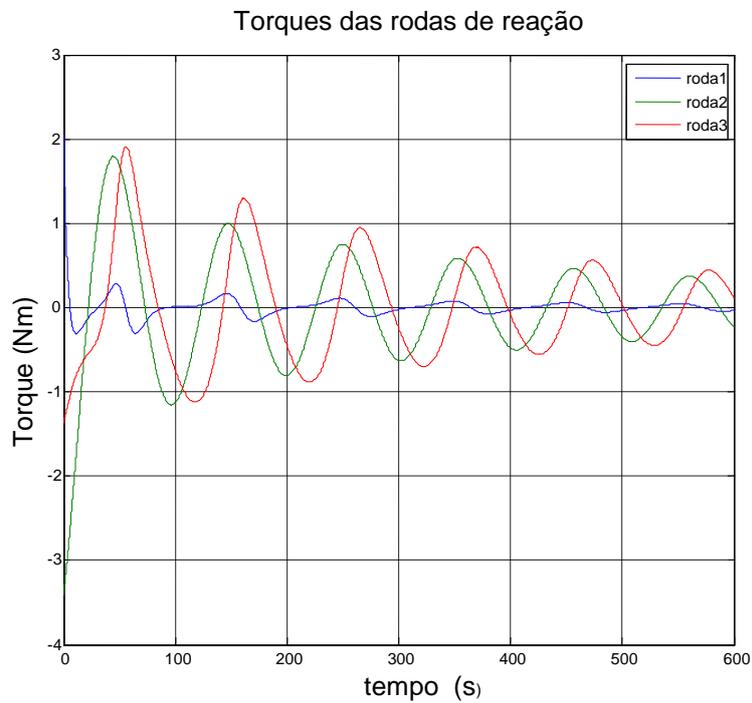


Figura 4.22 - Torques controlador SDRE clássico

4.2.2 Verificação de relação entre Q e R no controlador SDRE/Fuzzy

Foram realizadas várias simulações, usando diversas condições iniciais e também simulando variações no erro, os resultados estão na Tela 4.3 dada a seguir:

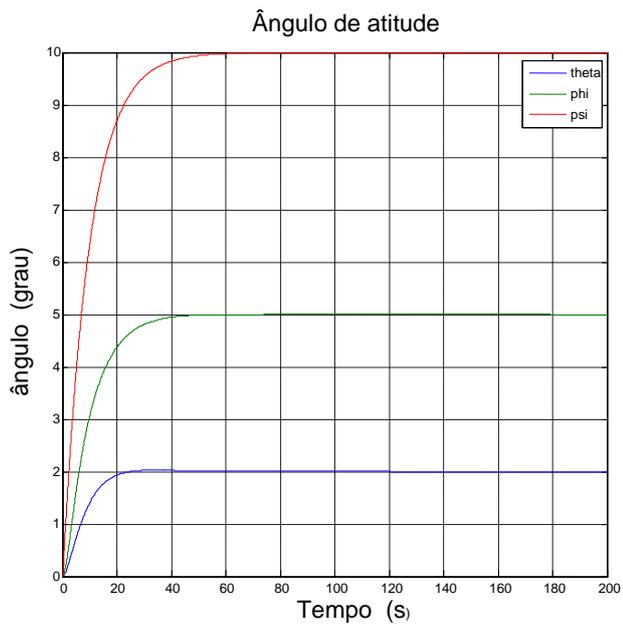
Tabela 4.3 Relações entre Q e R no SDRE/Fuzzy

Simulação	theta	phi	psi	R	erro	q
1	2	5	10	1e-4	0,1	2,333333
2	5	10	20	1e-4	0,15	2,335594
3	8	-15	-30	1e-4	0,2	2,341427
4	15	-25	-30	1e-4	0,1	2,333333
5	20	50	-30	1e-4	0,15	2,335594

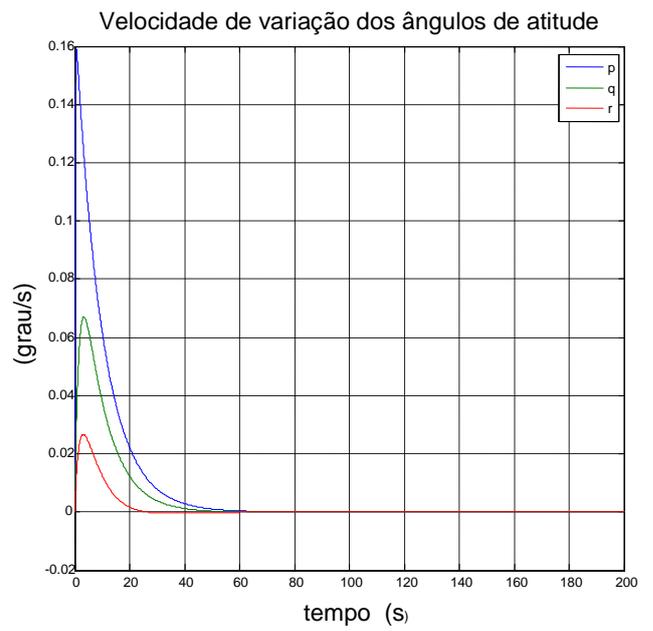
As Figuras 4.23a,b,c, 4.24a,b,c e 4.25a,b,c apresentam respectivamente, os resultados das simulações números 1, 3 e 5 da Tabela 4.3. Os ângulos são dados em graus.

Lembre-se que a matriz de ponderação de estado **Q** é dada pela expressão abaixo, em que a diagonal principal é formada pelo elemento **q** dado na Tabela 4.3.

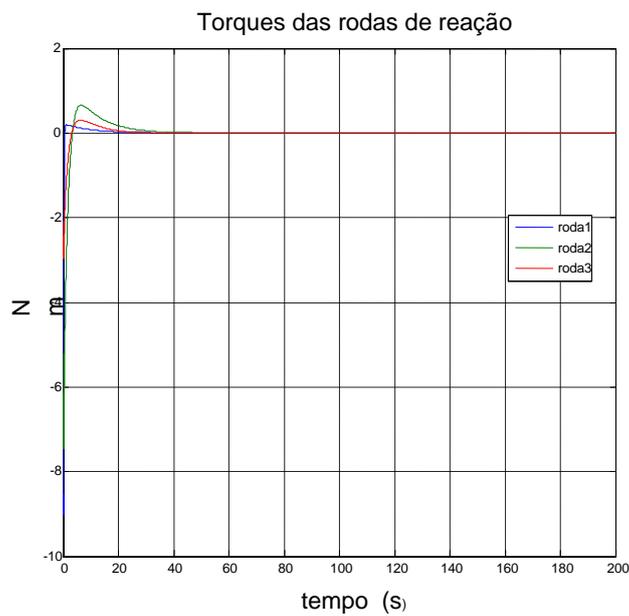
$$Q = \begin{bmatrix} q & 0 & 0 & 0 & 0 & 0 \\ 0 & q & 0 & 0 & 0 & 0 \\ 0 & 0 & q & 0 & 0 & 0 \\ 0 & 0 & 0 & q & 0 & 0 \\ 0 & 0 & 0 & 0 & q & 0 \\ 0 & 0 & 0 & 0 & 0 & q \end{bmatrix} \quad (4.7)$$



(a)

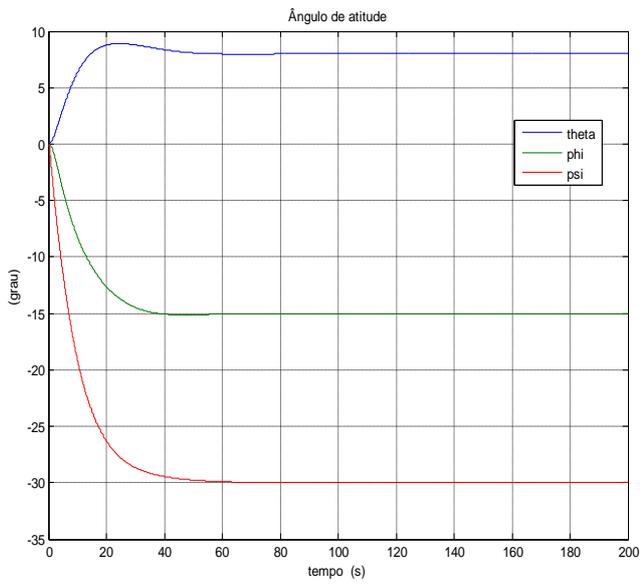


(b)

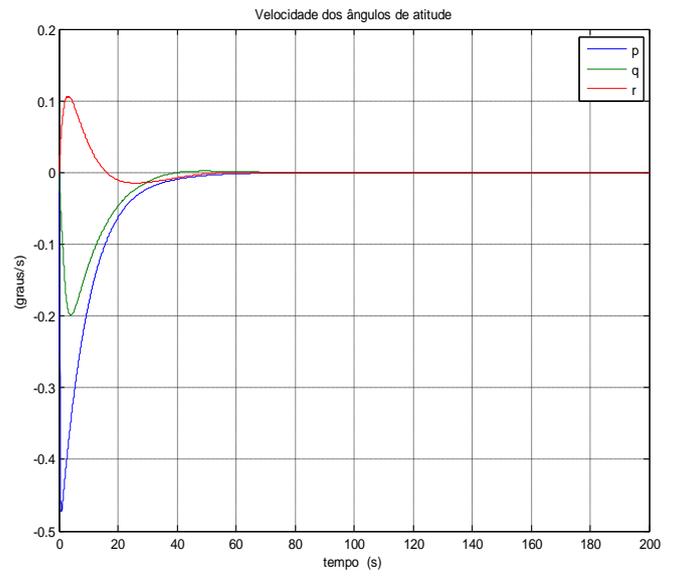


(c)

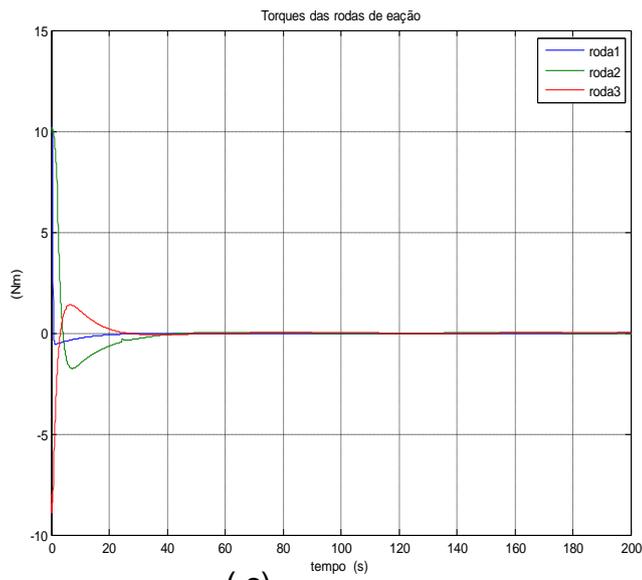
Figura 4.23 a,b,c - Respostas para simulação 1 da Tabela 4.3



(a)



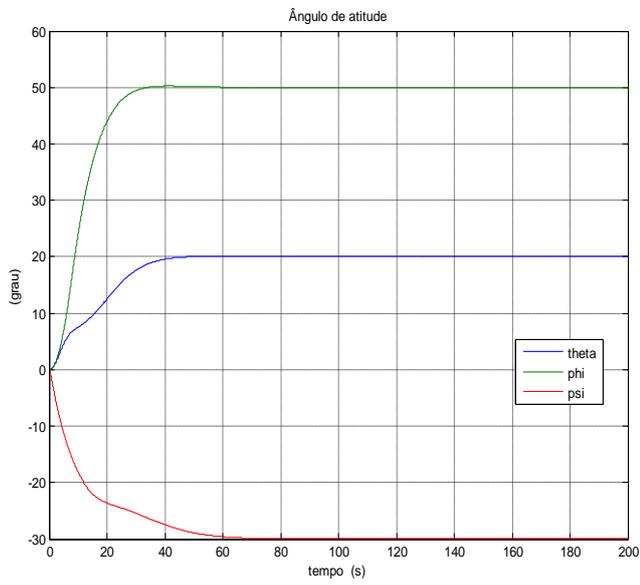
(b)



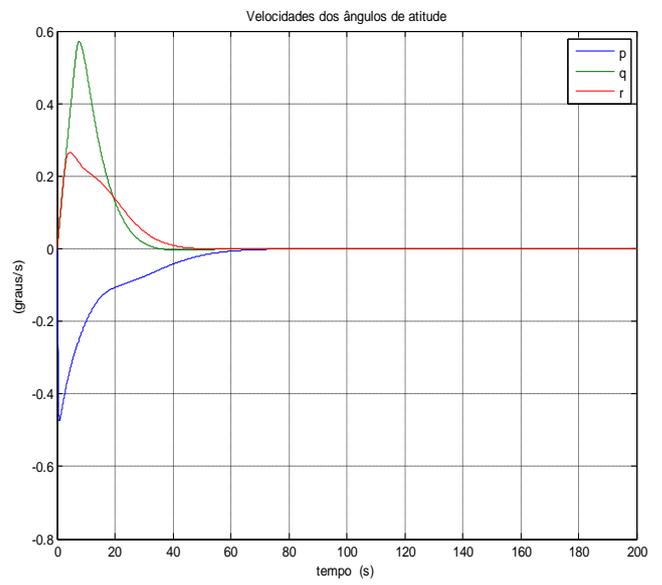
(c)

(c)

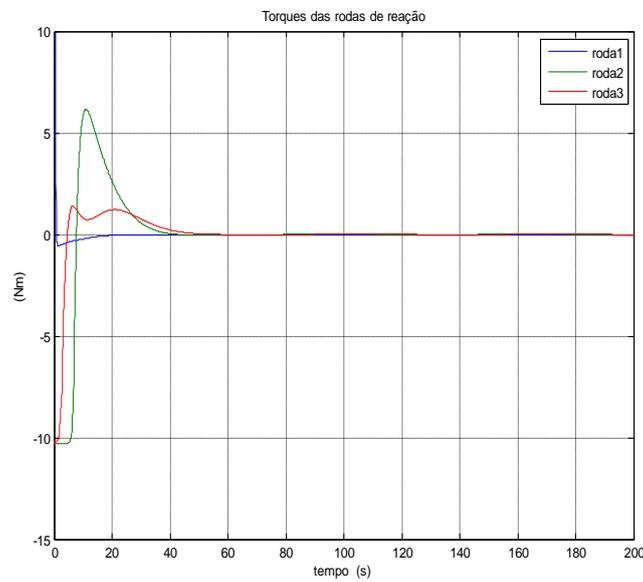
Figura 4.24a,b,c - Respostas da simulação 3 da Tabela 4.3



(a)



(b)



(c)

Figura 4.25a,b,c - Respostas da simulação 5 da Tabela 4.3

Da análise da Tabela 4, vê-se que o valor da matriz \mathbf{Q} depende do erro que, no caso real, é função das condições de entrada e da informação de um sensor de posição do simulador. Lembre-se que a matriz \mathbf{Q} , neste trabalho, é uma matriz diagonal, cujos elementos são os valores de q : $\mathbf{Q}=q \cdot \text{eye}(6)$.

Como no caso do LQR/*Fuzzy*, o valor de R foi mantido constante com um valor muito pequeno, conforme mostrado na Tabela 4.

4.2.3 Análise das respostas dos controlador LQR/*Fuzzy*

Para analisar o desempenho do controlador LQR/*Fuzzy*, introduzido neste trabalho, em relação a outro controlador (controlador por alocação de pólos) usou-se as curvas de respostas transitórias das Figuras 4.3a,b, 4.4a,b e 4.5a,b por fornecerem maior quantidade de informações, conforme segue:

1) Em relação ao controle de atitude do simulador de satélite, aqui estudado, vê-se que o controlador LQR/*Fuzzy* é cerca 2(dois) segundos mais lento que o controlador por alocação de pólos; a vantagem do controlador LQR/*Fuzzy* está na maior suavidade de controle mostrada, por exemplo, pela curva de controle do ângulo de atitude ϕ (curva azul);

2) Em relação ao controle de velocidade, vê-se que a pequena lentidão de 2 (dois) segundos, em relação ao controlador tomado como referência permanece, porém, observa-se que controlador LQR/*Fuzzy* proporciona maior suavidade de controle nas rotações das três rodas de reação.

A vantagem mais importante do LQR/*Fuzzy* é limitar as rotações das três rodas de reação a um valor mais baixo, ficando mais distante de possíveis saturações destas rodas;

3) As curvas de torque do controlador LQR/*Fuzzy* apresentam perfis coerentes com as observações feitas sobre as curvas de atitude e velocidades acima.

A vantagem deste controlador sobre controlador por alocação de pólos é a utilização de menor torque para o controle, obtendo com isto maior economia de energia.

4) Em síntese, em relação ao controlador por alocação de pólos, o controlador LQR/*Fuzzy* apresentou um pequeno retardo, porém, proporcionou maior suavidade de controle, menor rotação das rodas de reação, menor torque, além de menor tempo de projeto, visto que a matriz de ponderação Q usada foi determinada de forma automática pelo controlador *Fuzzy* inserido no controlador LQR.

Da comparação entre as curvas de resposta dos controladores acima tratados vê-se que o controlador por alocação de pólos apresentou um bom desempenho, contudo o controlador LQR/*Fuzzy* apresentou algumas vantagens como:

- As rodas de reação atingiram menor rotação durante o controle, isto é bom para ficar mais distante de possível saturação.
- O torque necessário para controle foi menor, isto é bom para a economia de energia.
- Uma terceira vantagem está no projeto que é mais imediato resultando menor trabalho computacional.

Da comparação acima, justifica-se o emprego do controlador *Fuzzy* para melhorar o desempenho do controlador LQR; este é o primeiro objetivo deste trabalho.

4.2.4 Controladores SDRE convencional e SDRE/*Fuzzy*

O controlador SDRE (State-Dependent Riccati Equation), cujos fundamentos básicos foram dados na Seção 3.5, pode ser considerado uma adaptação do controlador LQR para sistemas não lineares, formado pelas matrizes $\mathbf{A}(\mathbf{x})$ e $\mathbf{B}(\mathbf{x})$ como matrizes de estado e de controle dependentes dos estados e das matrizes $\mathbf{Q}(\mathbf{x})$ e $\mathbf{R}(\mathbf{x})$ como matrizes de ponderação também dependentes dos estados; para simplificação de projeto a matriz \mathbf{Q} pode ser uma matriz de constantes e a matriz \mathbf{R} pode ser um escalar, a exemplo do que ocorre em projetos de controlador LQR (ANDERSON; MOORE, 2007). O controlador SDRE herda do controlador LQR o inconveniente da determinação das matrizes de ponderação que, como já foi dito, são determinadas por tentativa e erro pelo projetista.

4.2.5 Funcionamento do controlador SDRE/*Fuzzy*

O funcionamento do controlador SDRE/*Fuzzy*, proposto neste trabalho, é similar ao funcionamento do controlador SDRE convencional, visto que seu algoritmo é o mesmo algoritmo do controlador SDRE, diferenciado pelo fato de a equação de Riccati do controlador LQR, que compõe o controlador SDRE, não ser constante e sim dependente do estado. Inicialmente, o controlador SDRE/*Fuzzy* roda com matrizes de ponderação \mathbf{Q} e \mathbf{R} de valores arbitrários, não nulos, incluídos no algoritmo pelo projetista; a saída deste controle preliminar é comparada com um valor de erro (seria a precisão do controle) determinado pelo projetista; o erro resultante desta comparação é enviado para o controlador *Fuzzy* (ver apêndice A₁), responsável por obter valores sub-ótimos e definitivos para estas matrizes de ponderação. O valor de \mathbf{Q} obtido é enviado para um controlador desenvolvido em Simulink por Gonzales (GONZALES, 2009; CAMPA, 2002), e adaptado neste trabalho para se tornar um controlador SDRE/*Fuzzy*, responsável pela solução numérica do controle.

4.2.6 Diferença entre os controladores SDRE e SDRE/*Fuzzy*

A diferença entre os controladores SDRE e SDRE/*Fuzzy*, conforme dito na seção anterior está no projeto do controlador LQR que compõem o controlador SDRE/*Fuzzy*; no caso do controlador SDRE convencional, as matrizes de ponderação $\mathbf{Q}(\mathbf{x})$ e $\mathbf{R}(\mathbf{x})$, que também podem ser constantes, \mathbf{Q} e \mathbf{R} e \mathbf{R} na forma de matriz diagonal unitária, são determinadas por tentativas "off line", ou seja, são determinadas pelo projetista, fora do algoritmo de controle. No controlador SDRE/*Fuzzy*, a matriz \mathbf{Q} é obtida de forma automática ("on line") por um controlador *Fuzzy* que compõe o algoritmo do controlador SDRE/*Fuzzy*; a matriz \mathbf{R} é considerada como uma matriz diagonal unitária.

A incorporação do controle *Fuzzy* no controlador SDRE, para obter o controlador SDRE/*Fuzzy*, como parte dos objetivos deste trabalho, foi obtida desenvolvendo-se, através do Simulink, o controle *Fuzzy* mostrado na Figura 4.26 e inserindo-o no bloco *Controller-SDRE* existente no diagrama da Figura 4.27, conforme mostrado em detalhe pelo bloco *Fuzzy* na Figura 4.28

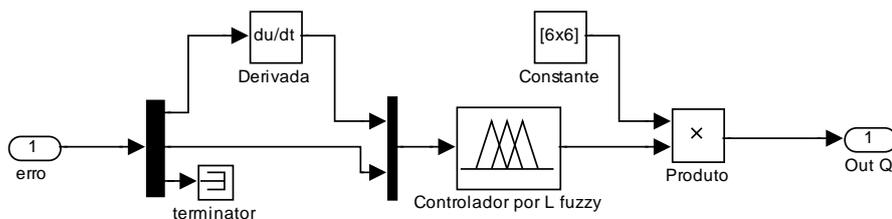


Figura 4.26- Controle *Fuzzy* do controlador SDRE/*Fuzzy*

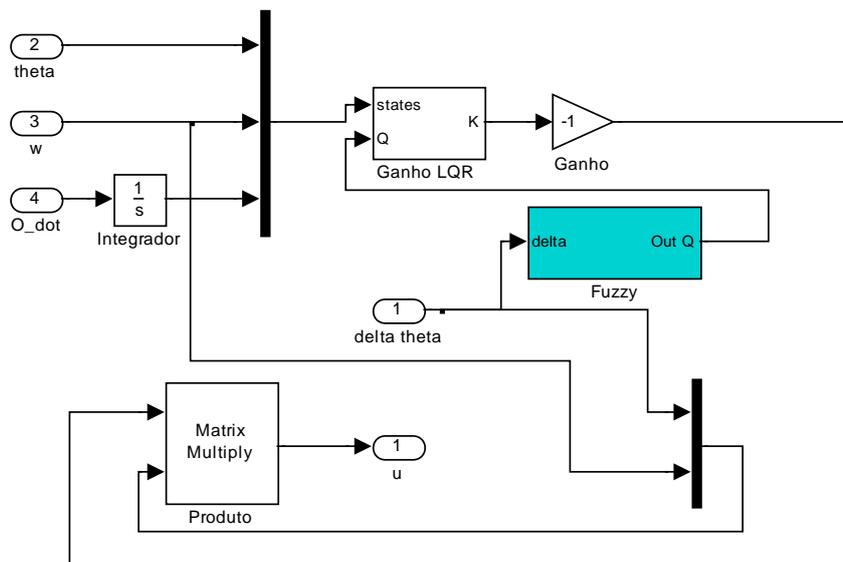
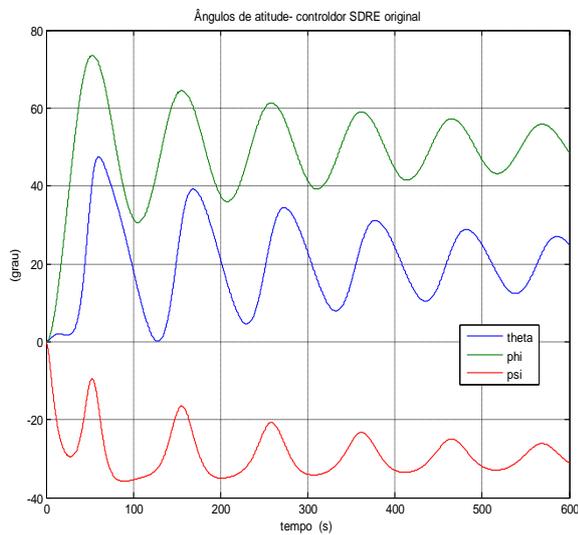


Figura 4.28 - Incorporação do controle *Fuzzy*

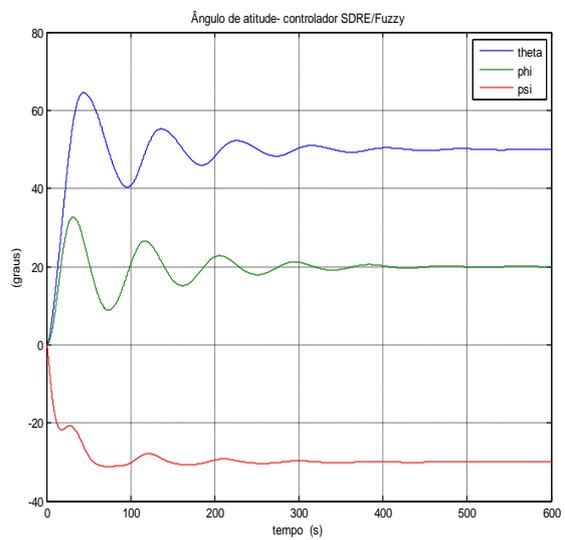
O bloco *Fuzzy Logic controller* da Figura 4.26 contém a lógica de controle *Fuzzy* que foi implementada analiticamente pelo *Fis Editor* que é um programa contido no *Fuzzy Logic Toolbox* do Matlab; este algoritmo é apresentado no Apêndice A1. Para operar este controle, inicialmente roda-se o programa responsável por passar os parâmetros do simulador e a lógica de controle *Fuzzy* para o controlador Simulink mostrado na Figura 4.28.

4.2.7 Comparação dos resultados entre controladores SDRE e SDRE/Fuzzy

Para fins de comparação, apresentam-se os resultados do controlador SDRE cujas matrizes \mathbf{Q} e \mathbf{R} , ambas constantes, foram determinadas por alocação de pólos por Gonzales (2009) e os resultados obtidos pelo controlador SDRE/Fuzzy. Para uma melhor visualização e facilidade de comparação estas Figuras foram postas lado a lado de acordo com o tipo de resposta, conforme mostra as Figuras 4.29a,b, 4.30a,b e 4.31a,b, dadas a seguir

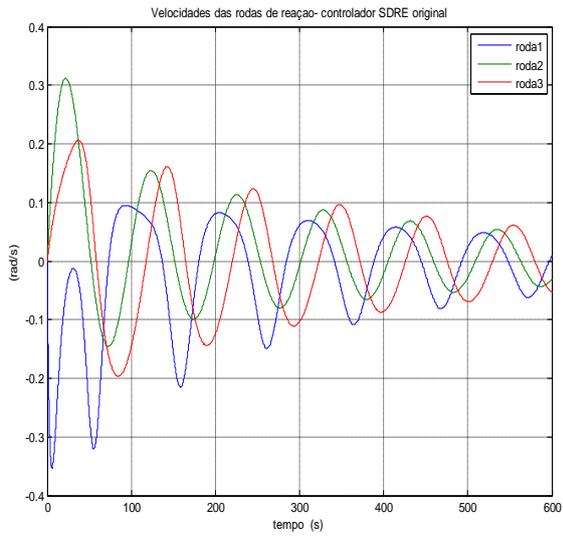


(a)

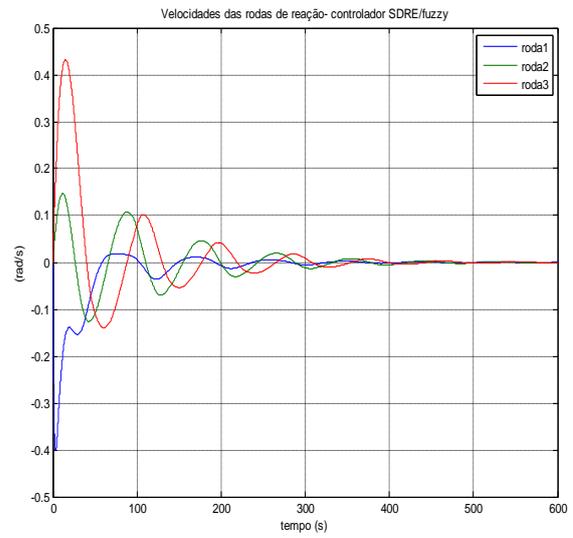


(b)

Figura 4.29a,b - Comparação entre Respostas – Ângulos de atitude

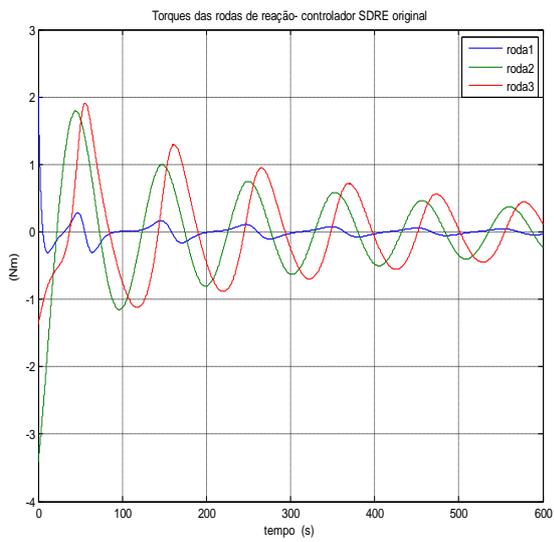


(a)

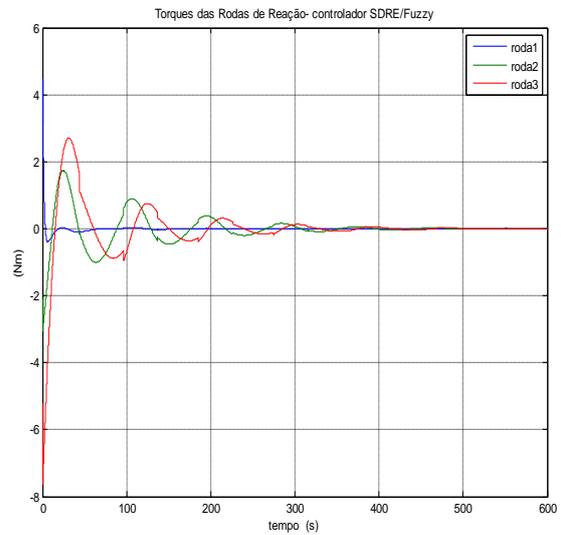


(b)

Figura 4.30a,b - Comparação entre Respostas –Velocidades das rodas de reação



(a)

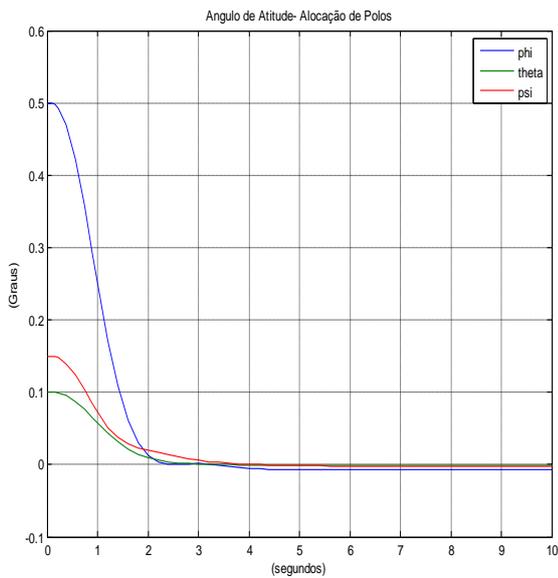


(b)

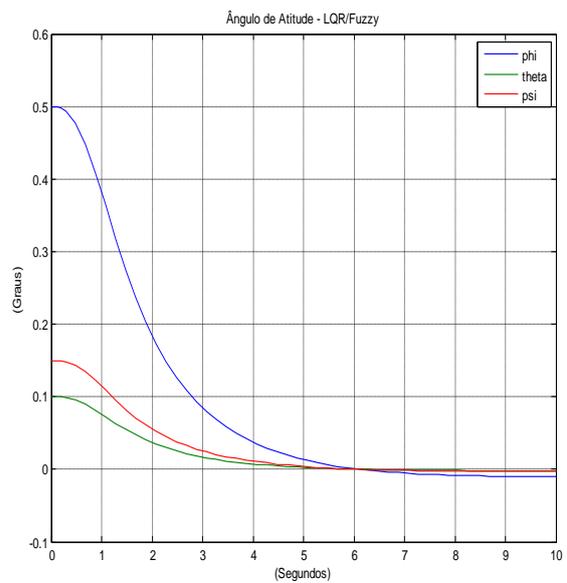
Figura 4.31a,b - Comparação entre Respostas – Torques

4.2.8 Considerações sobre os resultados

As figuras 4.3a,b, 4.4a,b e 4.5a,b repetidas a seguir, para facilitar a consulta, com os números 4.32a,b, 4.33a,b e 4.34a,b respectivamente, apresentam curvas com resultados numéricos muito próximos entre os controladores tratados, não mostrando vantagem numérica do controlador LQR/*Fuzzy* em relação ao controle por alocação de pólos; isto pode ocorrer porque sempre é possível, por uma busca exaustiva, procurar uma melhor posição para os pólos afim de obter melhores resultados. O problema é que esta busca, além de trabalhosa demanda muito tempo dando vantagem para o controlador LQR/*Fuzzy*.

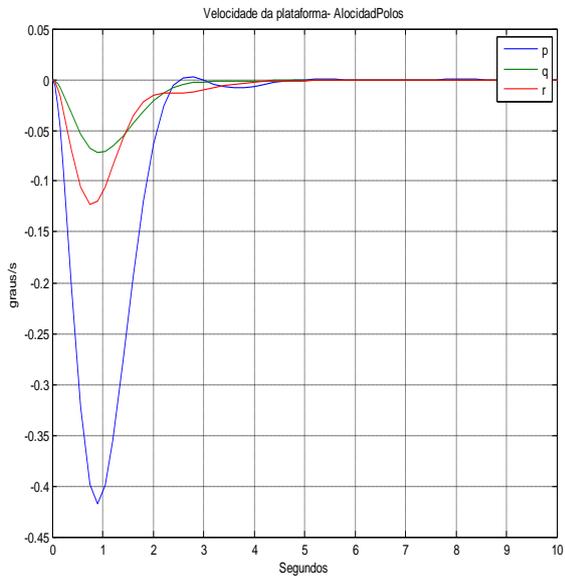


(a)

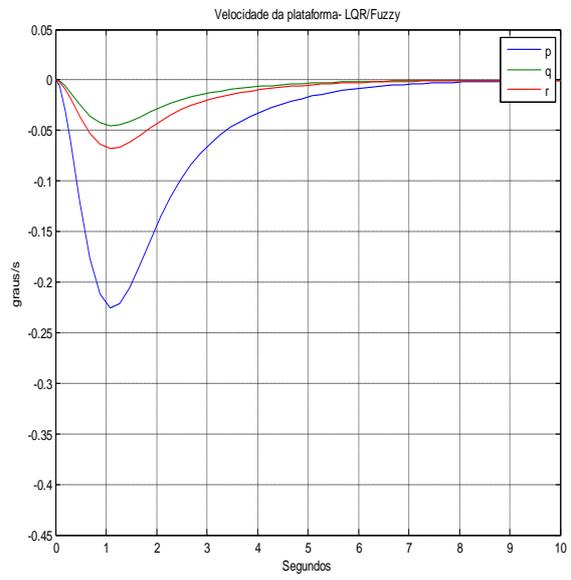


(b)

Figura 4.32a,b- Respostas transitórias de atitude - Alloc. de pólos versus LQR/*Fuzzy*

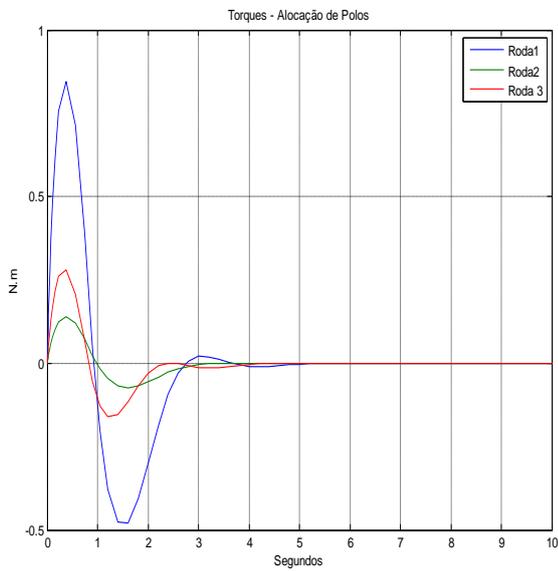


(a)

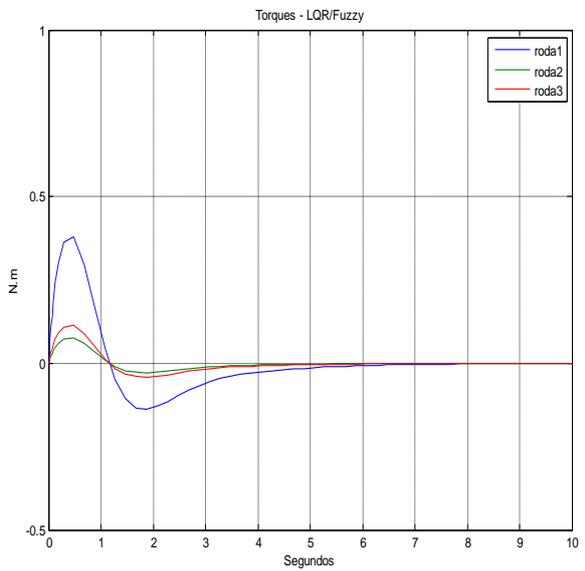


(b)

Figura 4.33a,b - Resposta transitória de velocidade – Aloc. pólos versus LQR/Fuzzy



(a)



(b)

Figura 4.34a,b - Resposta transitória de torque – Aloc. pólos versus LQR/Fuzzy

O que é importante ressaltar, em favor do uso da técnica *Fuzzy* no controlador LQR, é que esta técnica funciona de forma eficiente e tem a vantagem sobre o controlador convencional de determinar, de forma automática as matrizes de ponderações **Q** e **R**. Uma outra vantagem do controlador LQR/*Fuzzy* é sua robustez, verificada por exaustivas simulações, aplicando-se em cada simulação condições iniciais diferentes.

As figuras 4.29a,b, 4.30a,b e 4.31a,b comparam o desempenho de um controlador SDRE convencional, onde determinação de **Q** foi feita por tentativa e erro, com um controlador SDRE/*Fuzzy* que determina esta matriz de forma automática; neste caso é notória a vantagem deste último controlador.

Na seção 3.4.12, a revisão da técnica de controlador H_{INFINITO} foi complementada com um exemplo de controle H_{INFINITO} desenvolvido pelo programa *Mixsyn/Matlab*; constata-se que esta técnica tem um bom desempenho, conforme mostrado pelas Figuras 3.25a, 3.25b, 3.26a e 3.26b.

Na faixa de frequência de 0,1 a 0,95 rad/s o controle não é robusto visto que não atende às relações 3.49 e 3.50. A robustez só é verificada na faixa a partir 0.95 rad/s, conforme mostram as figuras 3.28 e 3.29; acima de aproximadamente 3 rad/s, não deve ser considerada como faixa útil para controle, visto estar a menos de uma década da frequência de ressonância do sistema que é de aproximadamente 26 rad/s, conforme mostrado na Figura 3.27; tem-se portanto uma faixa útil de controle de 1,0 a 3,0 rad/s na qual o controle do sistema é robusto. Poderão ser pesquisadas outras matrizes de ponderações **W_s** e **W_c** com o objetivo de obter faixa de robustez diferente da faixa obtida acima, conforme sugestão apresentada na Conclusão desta tese.

4.3 Controlador H_{INFINITO} convencional e $H_{\text{INFINITO}}/\text{AG}$

Para apresentar a última parte deste trabalho, conforme mencionado no Item 3 da Seção 1.3 e no final da Seção 4.1, nesta seção é desenvolvido um controlador H_{INFINITO} em que apenas os parâmetros M_s , ω_s , M_c , e ω_{bc} das funções de ponderação W_s e W_c são determinadas de forma automática com o auxílio de algoritmos genéticos (AG).

Conforme dito na Seção 1.3, o presente capítulo apresenta uma contribuição visando a simplificação e melhoria do trabalho de Cárdenas (CÁRDENAS, 2005); os resultados apresentados nesta seção foram obtidos aplicando um controlador H_{INFINITO} proposto auxiliado por algoritmos genéticos sobre o sistema formado pelas matrizes representativas de um satélite rígido-flexível, na sua versão linear, desenvolvido por Cárdenas e adaptado a este trabalho. As adaptações consistiram na substituição das funções obsoletas e reservadas existentes no programa, desenvolvidas pelo autor no ambiente Matlab[®], por funções existentes nas atuais versões do Matlab[®]. A contribuição de melhoria consistiu na adoção de valores fixos para os parâmetros δ_s e δ_c (deltas e deltac) das matrizes de ponderação W_s e W_c respectivamente, obtendo maior rapidez e maior precisão de controle. Os resultados obtidos neste trabalho foram comparados com os resultados do trabalho original, mostrados na seção 4.4.3.

4.3.1 Modelagem do satélite rígido-flexível

Cárdenas, usando a abordagem lagrangeana e a técnica de modos assumidos, modelou o satélite rígido-flexível sino-brasileiro CBERS-1. A Figura 4.35 mostra o referido artefato e a Figura 4.36 mostra os sistemas de coordenadas inercial e fixo no corpo e também os ângulos de atitude.



Figura 4.35- Satélite Sino-brasileiro CBERS-1

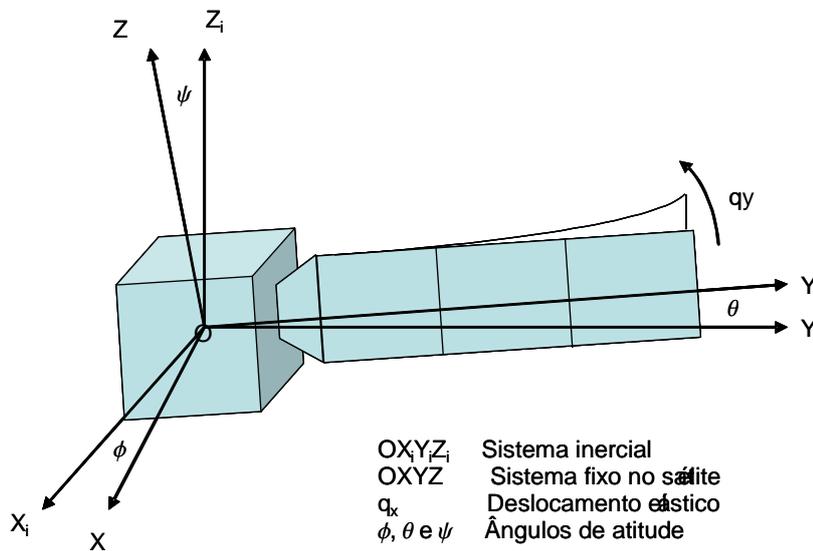


Figura 4.36- Satélite rígido-flexível CBERS-1

Após linearizar a matriz de massa do CBERS, Cárdenas chegou a uma matriz de dimensão seis por seis ($\mathbf{M}_{6 \times 6}$). Ao compor o sistema na forma de espaço de estado, usando a matriz \mathbf{M} encontrada, obteve uma matriz de dimensão doze por doze ($\mathbf{A}_{12 \times 12}$) não controlável e não observável; a instabilidade do sistema era causada pelo quinto elemento da matriz \mathbf{M} , obtida inicialmente. Este quinto elemento de \mathbf{M} , por ter valor numérico pouco significativo em relação aos demais elementos da matriz e por estar isolado na interseção da quinta linha

com a quinta coluna da matriz, foi eliminado gerando a matriz de ordem reduzida de \mathbf{M} ($\mathbf{M}_{5 \times 5}$).

Escrevendo o sistema na forma de espaço de estado, usando a matriz \mathbf{M} de ordem reduzida, Cárdenas obteve a matriz \mathbf{A} de dimensão dez por dez ($\mathbf{A}_{10 \times 10}$) dada por 4.4 e o sistema passou a ser controlável e observável; em correspondência a esta redução, as demais matrizes que formam o sistema na configuração de espaço tiveram suas ordens ajustadas para formar o sistema no espaço de estado dado por 4.8.

O modelo linear inicial, que era de doze estados foi reduzido para um modelo de dez estados, que é usado neste trabalho. Estes estados reduzidos são dados pelo vetor a seguir $x = [\phi \ \theta \ \psi \ \dot{\phi} \ \dot{\theta} \ \dot{\psi} \ q_x \ q_z \ \dot{q}_x \ \dot{q}_z]$, em que $(\phi, \theta \text{ e } \psi)$ e $(\dot{\phi}, \dot{\theta} \text{ e } \dot{\psi})$ são respectivamente os ângulos de atitude e velocidade de atitude do módulo rígido em relação ao referencial fixo $OX_i Y_i Z_i$ e (q_x, q_z) e (\dot{q}_x, \dot{q}_z) são respectivamente os deslocamentos elásticos e velocidades de deslocamentos elásticos da extremidade do módulo flexível (painel do satélite).

Usando a versão linearizada na forma numérica, foi obtido o sistema 4.8 abaixo

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}\tag{4.8}$$

Onde as matrizes \mathbf{A} , \mathbf{B} , \mathbf{C} e \mathbf{D} são dadas em 4.9, a seguir:

```

A= [0  0  0  0  0  1  0  0  0  0
      0  0  0  0  0  0  1  0  0  0
      0  0  0  0  0  0  0  1  0  0
      0  0  0  0  0  0  0  0  1  0
      0  0  0  0  0  0  0  0  0  1
      0  0  0  0  2,45 0  0  0  0  2,51e-05
      0  0  0  0  0  0  0  0  0  0
      0  0  0  -2,53 0  0  0  0  -2,51e-05 0
      0  0  0  -123,65 0  0  0  0  -0,0012 0
      0  0  0  0  -119,83 0  0  0  0  -0,0012]

```

```

B= [0  0  0
      0  0  0
      0  0  0
      0  0  0
      0  0  0
      0,00025  0  0
      0  0,00027  0
      0  00,00025
      0  00,00050
      -0,0005  0  0 ]

```

(4.9)

C= eye(6,10)

D=zeros(6,3)

Usando as funções *Sysic* e *Hinfsyn*, existente no Robust Control Toolbox do Matlab[®], Cárdenas desenvolveu um controlador H_{INFINITO} onde todos os parâmetros das funções de ponderação W_s e W_c foram determinadas por algoritmos genéticos. Este controlador foi usado para controle de atitude do satélite rígido-flexível cujas matrizes são dadas em 4.9; a Figura 4.37 apresenta o fluxograma deste controlador.

Para alcançar os objetivos deste trabalho, foram necessárias efetuar adaptações e modificações no programa original, conforme já foi dito na seção 4.3.

O fluxograma básico do programa adaptado, que neste trabalho é denominado também de programa proposto, é mostrado na Figura 4.37. A descrição da função de cada de cada bloco apresentado a pós o diagrama.

Fluxograma do Controlador H_{INFINITO} Auxiliado por AG

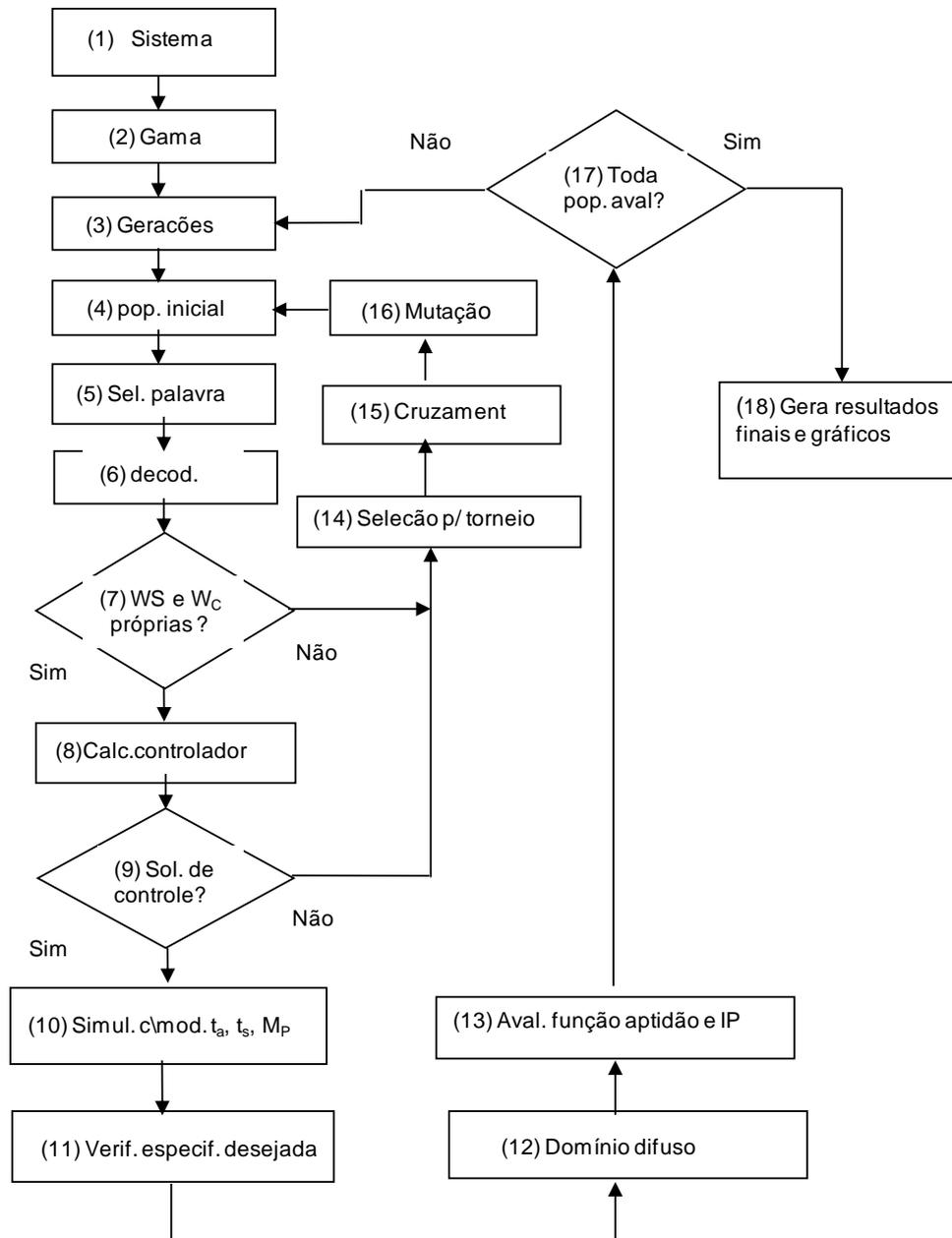


Figura 4.37: Fluxograma do Contr. H_{INF} Auxiliado AG-

Fonte: Cardenas (2005)

Descrição do Fluxograma

Bloco	Função
1	Introduz o sistema no programa;
2	Introduz o valor de gama (γ);
3	Inicia as gerações das populações;
4	Forma as populações iniciais;
5	inicia a seleção de palavras;
6	Decodifica as palavras selecionadas;
7	Verifica de se $W_s(s)$ e $W_c(s)$ são funções próprias ;
8	Usando a função <i>sysic</i> e <i>hinfsyn</i> calcula o controlador H_{INFINITO} ;
9	Verifica se ocorreu ou não saídas do controlador;
10	Usa as saídas do controlador para fazer simulação nos modelos de t_a , t_s e M_p ;
11	Compara os resultados da simulação com os parâmetros desejados;
12	Verificação do grau de pertinência dos resultados obtidos em Relação ao desejados;
13	Avaliação do grau de aptidão dos resultados e calcula um índice de desempenho;
14, 15, 16 e 17	Geram novas palavras a partir das palavras rejeitadas pelos blocos 7 e 8 e as devolvem para formar nova população inicial (bloco 4);
18	Estabelece o critério de parada (devolve as palavras não avaliadas para serem reavaliadas (bloco 3) e as avaliadas vão formar o controle final) e
19	Gera os resultados finais os apresenta na forma gráfica

Os programas, usando funções do Matlab[®], correspondentes ao fluxograma da Figura 4.37 são apresentados nos Apêndices A3, A4 e A5.

4.3.2 Controlador H_{INFINITO} original auxiliado por algoritmos genéticos

Cárdenas usou um controlador H_{INFINITO} no qual suas funções de ponderação, W_s e W_c , dadas pelas Expressões 3.47 e 3.48 e repetidas aqui, para facilitar a consulta, respectivamente pelas Expressões 4.10 e 4.11, tiveram todos os seus seis parâmetros, M_s , ω_s , δ_s , M_c , ω_{bc} e δ_c , determinados por algoritmos genéticos, conforme dito anteriormente; este controlador foi usado para controlar a atitude do satélite rígido-flexível cujas matrizes foram dadas em 4.9.

$$W_s = \frac{\frac{s}{M_s} + \omega_b}{s + \delta_s \omega_b} \quad (4.10)$$

$$W_c = \frac{s + \frac{\omega_{bc}}{M_c}}{\delta_c + \omega_{bc}} \quad (4.11)$$

O autor colocou como objetivos para este controlador, no domínio do tempo, usando uma entrada em degrau unitário, as especificações a seguir:

- tempo de subida (t_s) menor que 80 segundos;
- tempo de acomodação (t_a) menor que 500 segundos e (4.7)
- Sobre sinal ou magnitude de pico (M_p) menor que 10%.

4.3.3 Desenvolvimento do controlador H_{INFINITO} /AG proposto

Nesta seção trata-se das contribuições deste trabalho visando melhorar o programa desenvolvido por Cárdenas (2005) e alcançar o objetivo de melhoria e simplificação, citado em 4.1 que, recapitulando, são: adaptação e atualização

do programa usando funções existentes nas versões atuais do Matlab e sua adaptação para obter a simplificação na determinação dos parâmetros de W_s e W_c , tornando o programa mais rápido e reduzindo o valor de γ (gama), usado no processo iterativo para o cálculo do controlador H_{INFINITO} sub ótimo.

O programa original deixa a cargo dos algoritmos genéticos a busca pelos parâmetros M_s , ω_b e δ_b da matriz W_s e M_c , ω_{bc} e δ_c da matriz W_c .

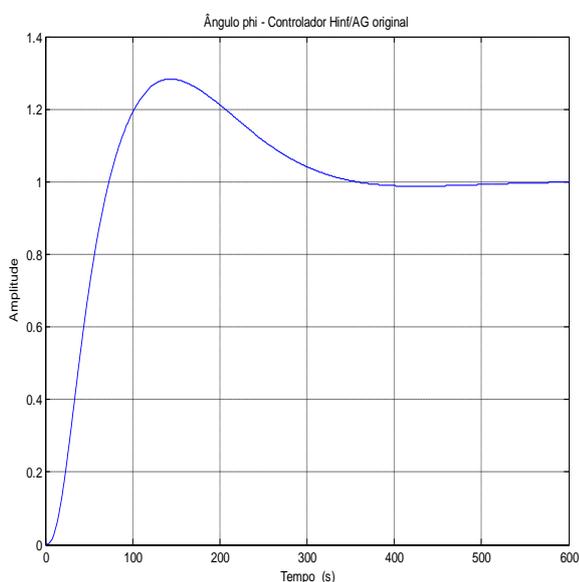
O presente trabalho atribui valores constantes para os parâmetros δ_s e δ_c , reduzindo o trabalho do AG; a escolha destes dois parâmetros deve-se ao fato de corresponderem aos erros de regime permanente das matrizes de ponderações $W_s(s)$ e $W_c(s)$ respectivamente, consideradas como funções de transferência no domínio da frequência complexa s (ZHOU, 1998; SKOGESTAD; POSTLETHWAITE, 2007). Foram escolhidos os parâmetros δ_s e δ_c por serem mais importantes que os demais parâmetros indicadores de méritos no domínio do tempo, visto serem indicadores de méritos de regime permanente, ou seja, são indicadores que são usados durante todo o tempo em que o sistema estiver em operação. Os valores de erro usualmente adotados pela literatura para expressar a precisão de resposta de um sistema, em regime permanente, a uma entrada degrau unitário é de 2% ou 5% (OGATA, 2003), dependendo da precisão exigida do controle.

Os benefícios desta contribuição são dados a seguir:

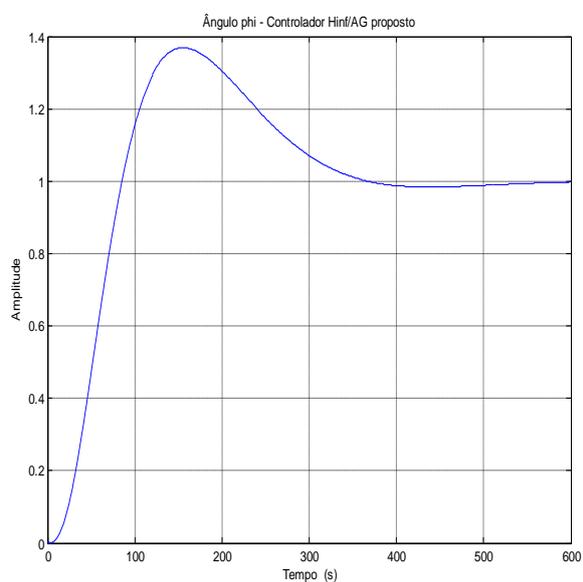
- Não deixar sob o processo de busca do AG os parâmetros δ_s e δ_c , de valores numéricos conhecidos, que devem ser determinado pelo projetista.
- Obtenção de redução do tempo execução do programa de controle e
- Redução do valor de γ (gama) do controlador H_{INFINITO}

Após rodar os programas adaptados por este trabalho, apresentados nos Apêndices A1, A2, A3, A4 e A5, obtém-se como respostas as curvas mostradas pelas Figuras 4.38b, 4.39b e 4.40b

Para facilitar as consultas aos gráficos das curvas de respostas dos controladores $H_{\text{INFINITO}}/\text{AG}$ original e simplificado eles foram colocados lado a lado e receberam um mesmo número, sendo diferenciado de seu respectivo par pelas letras *a* e *b*. Assim, as respostas de atitude do controlador $H_{\text{INFINITO}}/\text{AG}$ original do CBERS-1 são apresentadas pelas Figuras 4.38a, 4.39a e 4.40a; as Figuras 4.38b, 4.39b e 4.40b mostram as respostas de atitude do controlador $H_{\text{INFINITO}}/\text{AG}$ simplificado, apresentado neste trabalho; estas Figuras serão comentadas nas seções seguintes.

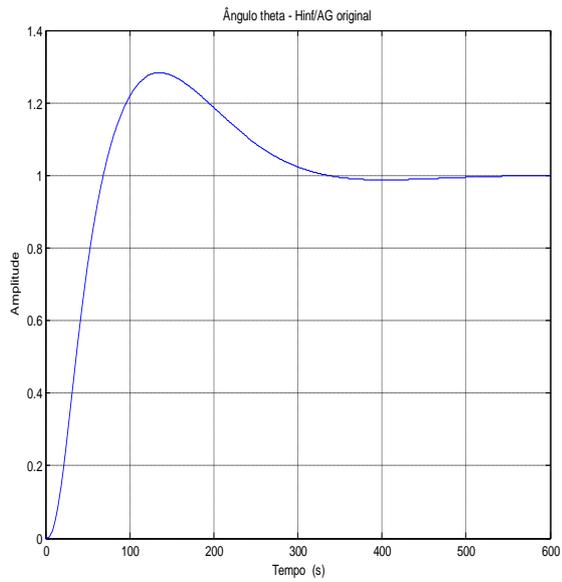


(a)

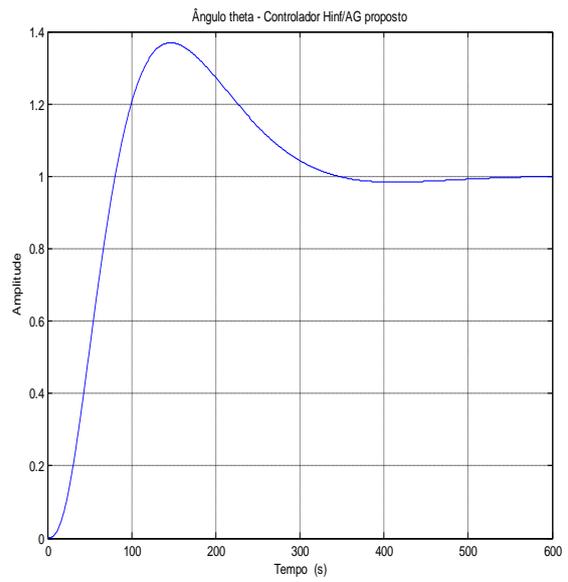


(b)

Figura 4.38a,b: Ângulo de atitude phi H_{INF}/AG original e proposto do CBERS

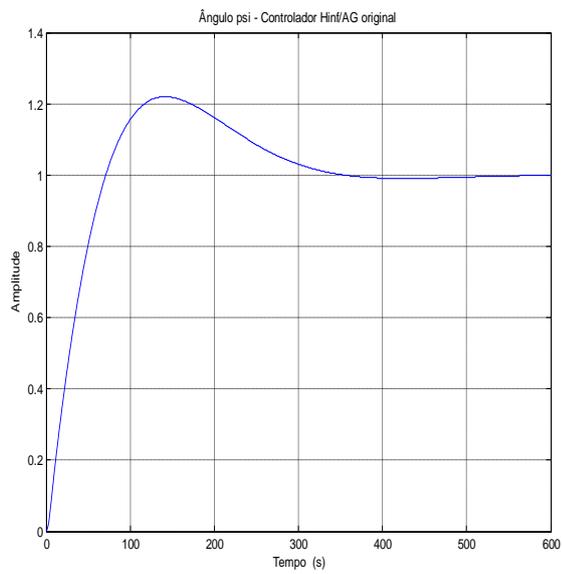


(a)

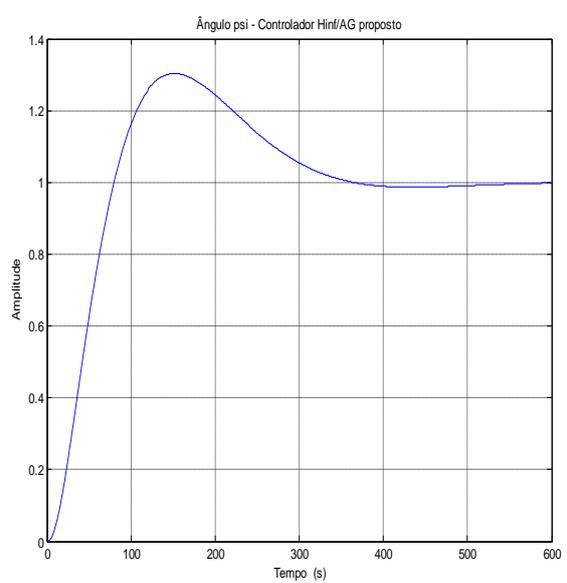


(b)

Figura 4.39a,b: Ângulo de atitude θ H_{∞} /AG original e proposto- CBERS



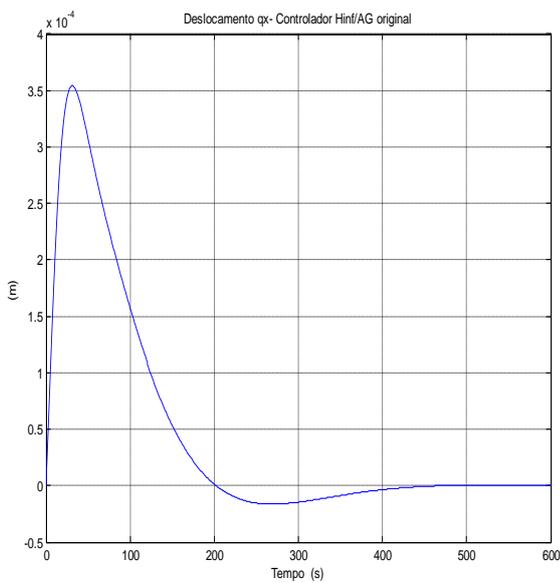
(a)



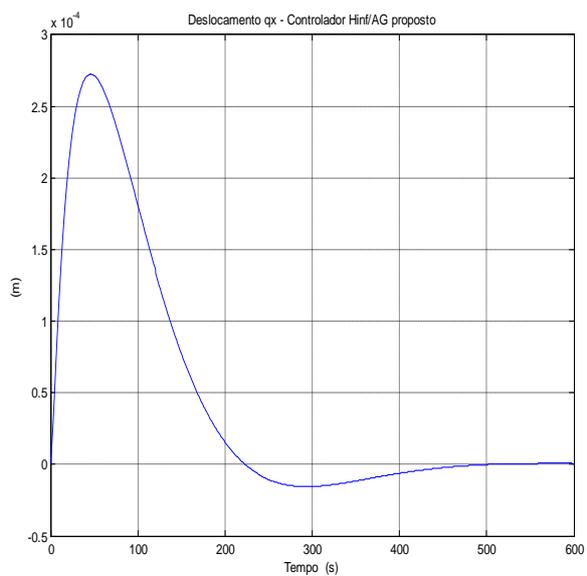
(b)

Figura 4.40a,b: Ângulo de atitude ψ H_{∞} /AG original e proposto - CBERS

Embora não sendo usado, neste trabalho, como instrumentos para comprovação e avaliação da eficiência do controlador simplificado em relação ao controlador original, apresentam-se aqui as respostas do módulo flexível do satélite, representado por seu painel solar; as figuras responsáveis por esta apresentação, a exemplo das figuras dos gráficos de atitude, foram postas lado a lado para facilitar a comparação entre elas. Estas figuras são as Figuras 4.41a,b e 4.42a,b que apresentam os deslocamentos elásticos q_y e q_z nos eixos X e Z respectivamente, dos controladores original e simplificado; o movimento no eixo Y deixou de ser controlado quando foi realizado a redução do modelo tratada na Seção 4.4.2.

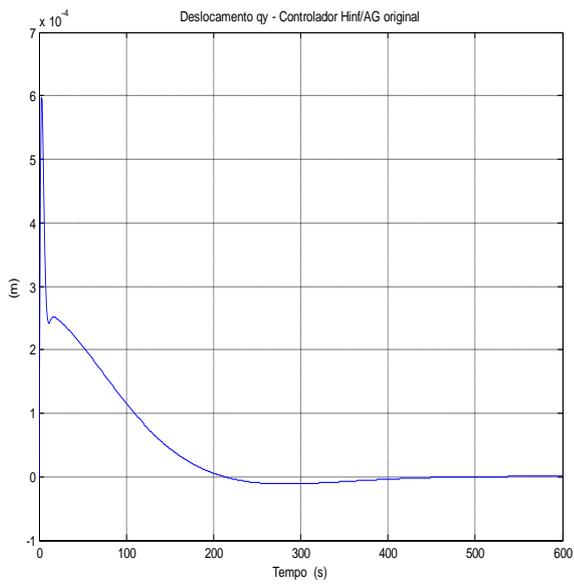


(a)

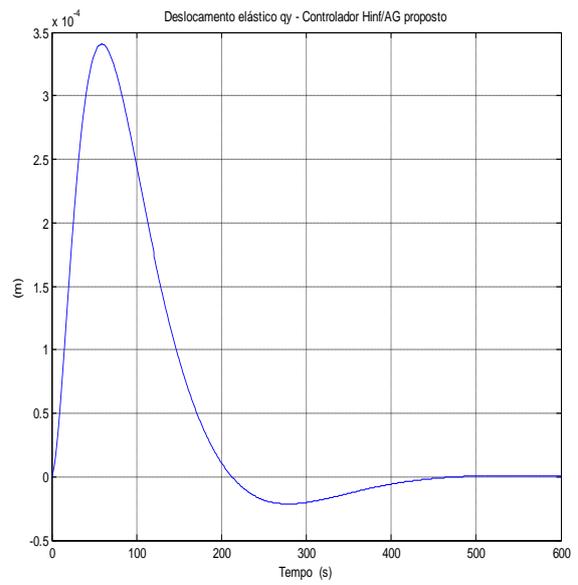


(b)

Figura 4.41a,b: Deslocamentos elásticos q_x dos controladores original e proposto do CBERS-1

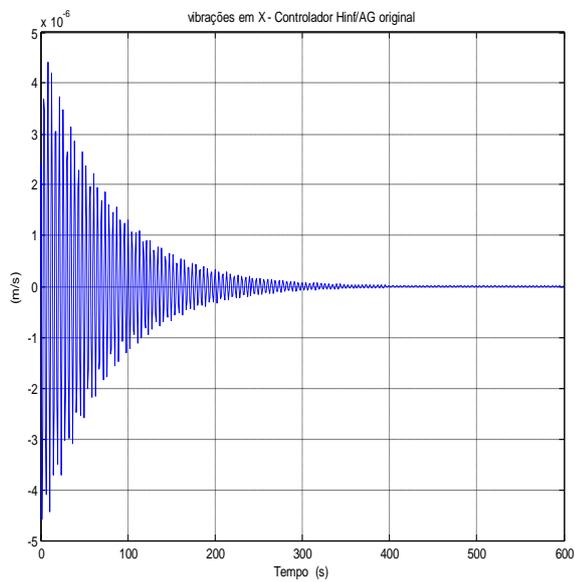


(a)

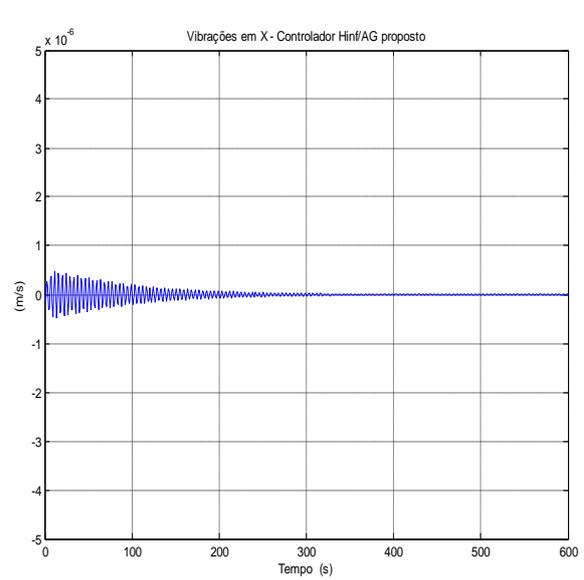


(b)Figura

Figura - 4.42a,b - Deslocamentos elásticos q_y dos controladores original e proposto do CBERS-1



(a)



(b)

Figura 4.43a,b: Vibrações do painel no eixo X nos controladores original e simplificado do CBERS-1

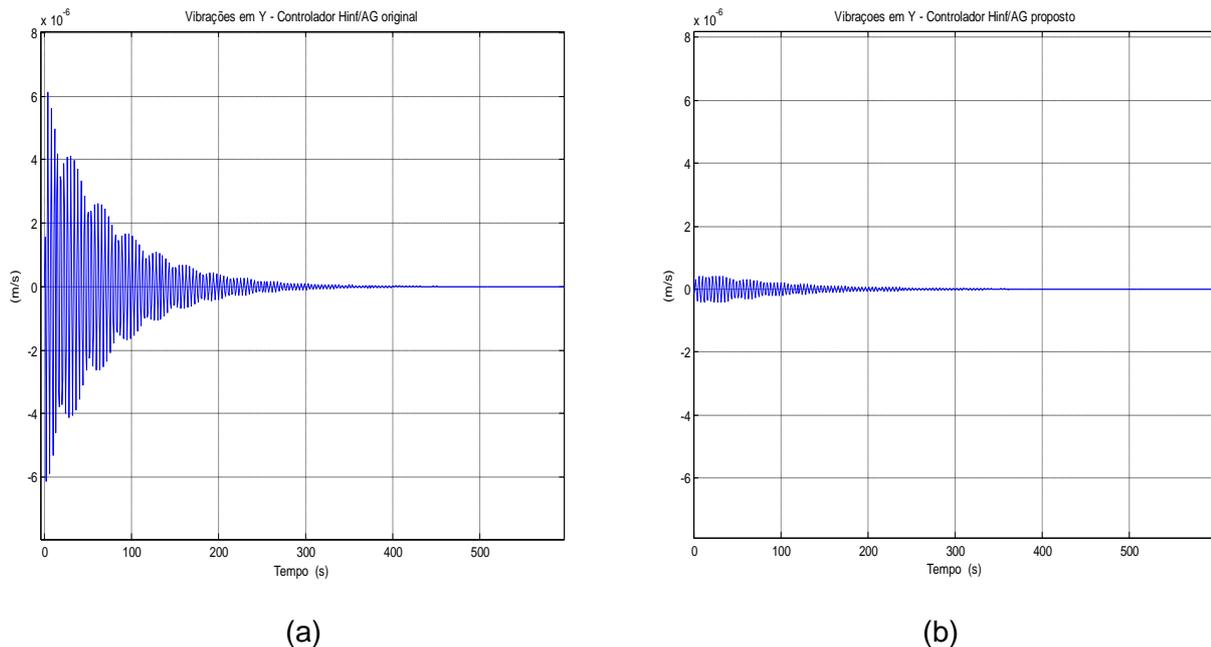


Figura 4.44a,b: Vibrações do painel no eixo Y nos controladores original e proposto

4.3.4 Robustez do controlador H_{∞} /AG proposto

Usando o sistema dado em 4.9 e as especificações no domínio do tempo dadas em 4.12 no programa H_{∞} /AG simplificado, obtiveram-se as respostas mostradas pelas Figuras 4.38b, 4.39b e 4.40b. Estas figuras mostram que o único parâmetro que não atendeu à especificação foi o sobre sinal M_p , tanto no controlador H_{∞} /AG original (Figuras 4.38a, 4.39a e 4.40a) como no controlador H_{∞} /AG simplificado (Figuras 4.38b, 4.39b e 4.40b). Contudo, se o objetivo mais relevante do controle for o rastreamento do sinal de entrada, o controlador H_{∞} /AG simplificado obtido pode ser aceito como razoável, considerando que o sistema a ser controlado neste trabalho é relativamente lento. O sobre sinal M_p poderia, se necessário, ser melhorado, pesquisando-se uma estrutura mais complexa para as funções de ponderação W_s e W_c , até obter um sobre sinal aceitável; a pesquisa destas funções é uma atividade laboriosa que, para o objetivo deste trabalho, não se fez necessária.

Observa-se nas Figuras 4.38a,b, 4.39a,b e 4.40a,b que as curvas de respostas dos dois controladores são muito semelhantes; o controlador gerado pelo programa simplificado ficou ligeiramente mais lento que o controlador gerado pelo programa original (cerca de 3% mais lento), fato que não constitui uma inconveniência para o controle final, visto que os tempos de subida (t_s) e de acomodação (t_a) atenderam às especificações iniciais dadas por 4.7

Foram realizadas algumas execuções preliminares do programa H_{INFINITO} simplificado e usando os resultados obtidos nestas execuções obteve-se as funções de ponderação \mathbf{W}_s , \mathbf{W}_c , a matriz do sistema $\mathbf{G}_{10 \times 10}$ e uma matriz de ganho \mathbf{k}_k da qual extraiu-se uma matriz reduzida de ganho que permitiram construir o gráfico mostrado na Figuras 4.45. Este gráfico mostra que na faixa de frequência de 10^{-6} a 8×10^{-2} rad/s, o controlador é robusto, visto que as expressões 4.13 , dadas a seguir, são respeitadas.

$$|S| < \frac{1}{|W_s(s)|} \quad \text{e} \quad |T| < \frac{1}{|W_c(s)|} \quad (4.13)$$

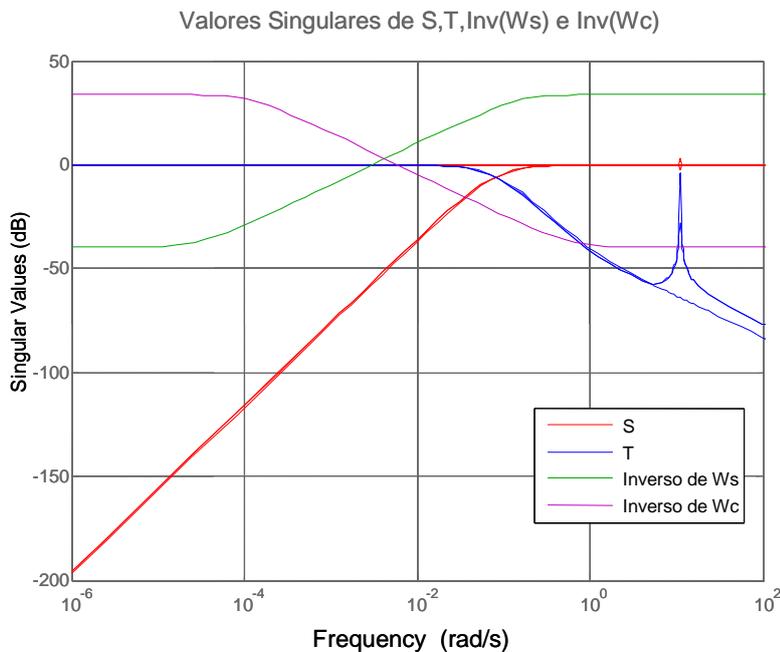


Figura 4.45: Curvas de S, $1/|W_s|$, T e $1/|W_c|$

4.3.5 Ganhos obtidos com o programa simplificado

Um dos méritos do programa proposto, apresentado por este trabalho, é evitar que o processo de busca do AG se estenda sobre os parâmetros δ_s (deltaS) e δ_c (deltaC), que são parâmetros conhecidos na literatura de controle (OGATA, 2003) cuja determinação é de responsabilidade do projetista do sistema de controle, conforme já mencionado anteriormente; o resultado imediato desta ação é a redução no tempo de execução do programa (maior rapidez de controle).

Os ganhos resultantes da simplificação tratada no parágrafo anterior são confirmados pelas Tabelas 4.4 e 4.5; estas tabelas foram obtidas com os resultados de trinta execuções de cada programa (original e proposto).

Através da observação dos resultados de cada execução e do desvio padrão de cada parâmetro, verifica-se a relativa estabilidade funcional dos dois programas.

Tabela 4.4- Execuções do programa original

Execução	Gama	PI	Ws1	Ms1	Deltas	Wc1	Mc1	Deltac	Tempo s
1	330,05	0,6997	1,9608	1	0,055	50	2	0,01	80,8
2	330,05	0,7467	50	1	0,055	49,99	2	0,01	78,9
3	330,6	0,7117	50	1	0,055	48,82	0,8	0,055	81,1
4	330,05	0,6997	50	0,4998	0,055	50	0,7995	0,01	80,1
5	330,6	0,6997	1,9608	1	0,055	48,83	0,7995	0,01	80,1
6	330,5	0,6997	50	0,999	0,055	1,959	2	0,055	83,9
7	330,6	0,6997	1,9589	0,4998	0,055	50	2	0,01	80,5
8	330,7	0,6997	49,99	1	0,055	50	2	0,01	82,8
9	330,05	0,7277	1,9589	0,4998	0,055	48,8	2	0,055	83,4
10	330,6	0,6997	50	0,5	0,055	1,95	2	0,055	84,6
11	330,5	0,7467	50	0,4998	0,055	1,9589	0,7995	0,01	81,2
12	330,5	0,6997	50	1	0,055	1,9608	2	0,055	85,9
13	330,5	0,6996	48,83	0,5	0,055	50	2	0,01	84,9
14	330,6	0,7227	50	0,4898	0,055	1,9608	0,8	0,01	83,6
15	330,5	0,6997	50	1	0,055	49,99	0,8	0,01	80,5
16	330,6	0,6997	50	1	0,055	50	2	0,01	85,6
17	330,6	0,6997	48,8	1	0,055	50	2	0,01	84,7
18	330,5	0,6997	50	1	0,055	50	1,99	0,01	80,1
19	330,5	0,6997	48,83	0,4998	0,055	49,99	2	0,055	78,9
20	330,6	0,6997	50	1	0,055	1,9608	2	0,055	86,3
21	330,05	0,6997	1,9608	1	0,055	50	2	0,01	80,8
22	330,05	0,7467	50	1	0,055	49,99	2	0,01	78,9
23	330,6	0,7117	50	1	0,055	48,82	0,8	0,055	81,1
24	330,05	0,6997	50	0,4998	0,055	50	0,7995	0,01	80,1
25	330,6	0,6997	1,9608	1	0,055	48,83	0,7995	0,01	80,1
26	330,6	0,6997	50	1	0,055	50	2	0,01	85,6
27	330,6	0,6997	48,8	1	0,055	50	2	0,01	84,7
28	330,5	0,6997	50	1	0,055	50	1,99	0,01	80,1
29	330,5	0,6997	48,83	0,4998	0,055	49,99	2	0,055	78,9
30	330,6	0,6997	50	1	0,055	1,9608	2	0,055	86,3
DesvPad	0,223364	0,014916	19,1221	0,2362	1,4E-17	20,2142	0,549605	0,021213	2,4879
Media	330,445	0,706897	40,1947	0,8329	0,055	38,592	1,63925	0,025	82,15

Tabela 4.5- Execuções do programa simplificado

Execução	Gama	IP	Ws1	Ms1	Deltas	Wc1	Mc1	Deltac	Tempo
1	100	0,74		1	0,01	50		0,01	61,1
2	100	0,74	48,8	0,999	0,01	50	2	0,01	61,6
3	100	0,74	50	1	0,01	49,9	2	0,01	61,7
4	100	0,74	50	1	0,01	50	2	0,01	62,1
5	100	0,75	50	0,49	0,01	50	1,8	0,01	60,7
6	99	0,74	50	1	0,01	50	2	0,01	62,1
7	100	0,76	50	1	0,01	50	2	0,01	61,5
8	100	0,744	49,9	0,499	0,055	50	2	0,01	61,7
9	98	0,766	48,8	0,99	0,01	50	2	0,01	61,2
10	100	0,76	50	1	0,01	48,8	2	0,01	61,2
11	100	0,76	50	1	0,01	49,9	2	0,01	61,1
12	100	0,76	50	1	0,01	50	2	0,01	60,9
13	99	0,76	50	1	0,01	50	0,8	0,01	61,1
14	99	0,76	50	1	0,01	50	2	0,01	61,1
15	100	0,76	49,9	0,5	0,055	49,9	2	0,01	61,7
16	100	0,76	50	1	0,055	50	2	0,01	61,1
17	98	0,74	50	1	0,01	50	2	0,01	62,7
18	100	0,74	48,8	0,999	0,01	50	2	0,01	61,6
19	100	0,74	50	1	0,01	49,9	2	0,01	61,7
20	100	0,74	50	1	0,01	50	2	0,01	62,1
21	99	0,74	50	1	0,01	50	2	0,01	62,1
22	100	0,76	50	1	0,01	50	2	0,01	61,5
23	100	0,744	49,9	0,499	0,055	50	1,8	0,01	61,7
24	98	0,766	48,8	0,99	0,01	50	2	0,01	61,2
25	100	0,76	50	1	0,01	50	2	0,01	60,9
26	99	0,76	50	1	0,01	50	0,8	0,01	61,1
27	99	0,76	50	1	0,01	50	2	0,01	61,1
28	100	0,76	49,9	0,5	0,055	49,9	2	0,01	61,7
29	98	0,74	50	1	0,01	50	2	0,01	62,7
30	100	0,74	48,8	0,999	0,01	50	2	0,01	61,6
DesvPadr	0,718	0,01009	0,4483	0,1869	0,01677	0,2155	0,3045	3,5E-18	0,494
Média	99,53	0,751	49,779	0,9155	0,0175	49,943	1,9034	0,01	61,52

Das Tabelas 4.4 e 4.5 foram extraídas as médias do parâmetro γ (gama), tempo gasto por cada execução de cada um dos programas e o índice de desempenho de cada execução (IP); com estes dados foi montada a Tabela 4.6 a seguir. O parâmetro γ indica o número de iterações usadas no cálculo para a obtenção de um controlador H_{INFINITO} sub ótimo.

Tabela 4.6- Valores médios de gama, tempo de execução e desempenho IP

	Controlador original	Controlador proposto
γ (gama)	330,34	99,55
Tempo (s)	82,26	61,50
Índice de desempenho (IP)	0,707	0,751

Da Tabela 4.6, observa-se uma redução de cerca de 25 % no tempo de execução e cerca de 70% na redução do valor de gama; o índice de desempenho (IP), permaneceu praticamente constante

O tempo médio gasto para executar o programa do controlador $H_{\text{INFINITO}}/\text{AG}$ original, usando um computador Intel[®] T5750 de 1,7 GHz, foi de 61,5 segundos; o tempo médio gasto por esta mesma máquina para executar o programa do controlador proposto foi de 61,50 segundos.

Com os dados das Tabelas 4.4 e 4.5, foram plotados as curvas mostradas pelas Figuras 4.46a,b,c e 4.49a,b,c respectivamente, dadas a seguir, que visualizam as variações dos parâmetros gama, tempo de execução e índice de desempenho dos controladores $H_{\text{INFINITO}}/\text{AG}$ original e proposto, resumidos na Tabela 4.6.

Da observação da Tabela 4.6 e das comparações entre os parâmetros das Figuras 4.46a,b,c e 4.49a,b,c vê-se que o controlador $H_{\text{INFINITO}}/\text{AG}$ proposto nesta tese apresenta melhoria em relação ao controlador $H_{\text{INFINITO}}/\text{original}$.

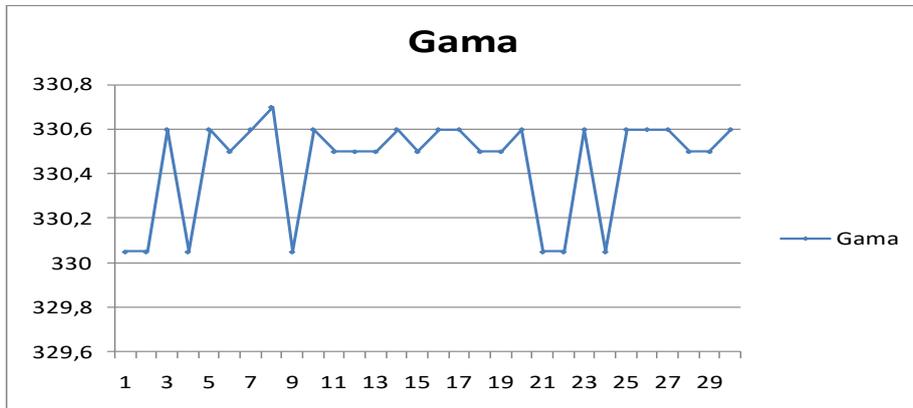


Figura 4.46a - Variação do número de iteração gama (γ) do controlador $H_{\text{INFINITO}}/\text{AG}$ original

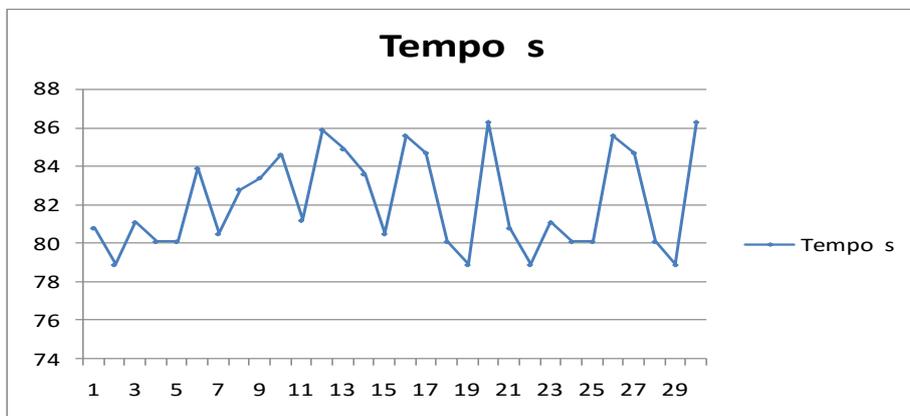


Figura 4.47b – Variação do tempo de execução do controlador $H_{\text{INFINITO}}/\text{AG}$ original

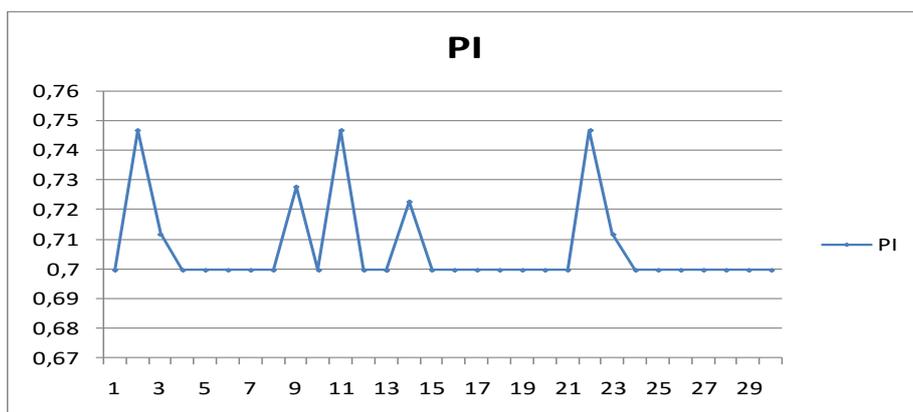


Figura 48c - Variação do índice de desempenho do controlador $H_{\text{INFINITO}}/\text{AG}$ original

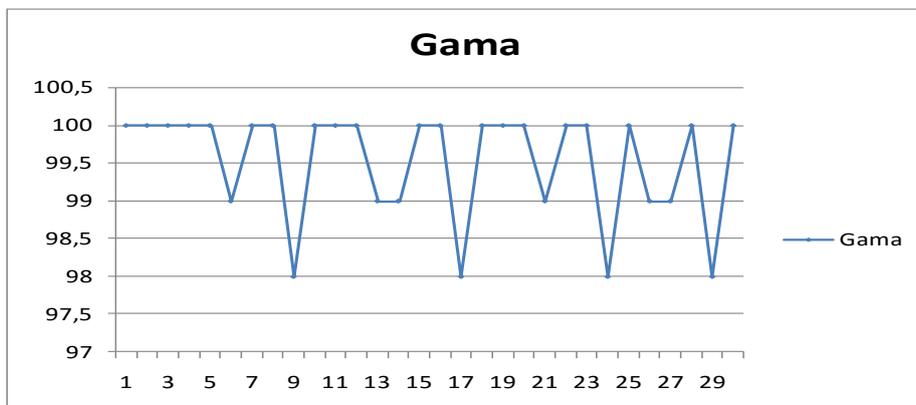


Figura 4.49a – Variação do número de iteração gama (γ) do controlador $H_{\text{INFINITO}}/\text{AG}$ proposto

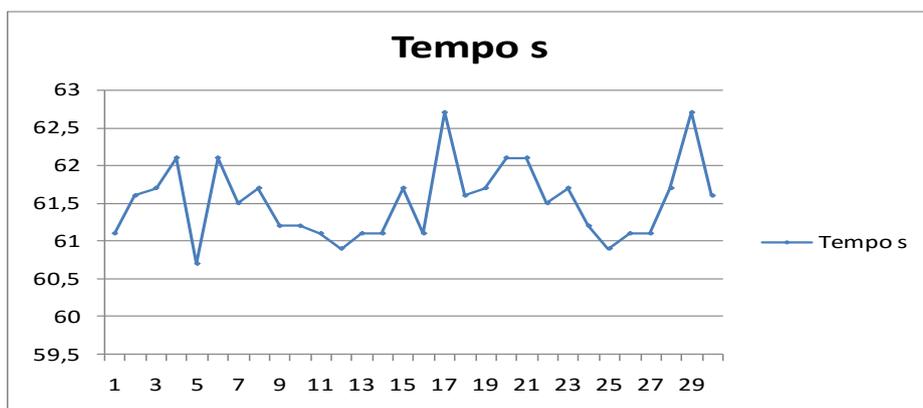


Figura 4.50b – Variação do tempo de execução do controlador $H_{\text{INFINITO}}/\text{AG}$ proposto

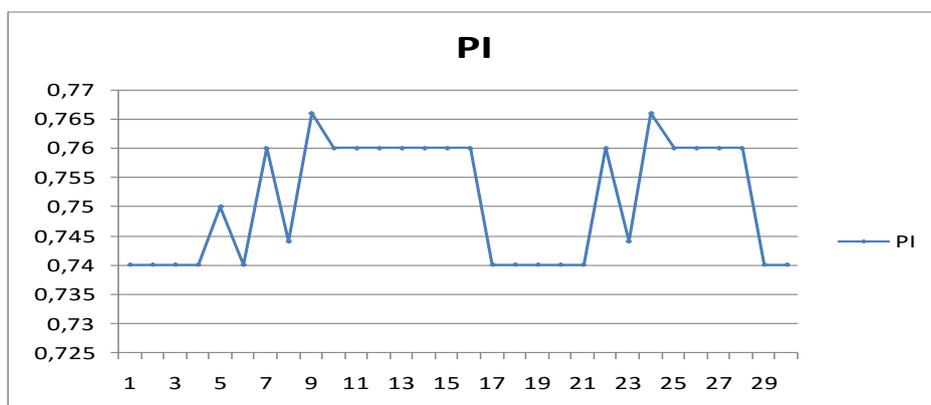


Figura 4.51c - Variação do índice de desempenho do controlador $H_{\text{INFINITO}}/\text{AG}$ proposto

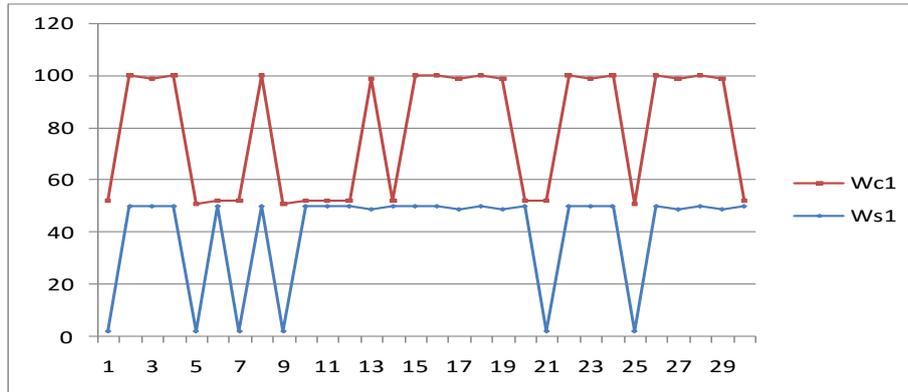


Figura 4.52 – Variação de W_{S1} e W_{C1} do controlador $H_{HINFINITO}/AG$ original

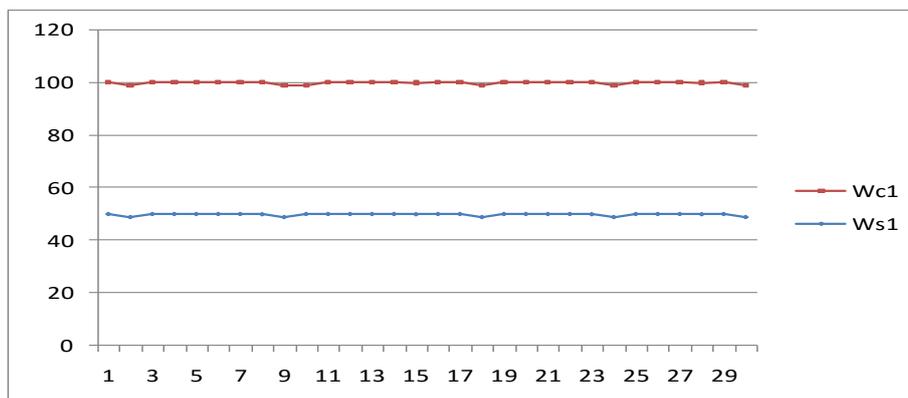


Figura 4.53 – Variação de W_{S1} e W_{C1} do controlador $H_{HINFINITO}/AG$ proposto

Comparando-se as curvas de W_{S1} e W_{C1} das Figuras 4.52 e 3.53 respectivamente, observa-se as curvas de W_{S1} e W_{C1} da Figura 4.53, correspondente ao controlador $H_{HINFINITO}/AG$ proposto, são mais regulares que suas correspondentes da Figura 4.52 correspondente ao controlador $H_{HINFINITO}/AG$ original; esta regularidade é consequência da fixação de δ_S e δ_C pelo projetista e é a responsável pela maior rapidez de execução do programa do programa do controlador $H_{HINFINITO}/AG$ proposto em relação ao programa do controlador $H_{HINFINITO}/AG$ original.

Lembra-se W_{S1} e W_{C1} são valores numéricos de W_S e W_C respectivamente dados pelas expressões 4.10 e 4.11.

Das respostas gráficas e numéricas apresentadas anteriormente conclui-se que os objetivos propostos nesta seção foram alcançados.

Especificamente em relação ao controlador $H_{\text{INFINITO}}/\text{AG}$ proposto, viu-se que, com relação ao controle de atitude, este controlador apresentou praticamente os mesmos resultados gráficos, com ligeira redução do sobre sinal para os ângulos theta (θ) e psi (ψ).

Para o deslocamento elástico em Y (q_y) que é a deflexão na extremidade do painel, o controle $H_{\text{INFINITO}}/\text{AG}$ simplificado apresentou um menor deslocamento em relação ao $H_{\text{INFINITO}}/\text{AG}$ original; o deslocamento elástico em Z (q_z) do controlador proposto teve um amortecimento mais lento em relação ao controlador original, contudo atendeu a especificação proposta (tempo de acomodação menor que 500 segundos).

Com relação às vibrações do painel flexível, o controlador $H_{\text{INFINITO}}/\text{AG}$ proposto produziu menor vibrações no eixo X que o controlador $H_{\text{INFINITO}}/\text{AG}$ original; no eixo Z as vibrações produzidas pelo controle proposto apresentaram um pequeno retardo no tempo, só amortecendo após seiscentos segundos; isto não compromete o controle como um todo por se tratar de vibrações de pequenas amplitudes (da ordem de 10^{-7} m).

Finalmente, o resumo apresentado acima e os ganhos apresentados na Tabela 4.6, mostram que a contribuição de melhoria para o programa que gera o controlador $H_{\text{INFINITO}}/\text{AG}$ original de Cardenas, o último objetivo desta tese, obteve sucesso.

5 CONCLUSÃO E COMENTÁRIOS

Destaca-se neste capítulo as principais atividades desenvolvidas neste trabalho, com o objetivo de usar técnicas *Fuzzy* e algoritmos genéticos para otimizar os controladores LQR, SDRE e H_{INFINITO} .

Foram revistos os modelos matemáticos linear e não linear do simulador de satélite construído pelo INPE e o modelo linear de um dispositivo formado por uma haste flexível engastada em um cubo; ambos os modelos representando, respectivamente um satélite rígido e rígido-flexível.

Foram também revistos os fundamentos teóricos básicos das técnicas de controle utilizadas neste trabalho, ou seja, alocação de pólos, controlador LQR, controlador SDRE, controlador H_{INFINITO} e algoritmos genéticos.

Para atender aos objetivos deste trabalho que são a fusão de lógica *Fuzzy* e/ou algoritmos genéticos com as técnicas clássicas de controle LQR, SDRE, e H_{INFINITO} , inicialmente foi aplicado sobre o modelo linear do simulador de satélite um controle por realimentação de estado, cujo ganho de controle foi obtido por alocação de pólos. Os resultados deste controle foram usados para base de comparação para os resultados obtidos com o controlador LQR, cuja matriz de ponderação de estado \mathbf{Q} foi obtida por lógica *Fuzzy*; A matriz de ponderação de controle \mathbf{R} foi considerada unitária.

Em seguida aplicou-se sobre o modelo do simulador um controlador SDRE, implementado pelo programa Simulink/matlab e foram usados os resultados deste controle para servir de base de comparação para os resultados produzidos pelo controlador SDRE, cuja matriz \mathbf{Q} foi determinada pela lógica *Fuzzy*; usou-se para implementar este controle híbrido o diagrama Simulink do controle anterior tratado acima modificado para esta nova realização.

Seguindo a metodologia adotada, para o sistema rígido-flexível usou-se os resultados fornecidos pelo controlador LQR clássico aplicado sobre o modelo fornecido pelo fabricante (Quanser[®]) do sistema rígido-flexível (SVR2), para

servir de base de comparação para os resultados obtidos pelo controlador LQR/*Fuzzy* desenvolvido neste trabalho.

Finalmente, na seção 4.3.5 desta tese são apresentados os resultados da aplicação de algoritmos genéticos (AG) para obter os parâmetros M_S , W_S , M_C , e W_C do controlador $H_{\text{HINFINITO}}/\text{AG}$. O controlador $H_{\text{HINFINITO}}/\text{AG}$ proposto, com relação ao controle dos ângulos de atitude e vibrações, apresentou praticamente os mesmos resultados do controlador $H_{\text{HINFINITO}}/\text{AG}$ original; obteve-se significativas melhoria no esforço de controle e tempo de execução do controle, conforme mostrado pela tabela 4.6.

O resumo das contribuições deste trabalho são:

1)-Aplicação direta e com êxito da lógica *Fuzzy* para determinar a matriz de ponderação de estado \mathbf{Q} controlador LQR;

2)-Foram obtidos bons resultados aplicando-se a lógica *Fuzzy* para a determinação da matriz \mathbf{Q} de um controlador SDRE; a matriz \mathbf{R} foi mantida unitária.

3)-Comparando-se os resultados dos controladores otimizados pela técnica *Fuzzy* com os controladores clássicos originais, os controladores otimizados apresentam as características dadas abaixo como vantagens em relação aos controladores originais:

3.1 Maior suavidade na ação de controle, reduzindo possíveis vibrações a cada ação de controle conforme Figuras 4.29ab, 4.30ab e 4.31ab;

3.2 Controle sem muita oscilação, ou seja, a saída de controle vai para o ponto de equilíbrio sem oscilar;

3.3 Melhor controle do controlador SDRE/*Fuzzy* sobre o simulador de satélite em relação ao controle SDRE clássico conforme Figuras 4.29ab, 4.39ab e 4.31ab.

3.4 Foi verificado em exaustivas simulações que o controle LQR/*Fuzzy* é robusto para entrada degrau unitário, mesmo na presença de grandes sinais.

Como desvantagem, no caso deste trabalho, aponta-se o fato de ser de 2 a 4% mais lento que o controlador convencional (é o caso o controlador LQR/*Fuzzy* sobre o simulador de satélite);

4)-Do ponto de vista de projeto, importante vantagem em relação aos controladores originais (LQR e SDRE) é a determinação automática da matriz de ponderação Q e

5)-No caso do controlador auxiliado por algoritmos genéticos, o controlador $H_{\text{INFINITO}}/\text{AG}$ proposto apresentou maior rapidez de execução (controle mais rápido), melhor índice de desempenho IP e menor valor de γ (gama) que o controlador $H_{\text{INFINITO}}/\text{AG}$ original.

Sugestões para futuros trabalhos

Como sugestões para a continuação deste trabalho e trabalhos futuros, apresenta-se:

1)-Estudar os efeitos sobre a precisão dos resultados, da elaboração de regras de controle *Fuzzy* mais complexa, usando funções de pertinências não triangulares ;

2)-Estudar, através de simulação, o controle de sistemas não lineares;

3)-Pesquisar funções de ponderações $W_S(s)$ e $W_C(s)$ mais complexas visando obter menor sobre pico (*overshoot*) para as respostas entradas do tipo degrau unitário e

4)- Pesquisar uma função objetiva mais simples para o caso do controlador $H_{\text{INFINITO}}/\text{AG}$ original desenvolvido por Cárdenas (CÁRDENAS, 2005) e o controlador $H_{\text{INFINITO}}/\text{AG}$ proposto neste trabalho

REFERÊNCIAS BIBLIOGRÁFICAS

- ANDERSON, B.D.O.; MOORE, J.B. **Optmal control**: linear quadratic methods. New York: Dove Publications, 2007.
- ASTRÖM, K.; HÄGGLUND, T. PID controller, theory, design and tuning. Research Triangle Park: Instrument Society of America, 1995.
- BARBOSA, H. J. C. Algoritmos genéticos para otimização em engenharia: uma introdução. In: SEMINÁRIO SOBRE ELEMENTOS FINITOS E MÉTODOS NUMÉRICOS EM ENGENHARIA, 4., 1996, Juiz de Fora. **Anais...** Juiz de Fora: Universidade Federal de Juiz de fora, 1996.
- BERENJI, H. R.; LEA, R.; JANI, Y.; KHEDKAR, P.; MALKANI, A.; HOBLIT, J. Space shuttle attitude control by reinforcement learning and *Fuzzy* logic. In: IEEE ITERNATIONAL CONFERENCE ON *FUZZY* LOGIC SYSTEMS, 2., 1993, San Francisco. **Proceedings...** San Francisco: IEEE, 1993.
- CARDENAS,P.R.Y.A. **Modelagem e controle de atitude de satélites artificiais com apêndices flexíveis**. Tese (Doutorado em Engenharia Elétrica) - Escola Politécnica, USP, SP, 2005;
- CAMPA,G. **Algebraic riccati equation solution in simulink via C+ Fortran, matlab control file-LqrSim**. [S.l.]: The MathWorks, Inc., 2002.
- CARRARA, V. **Cinemática e dinâmica de atitude de satélites artificiais**. São José dos Campos: INPE, 2012. 111 p. (sid.inpe.br/mtc-m19/2012/01.26.19.13-PUD). Disponível em: <<http://urlib.net/8JMKD3MGP7W/3B96GD8>>. Acesso em: 15 maio 2015.
- CHIANG, R.Y.; JANG, J. S. R. *Fuzzy* logic attitude control for cassini spacecraft. In: IEEE WORLD CONGRESS ON COMPUTATION INTELLIGENCE, 3., 1994, Orlando, FL, USA. **Proceedings...** Orlando: IEEE, 1994. p.1532-1537.
- CLOUTIER, J. R. **State-dependent riccati equation**: an overview. In: I, 1997, Albuquerque. **Proceedings...** Albuquerque: IEEE, 1997.

CROUCH, P. E. Spacecraft attitude control and stabilization: application of geometric control theory to rigid body models.. **IEEE Transaction on Automatic Control**, v. 29, n. 4, p.87-95, 1994.

CASTRO, R. E. Otimização com multi objetivos via algoritmos genéticos. Tese (Doutorado em Engenharia Civil)-Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2001.

CASTRO, L. N.; VON ZUBEN, F.J. **Algoritmos genéticos**. Campinas: DCA/FEEC/Unicamp, 2002;

DAFTA,T.K. **A state-of-the-art review on active control of structures**. **Journal of Earthquake Technology**, v. 40, n. 1, Mar 2003, p. 1-17, 2003.

DAVIS, L. Job-shop scheduling with genetic algorithms. In: INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, 1., 1985, July 1985, Pittsburgh, PA, USA. Proceedings... Pittsburgh, PA: Lawrence Erlbaum Associates, 1985. p. 136-140.

DE JONG, K.A. **An analysis of behavior of a class of genetic adaptive systems**. Doctoral dissertation, University of Michigan, 1975.

DE GOLDBERG; KUO, C.H. Genetic algorithms in pipeline optimization. ASCE, **Journal of Computing in Civil Engineering**,l(2), p. 128-141, 1987.

DORUK, R.O. **Nonlinear control designs for reaction wheel actuated observatory satellite**. 2008. Thesis (Doctor of Philosophy in Electrical and Electronics Engineering) - Middle Est Technical University, Ankara, [Turkey](#), 2008.

ERDEM, E.B. **Analysis and real time implementation of state-dependent Riccati equation controlled systems**. University of Illinois at Urbana-Champaign, 2001. 280 p.

FENILI, A.; SOUZA, L. C. G. Control of a Nonlinear Slewing Flexible Beam. In: INTERNATIONAL CONGRES OF THEORETICAL AND APPLIED MECHANICS, 21ST., Warsaw - Poland. **Proceedings...** 2004. p. 2. On-line. ISBN 83-89697-01-1. (INPE-12147-PRE/7489).

FRANKLIN, G.F.; POWELL, J. D.; EMAMI-NAEINI, A. **Feedback control of dynamic systems**. [S,I.]: Addison-Wesley, 1986.

FOSSSEN, T. I. **Mathematical models for control of aircraft and satellites**. Department of Engineering Cybernetics. Universidade de Ciência e Tecnologia da Noruega, 2011. Lecture Notes.

FONSECA, I. M. S. **Mecânica analítica**. São José dos Campos: INPE, 2011. Notas de aulas da disciplina.

FONSECA, J. B. S. **Relatório final de projeto de iniciação científica**. São José dos Campos: INPE.

FOURMAN, M. P. Compaction of symbolic layout using genetic algorithms.

In: INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, 1., 1985, Hillsdale. NJ, USA **Proceedings...** Hillsdale: L. Erlbaum Associates Inc.. 1985. p.141-153.

FRIEDLAND, B. **Control system design**: an introduction to state space methods. [S.I.]: McGraw Hill, 1986.

GOLDBERG, D.E.; LINGLE, R. Alleles, Loci and Travelling Salesman Problem" In: INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, 1., 1985, Hillsdale. NJ, USA **Proceedings...** Hillsdale: L. Erlbaum Associates Inc.. 1985, p. 154-159.

GOLDBERG, D. E.; KRISHNAKUMAR, K. Control system optimization using genetic algorithms. **Journal of guidance Control and Dynamics**, v. 15, n. 3, p. 735-740, 1992.

GOMIDE, R.; GUDWIN, M.L.; ANDRADE NETTO, M.L. **Controle por Lógica Fuzzy**. Campinas: UNICAMP, 1995.

GONZALES, R. G. **Utilização dos métodos SDRE e filtro de Kalman para o controle de atitude de simuladores de satélites**. 2009. 120 p. (INPE-15691-TDI/1465). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2009. Disponível em: <<http://urlib.net/8JMKD3MGP8W/354F2L5>>. Acesso em: 20 maio 2015.

HUGUES, P.C. **Spacecraft attitude control**. [S.I.]: John Wiley and Sons, 1986

HOLLAND, J. Genetic algorithms and optimal allocation of trials. **SIAM, Journal of Computation**, p. 88-105, 1973. DOI: 10.1137/0202009.

JUNKINS, J. L.; TURNER, J. D. **Optimal spacecraft rotational maneuvers.** New York: Elsevier Science Publishers B. V.

JUNKINS, J.; YODAN, K. **Introduction to dynamics and control of flexible structures.** [S.I.]: Amer Inst of Aeronautics, 1983. AIAA Educational Series.

JONG, H. D. Modeling and simulation of genetic regulatory systems: literature review. **Journal of Computation**, n. 9, p.67-103, 2002.

KIM, B.J.; LEE, H.; CHOI, S.D. **Three axes reaction wheel attitude control system for kitsat-3 microsatellite.** Taejon, Korea: Satellite Technology Research Center, , 303-701, Korea,1996;

KWAKERNAAK, H.; SILVAN, R. **Linear optimal control systems.** Netherlands: Twente University of Technology, 1976.

KAPLAN, M.H. **Modern spacecraft dynamics and control.** New York: John Wiley and Sons, 1976;

KANE, T. R.; LIKINS, P.W.; LEVNSON, D. A. **Spacecraft dynamics.** [S.I.]: Mc Graw-Hill Inc, 1983.

KIKUCHI, S.; TOMINAGA, D.; ARITA, M.; TAKAHASHI, K.; TOMITA, M. Dynamic modeling of genetic networks using genetic algorithms and S-system. **Bioinformatics**, v. 19, n. 5, ,2003.

KONSTANTINOV, M. M.; PETKOV, P.H.; GU, D.-W. **Robust control design with Matlab.** 2005; ISBN: 978-1-4471-4681-0 (Print) 978-1-4471-4682-7 (Online).

KUIPERS, I.S. **Quaternions and rotation sequences.** Princeton: Princeton University Press, 1998.

KRISHNAKUMAR, K. GOLDBEERG, D.E. " Control system optimization using genetic algorithms. **Journal of Guidance, Control, and Dynamics**, v. 15, n. 3 p. 735-740, 1992. doi: 10.2514/3.20898.

KLIR, G.J.; BO, Y. **Fuzzy set and Fuzzy logic: theory and applications.** Prentice-Hall, 1995.

KATSEV, S. **Streamlining of state-dependent riccati equation controller algorithm for an embedded implementation.** M.S. Dissertation - Computer Engineering Dept., Rochester Univ., New York, Nov, 2006.

LINDEN, R. **Algoritmos genéticos**. 3. Rio de Janeiro: Ed. Editora Moderna Ltda., 2012.

LEE, C.C. *Fuzzy logic control system*. **IEEE Transactions on Systems, Man and Cybernetics**, v 20, n. 2, p. 404 – 418, 1990.

MENG,Q., ZHANG, T. Fault tolerant attitude control for flexible satellite with uncertainties and actuation saturation. **International journal of advanced robotic systems**. v. 10, Mar 2013.

MENON, P. K.; LAM,T.; CROWFORD, L. S.; V.H.L. AND CHENG, V. H. L. Real-time computational control of missiles. In: AMERICAN CONTROL CONFERENCE, 2002, Anchorage, Alaska. Proceedings... Anchorage, Alaska: IEEE, 2002. p. 232 – 237. v.1. DOI: [10.1109/ACC.2002.1024809](https://doi.org/10.1109/ACC.2002.1024809).

MACIEJOWIKI, J. **Multivariable feedback design**. Addison Wesley,1989.

MEIROVVITCH, L. **Methods of analytical dynamics**. Mc Graw-Hill, 1970.

MITCHELL, M. **An introduction to genetic algorithms**. MIT Press, Cambridge, US, 1999.

SPIEGEL M. **Análise vetorial** - col. Shaum. Rio de Janeiro: Livro Técnico, 1969;

NASA Town Hall, Small Satellite Conference: Franklin Small Satellite Subsystem Technology and Edson Small Satellite Demonstration Missions Program, Nasa, down town, July 13, 2010.

NEEK, S. P.; MGNER, T. J.; PAULES, G. E. **NASA's small satellites mission for earth observation**. Washington, DC: NASA Headquarters Office of Earth Science, 1998.

OLIVER, J.R. Discovering individual decision rules: an application of genetic algorithms. In: INTERNATIONAL CONFERENCE ON GENETIC ALGORITHMS, 5., 1993, San Mateo, **Proceedings...** San Mateo: Morgan Kauffmann, 1993. p. 216-222.

OGATA, K. **Engenharia de controle moderno**. Prentice Hall, 2012.

OGATA, K. **Engenharia de controle moderno**. Prentice Hall, 1992.

PETERMANN, B. **Attitude control of small satellites using Fuzzy logic**. Thesis, McGill University, Montreal Canada, 1997.

PEDRYCZ, W.; PETERS, J.F.; RAMANNA, S. Hierarchical *Fuzzy* Neural Attitude Control for Satellites. In: IEEE AEROSPACE CONFERENCE, 2006, Snowmass at Aspen, CO, USA. **Proceedings...** Snowmass: IEEE, 2006. P.385-400.

PEDRYCS, W. **Fuzzy control and Fuzzy systems**. New York: John Wiley and Sons Inc. 1989.

Quanser® " **EE-479 digital control system**: Project 1 Flexible Link, 2009;

RAMOS, F.O.; SOUZA, L.C.G. CI- Based Design of an Attitude Contrller" , **Journal of Aerospace, Sciences and Applications**, v. 2. n. 1, 2011.

Ricci, M. C. **Curso de lógica Fuzzy**. São José dos Campos: INPE, 2011.Apostila.

ROSS, T. J. **Fuzzy logic**: with applictions. New York: John Wiley and Sons, 2010.

SANIAL, S.; BARAI, R.K. Design of attitude control system of a satellite **International Journal of Advanced Engineering Technology**, v. 3, n. 2, p. 13-16,, April-June, 2012.

SANTOS, I. F. **Dinâmica de sistemas mecânicos**. São Paulo: Makron Books, 2001.

SATYADAS, A.; KRISHNAKUMAR, K. GA-optimzed *Fuzzy* controller for spacecraft attitude control. IEEE INTERNATIONAL CONFERENCE ON INDUSTRIAL AND CONTROL, 1995;

SHAW, I.S.; SIMÕES, M. G. **Controle e modelagem Fuzzy**. Edgar Blucher, 1999;

SKOGESTAD, S.; POSTLETHWAITE, I. **Multivariable feedback control-analysis and design**. New York: John Wiley and Sons, 2007.

SOUZA, L. C. G. Robust controller design for a flexible space system with mixed uncertainty model. **Revista Brasileira de Ciências Mecânicas**, v. 16, n.

Special Issue, p. 116-123, 1994. International Symposium on Spacecraft Ground Control and Flight Dynamics, 7-11 Feb., 1994, São José dos Campos. (INPE-9198-PRE/4871).

SOUZA, L. C. G.; ARENA, V.R. Design of satellite attitude control algorithm based on SDRE method using gas jets and reaction wheels. **Journal of Engineering**, v. 13, Article ID 318072, 2013. doi: <[10.1155/2013/318072](https://doi.org/10.1155/2013/318072)>.

SOUZA, A. G.; SOUZA, L. C. G. Implementação de estimador de estados num simulador de sistema de controle de atitude de satélites. In: SEMINÁRIO DE INICIAÇÃO CIENTÍFICA DO INPE (SICINPE)., 2010, São José dos Campos. **Anais...** São José dos Campos: INPE, 2010. p. 29. CD-ROM; Papel; On-line. Disponível em: <<http://urlib.net/8JMKD3MGP7W/388RGCL>>. Acesso em: 20 maio 2015.

SNIDER, R.E. **Attitude control of a satellite simulator using reaction wheels and PID controller**. PhD Tesis, Air Force Institute of Technology, Ohio, USA, 2010.

THISIOTRAS, P. Stabilization and optimal results for attitude control problem. **Journal of Guidance, Control and Dynamics**, v. 19, n. 4, July-August, 1996.

TANAKA, K. **An introduction to Fuzzy logic for practical applications**. New York: Springer, 1997.

CASTRO, J. C. V. **Estudo experimental da dinâmica do sistema de controle de um satélite rígido-flexível**. 2009. 121 p. (INPE-15765-TDI/1508). Dissertação (Mestrado em Mecânica Espacial e Controle) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2009. Disponível em: <<http://urlib.net/8JMKD3MGP8W/353S76S>>. Acesso em: 20 maio 2015.

SHAW, I.S.; SIMÕES, M. G. **Controle e modelagem Fuzzy**. Edgar Blucher, 1999.

ZADEH, L.A. A critical view of our research in automatic control. **IRI Trans. Automatic Control**, AC-7, 74, 1968.

ZADEH, L.A. *Fuzzy algorithms*. **Information on J. Control**, v.12, p. 94-102, 1962.

ZADEH, L.A., *Fuzzy sets*, Information and contro, vol. 8, N° 3, pp 338-353, June, 1965

ZHOU, K.; GLOVER, K.; DOYLE, J. C. **Robust and optimal control**. Prentice-Hall, 1995.

WERTZ, J. R. **Spacecraft attitude determination and control**. D.Reidel Publishing Company, 1978.

WANG, H.; IKEDA, T.; GRIFFITIN, M. An approach to *Fuzzy* control of nonlinear system: stability and design issues. **IEEE Transactions on Fuzzy Systems**, v. 4, n. 1, p.14-23, 1996.

WELL, D.A. **Theory and problems of lagrangian dynamics**. McGraw-Hill, 1967.

APÊNDICE A - PROGRAMAS DE CONTROLE

A1

Programa responsável por passar os valores numéricos do simulador e inserir o controle *Fuzzy* no Simulink tratado na seção 4.2

Parâmetros de simulação e Lógica *Fuzzy*

```
clc
```

```
close all
```

```
clear all
```

```
% Wheel motor gains
```

```
K_motor_in =.1;
```

```
K_motor=5;
```

```
% time
```

```
dt=.05; % time step
```

```
% Wheel MOI
```

```
lw=66.17/32.17; %66.17/32.17
```

```
% MOIs
```

```
l11=3800.66/32.2; %3800.66/32.2
```

```
l22=38318/32.2; %38318/32.2
```

```
l33=36652/32.2; %36652/32.2
```

```

% LQR parameters

n=6;

m=3;

p=6;

% Kalman parameters

C=eye(6,6);

pnoise=.1;

thetanoise=.2*pi/180;

ratenoise=.1*pi/30;

tn=thetanoise;

rn=ratenoise;

z=[tn tn tn rn rn rn]';

Sz=z*z';

Sw=eye(6,6);

%

e=0;

ep=0;

for j = 1;

    % controle fuzzy

    b = newfis('Detreminação de Q');

    b.input(1).name='erro';

    b.input(1).range=[0 1];

```

```
b.input(1).mf(1).name='pequeno';
b.input(1).mf(1).type='trapmf';
b.input(1).mf(1).params=[0 0 0.1 0.7];
b.input(1).mf(2).name='medio';
b.input(1).mf(2).type='trimf';
b.input(1).mf(2).params=[0.3 0.5 0.7];
b.input(1).mf(3).name='grande';
b.input(1).mf(3).type='trapmf';
b.input(1).mf(3).params=[0.3 0.9 1 1];
b.input(2).name='deltaErro';
b.input(2).range=[0 0.1];
b.input(2).mf(1).name='pequeno';
b.input(2).mf(1).type='trapmf';
b.input(2).mf(1).params=[0 0 0.01 0.07];
b.input(2).mf(2).name='medio';
b.input(2).mf(2).type='trimf';
b.input(2).mf(2).params=[0.03 0.05 0.07];
b.input(2).mf(3).name='grande';
b.input(2).mf(3).type='trapmf';
b.input(2).mf(3).params=[0.03 0.09 0.1 0.1];
b.output(1).name='Q';
b.output(1).range=[0 10];
b.output(1).mf(1).name='pequeno';
```

```

b.output(1).mf(1).type='trimf';
b.output(1).mf(1).params=[0 2 5];
b.output(1).mf(2).name='medio';
b.output(1).mf(2).type='trimf';
b.output(1).mf(2).params=[3 5 7];
b.output(1).mf(3).name='grande';
b.output(1).mf(3).type='trimf';
b.output(1).mf(3).params=[5 8 10];
b.rule(1).antecedent=[3 3];
b.rule(1).consequent =3
b.rule(1).weight=1;
b.rule(1).connection=1;
b.rule(2).antecedent=[2 1];
b.rule(2).consequent=2;
b.rule(2).weight=1;
b.rule(2).connection=1;
b.rule(3).antecedent=[1 0];
b.rule(3).consequent=1;
b.rule(3).weight=1;
b.rule(3).connection=1;
evalfis([e ep],b);
q = evalfis([e ep],b);
fprintf('\n O valor de q é: %d \n',q );

```

```

% Ponderações de Q e R
Pq=100;
PQ=[eye(3) zeros(3);zeros(3) Pq*eye(3)];
Pr=0.0001;
R=Pr*eye(3);
i = i + 1;

```

```
end
```

A2

Programa do controlador H_{infinito} usando a função *Mixsym* do Matlab, tratado na seção 3.4.12

```
clear all;close all;
```

```
% Projeto de um controlador usando o MIXSYN
```

```
% Matrizes do sistema
```

```
A=[0 0 1.0 0; 0 0 0 1.0; 0 591.9697 -31.9744 0;0 -947.2755 31.9744 0];
```

```
B=[0 0;0 0;56.2361 0;-56.2361 0]; C=[1 0 0 0;0 1 0 0]; D=[0 0;0 0];
```

```
G=ss(A,B,C,D);
```

```
%clf;
```

```
%impulse(G,1),grid, title('Resposta de Malha Aberta ao Impulso');
```

```
% Set up the performance and robustness bounds W1 & W3
```

```
%s=zpk('s'); % Laplace variable s
```

```
s=tf([1 0],1);
```

```

MS=2; AS=0.001; WS=4;
W1=(s/MS+WS)/(s+AS*WS);
W2=[];
MT=2; AT=.001; WT=8;

W3=(s+WT/MT)/(AT*s+WT);
% Compute the H-infinity mixed-sensitivity optimal controller K1
[K1,CL1,GAM1]=Mixsyn(G,W1,[],W3);

% Next compute and plot the closed-loop system.
% Compute the loop L1, sensitivity S1, and comp sensitivity T1:
L1=G*K1;
I=eye(size(L1));
S1=feedback(I,L1); % S=inv(I+L1);
T1=I-S1;
% Plot the results:
% frequency response plots
figure(1);
sigma(I+L1,'--',T1,':',L1,'r--',...
W1/GAM1,'k--',GAM1/W3,'k-',{.1,25});grid
legend('1/\sigma(S) performance',...
'\sigma(T) robustness',...
'\sigma(L) loopshape',...

```

```

'sigma(W1) performance bound',...
'sigma(1/W3) robustness bound');
% Filtros de saída para o simulink
num1=[0 250];den1=[1 250];G1=tf(num1,den1);
num2=[50 0];den2=[1 50];G2=tf(num2,den2);
num3=[0 25];den3=[1 225];G3=tf(num3,den3);
num4=[25 0];den4=[1 25];G4=tf(num4,den4);
[A1,B1,C1,D1]=tf2ss(num1,den1);
[A2,B2,C2,D2]=tf2ss(num2,den2);
[A3,B3,C3,D3]=tf2ss(num3,den3);
[A4,B4,C4,D4]=tf2ss(num4,den4);

```

% Tentativa

```

x1 = 0;
x2 = 0.05;
x3 = 0.1;
x4 = 0.2;
inic = [x1; x2; x3; x4];

tinic = 0;
tp = 0.01;
tfim = 10;
t = tinic: tp: tfim;

```

```

%[y, t, x] = impulse(G*K1,t);
sys2=ss(G*K1);

[y, tt, x] = initial(sys2, [inic; zeros(8,1)],t);

figure(2)
plot(t,x(:,1));grid;
legend('theta_LQG');
ylabel('Theta rad'), xlabel('Tempo s');
figure(3)
plot(t,x(:,2)); grid;
legend('Thetap_LQG');
ylabel('thetap rad/s'), xlabel('tempo s');
figure(4)
plot(t,x(:,3)); grid;
legend('alpha_LQG');
ylabel('alpha m'), xlabel('tempo s');
figure(5)
plot(t,x(:,4)); grid;
legend('alphap_LQG');
ylabel('alphap m/s'), xlabel('tempo s');

```

A3

Programa usando algoritmos genéticos, responsável por gerar o controlador $H_{\text{INFINITO}}/\text{AG}$ original e simplificado.

%CompletoFlexívelAG

```
clear all; close all; clc;
```

```
% Programa Para Definir H_inf Com AG (Pablo 2005);
```

```
% As definições das variáveis estão na pag. 287 da Tese;
```

```
% Este programa desenvolvido por Pablo R. Y. A. Cardenas e
```

```
% adaptado por S. P. Santos maio de 20015
```

```
%
```

```
global npop lchrom nop pmutate pcross ncross nbits pslope pc  
bestp %initpop
```

```
global nmin nmax pone Plmax popmax mm ;
```

```
% mm=1 para minimizacao; mm=-1 para maximizacao
```

```
mm = 1;
```

```
%satelite;
```

```
%modelo do satélite
```

```
A=[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0;
```

```
    0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0;
```

```
    0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0;
```

```
    0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0;
```

```
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1;
```

```
0, 0, 0, 0, 2.5371, 0, 0, 0, 0, 0, 0.0004;  
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;  
0, 0, 0, -2.5371, 0, 0, 0, 0, 0, -0.0004,0;  
0, 0, 0, -123.6580, 0, 0, 0, 0, 0, -0.02, 0;  
0, 0, 0, 0, -119.8505, 0, 0, 0, 0, 0, -0.02];
```

```
B=[0, 0, 0;  
0, 0, 0;  
0, 0, 0;  
0, 0, 0;  
0, 0, 0;  
0.0002566, 0, 0;  
0, 0.0002776, 0;  
0, 0, 0.0002566;  
0, 0, 0.0005034;  
-0.0005034, 0, 0];
```

```
% C=[1,0,0,0,0,0,0,0,0,0,0;  
% 0,1,0,0,0,0,0,0,0,0,0;  
% 0,0,1,0,0,0,0,0,0,0,0;  
% 0,0,0,1,0,0,0,0,0,0,0;  
% 0,0,0,0,1,0,0,0,0,0,0;  
% 0,0,0,0,0,1,0,0,0,0,0];
```

```

%
C=eye(10);

D=[0,0,0;0,0,0;0,0,0;0,0,0;0,0,0;0,0,0;0,0,0;0,0,0;0,0,0;0,0,0];

% x=[phi,theta,psi,phipt,thetapt,psipt,qx,qz,qxpt,qzpt];

%Satelite=ss(A,B,C,D);
% Função do modelo satélite
%Satelite;
gamma;
% função do parametro gamma do H infinito;
%função de transferência formada pelo espaço de estado;
sysG=pck(A,B,C,D);
% contador de tempo
tic;
% inicialização dos parâmetros do AG
mini = 1.0;
nop=6;
npop=20;

```

```

ngen=10;
if isempty(rem(npop,2))
    npop =npop+1;
end;
PI=rand(npop,1);
% inicializa índice de desempenho (PI)
pmax=zeros(nop,1)+10.0;
pmin=zeros(nop,1)-10.0;
pres=zeros(nop,1)+0.02;
range=(pmax-pmin)./pres;
nbits=fix(log(range+1.0)./log(2.0));
pslope=pres;
pc=pmin;
lchrom=sum(nbits);
% Parâmetros dos Operadores Genéticos;
pmutate=0.01;    % original =0.1
pcross=0.9;     % original = 0.90
ncross=10;
% Inicializa a primeira geração;
gno=1;
genn(gno)=gno;
% Inicializa a população;
oldpop=initpop;

```

```

% função;

% Estatística do Índice de Desempenho;
[maxi(gno),mini(gno),avg(gno),sd(gno),nmax,nmin]=statsp(PI);

% %I INICIO DO LOOP PRINCIPAL

Plmax=0;
popmax=fix(2*rand(1,lchrom));
for k=1:ngen
    newpop=gen(oldpop,PI,popmax);
    gno=gno+1;
    if gno > 1 ||Plmax >0;
        pmutate=0.01; % 0.01
    end
    for i=1:npop
        k_k=[] ; % Foi usado o k singelo no prgm anterior
        while isempty(k_k);
            % Decodificação de parâmetros;
            p=abs(myDecode(oldpop(i,:)));
            %Avaliação do Índice de Desempenho PI;
            % Parâmetro da Função de Ponderação;
            [nsa,dsa,nsv,dsv,nco,dco]=Ponderacao(p);
            nsa=size(roots(nsa));

```

```

rdsa=size(roots(dsa));
rnsv=size(roots(nsv));
rdsv=size(roots(dsv));
rnco=size(roots(nco));
rdco=size(roots(dco));
if rnsa>rdsa|rnsv>rdsv|rnco>rdco
    k_k=[];
else

```

```

    %Função de Ponderação de Atitude e Velocidade;

```

```

    Wsa=nd2sys(nsa,dsa);

```

```

    Wsv=nd2sys(nsv,dsv);

```

```

Ws=daug(Wsa,Wsa,Wsa,Wsa,Wsa,Wsv,Wsv,Wsv,Wsv,Wsv);

```

```

    % Função de Ponderação Wc de controle;

```

```

    Wco=nd2sys(nco,dco);

```

```

    Wc=daug(Wco,Wco,Wco);

```

```

    % Conformação da Planta Estendida;

```

```

    systemnames='sysG Ws Wc';

```

```

    inputvar='[d(10);di(3);u(3)]';

```

```

    outputvar='[Ws;Wc;-sysG-d]';

```

```

input_to_sysG='[u+di]';
input_to_Ws='[sysG+d]';
input_to_Wc='[u]';
cleanup_system='yes';
sysoutname='tss';
sysic;
tss=syscl(tss);
[k_k,g,gfin]=hinfsyn(tss,10,3,gmin,gmax,0.001,1,1e-6,1e-
6,0);
end

%Gera nova população;
if isempty(k_k)
    disp('ERROR: Funções de ponderação incorreta ou
Gamma grande');
    oldpop(i,:)=gemb(oldpop,PI);
end
end

% Conformação da Planta Estendida para Análise;
[sen,ksen,clp]=sensibil(sysG,k_k);
% Função Objetivo;
Plaux(i) = objetivo(sen,ksen,clp,gfin);
PI(i)=Plaux(i)*mm;

```

```

    if Plaux(i)>Plmax
        pone=p;
        Plmax=Plaux(i);
        popmax=oldpop(i,:);
    end
end
genn(gno)=gno;
oldpop=newpop;
[maxi(gno),mini(gno),avg(gno),sd(gno),nmax,nmin]=statsp(PI);
best=abs(decode(oldpop(nmin,:)));
% Figura;
figure(1)
subplot(2,1,1)
plot(genn,(maxi*mm),'r',genn,(mini*mm),'g',genn,(avg*mm),'b');
ylabel('Índice de Desempenho');
xlabel('Número de Gerações')
title('azul=média, verde =melhor, vermelho=pior desempenho');
grid;
subplot(2,1,2)
plot(genn,(mini*mm),'g');
xlabel('Número de Gerações');
ylabel('Índice de Desempenho')
grid;

```

```

    pause(1);
end
disp('-fim do loop principal-')
% Fim do Loop Principal
mini=mini;
maxi=maxi;
avg=avg;
[nsa,dsa,nsv,dsv,nco,dco]=ponderacao(pone);
Ws=daug(Wsa,Wsa,Wsa,Wsa,Wsa,Wsa,Wsa,Wsv,Wsv,Wsv);
Wc=daug(Wco,Wco,Wco);
% Conformaçãõ da Planta Estendida
systemnames='sysG Ws Wc';
inputvar= '[d(10);di(3);u(3)]';
outputvar='[Ws;Wc;-sysG-d]';
input_to_sysG ='[u+di]';
input_to_Ws='[sysG+d]';
input_to_Wc='[u]';
cleanup='yes';
sysoutname = 'tss';
sysic;
tss=sysicl(tss);
[k_k,g,gfin]=hinfoyn(tss,10,3,gmin,gmax,0.001,1,1e-6,1e-6);
if isempty(k_k)

```

```
disp('ERROR:Funções de ponderação incorretas ou Gamma grande');
```

```
else
```

```
[sen,ksen,clp]=sensibil(sysG,k_k);
```

```
[AT,BT,CT,DT]=unpck(k_k);
```

```
end
```

```
% Achando as Matrizes do Compensador Ac,Bc,Cc,Dc.;
```

```
[Ac,Bc,Cc,Dc]=unpck(k_k);
```

```
% Guardando os Dados
```

```
if Plmax>0.7
```

```
save ponewst pone
```

```
end
```

```
save old_con k_k
```

```
save old_mini mini
```

```
disp('Parametros:')
```

```
Plmax
```

```
ws1=1/p(1)
```

```
Ms1=1/p(2)
```

```
deltas=p(3)
```

```
Wc1=1/p(4)
```

```
Mc1=1/p(5)
```

```
deltac=p(6)
```

```
disp('Tempo');
```

```
toc
```

```
% Resposta ao degrau;
```

```
mats1=sel(clp,[1],[1]);
```

```
mats2=sel(clp,[2],[2]);
```

```
mats3=sel(clp,[3],[3]);
```

```
mats4=sel(clp,[4],[4]);
```

```
mats5=sel(clp,[5],[5]);
```

```
mats6=sel(clp,[6],[6]);
```

```
mats7=sel(clp,[7],[7]);
```

```
mats8=sel(clp,[8],[8]);
```

```
mats9=sel(clp,[9],[9]);
```

```
mats10=sel(clp,[10],[10]);
```

```
t=0:600;
```

```
[a1,b1,c1,d1]=unpck(mats1); [y1]=step(a1,b1,c1,d1,1,t);
```

```
[a2,b2,c2,d2]=unpck(mats2); [y2]=step(a2,b2,c2,d2,1,t);
```

```
[a3,b3,c3,d3]=unpck(mats3); [y3]=step(a3,b3,c3,d3,1,t);
```

```
[a4,b4,c4,d4]=unpck(mats4); [y4]=step(a4,b4,c4,d4,1,t);
```

```
[a5,b5,c5,d5]=unpck(mats5); [y5]=step(a5,b5,c5,d5,1,t);
```

```
[a6,b6,c6,d6]=unpck(mats6); [y6]=step(a6,b6,c6,d6,1,t);
```

```
[a7,b7,c7,d7]=unpck(mats7); [y7]=step(a7,b7,c7,d7,1,t);
```

```
[a8,b8,c8,d8]=unpck(mats8); [y8]=step(a8,b8,c8,d8,1,t);
```

```
[a9,b9,c9,d9]=unpck(mats9); [y9]=step(a9,b9,c9,d9,1,t);
```

```
[a10,b10,c10,d10]=unpck(mats10);
```

```
[y10]=step(a10,b10,c10,d10,1,t);
```

```
TT=t';
```

```
% Respostas
```

```
figure(2);
```

```
sigma(Ac,Bc,Cc,Dc,{ 0.001, 0.1}),grid;
```

```
figure(3);
```

```
plot(t,y1),grid;
```

```
title('phi');
```

```
xlabel(' Tempo s')
```

```
ylabel(' Amplitude')
```

```
figure (4);
```

```
plot(t,y2),grid;  
title('theta');  
xlabel(' Tempo s')  
ylabel(' Amplitude')
```

```
figure(5);  
plot(t,y3),grid;  
title('psi');  
xlabel(' Tempo s')  
ylabel(' Amplitude')
```

```
figure(6);  
plot(t,y4),grid;  
title('phipt');  
xlabel('Tempo s');  
ylabel('rad/s');
```

```
figure(7);  
plot(t,y5),grid;  
title('thetapt');  
xlabel('Tempo s');
```

```
ylabel('rad/s');
```

```
figure(8);
```

```
plot(t,y6),grid;
```

```
title('psipt');
```

```
xlabel('Tempo s');
```

```
ylabel('rad/s');
```

```
figure(9);
```

```
plot(t,y7),grid;
```

```
title('qx');
```

```
xlabel('Tempo s');
```

```
ylabel('m');
```

```
figure(10);
```

```
plot(t,y8),grid;
```

```
title('qz');
```

```
xlabel('Tempo s');
```

```
ylabel('m');
```

```
figure(11);
```

```
plot(t,y9),grid;
```

```
title('qxpt');
```

```

xlabel('Tempo s');
ylabel('m/s')

figure(12);
plot(t,y10),grid;
title('qzpt');
xlabel('Tempo s');
ylabel('m/s');

y=[y1 y2 y3 y4 y5 y6 y7 y8 y9 y10];

```

A4

Programa Objetivo, responsável por passar as especificações desejadas para o controle, definir o índice de desempenho (IP) e servir como função Fitness para o controle AG, tratado na seção 4.4.

```

% Programa elaborado por Cárdena, 2005;
% Adaptado por Santos, 2015
% Função OBJETIVO.m avalia os parâmetros para
% o controlador

function PI=objetivo(sen,ksen,clp,gfin)

% Objetivos: Tempo de acomodação (ta), tempo de
%subida (ts) e sobre sinal (ss)

tamin=200;    % Original 200
tades=500;    % ,,      400
tamax=600;    % ,,      600

```

```

tsmin=30;      % original  40
tsdes=70;     % ,,      80
tsmax=100;    % ,,      120
ssmin=1.0;    % ,,      0.9
% 0.8 0.95;
ssmax=1.10;   % ,,      1.10
% 1.1 1.05
% Resposta degrau;
mats1=sel(clp,[1],[1]);
mats2=sel(clp,[2],[2]);
mats3=sel(clp,[3],[3]);
t=0:600;
[a1,b1,c1,d1]=unpck(mats1); [y1]=step(a1,b1,c1,d1,1,t);
[a2,b2,c2,d2]=unpck(mats2); [y2]=step(a2,b2,c2,d2,1,t);
[a3,b3,c3,d3]=unpck(mats3); [y3]=step(a3,b3,c3,d3,1,t);
TT=t';
% Tempo de acodação - > 0.02 perto;
ta1=max(TT.*(abs(y1-1)>0.02));
ta2=max(TT.*(abs(y2-1)>0.02));
ta3=max(TT.*(abs(y3-1)>0.02));
% Tempo de subida -> 0 to 0.85*r;
ts1=max(TT.*(y1<0.9)); % original 0.85
ts2=max(TT.*(y2<0.9));

```

```

ts3=max(TT.*(y3<0.9));
% Sobre sinal;
ss1=max(y1);
ss2=max(y2);
ss3=max(y3);
% Função sócio-triangular;
[ta(1)]=trimf(ta1,[tamin tades tamax]);
[ta(2)]=trimf(ta2,[tamin tades tamax]);
[ta(3)]=trimf(ta3,[tamin tades tamax]);

[tsp(1)]=trimf(ts1,[tsmin tsdes tsmx]);
[tsp(2)]=trimf(ts2,[tsmin tsdes tsmx]);
[tsp(3)]=trimf(ts3,[tsmin tsdes tsmx]);

[ssp(1)]=trimf(ss1,[ssmin 1 ssmx]);
[ssp(2)]=trimf(ss2,[ssmin 1 ssmx]);
[ssp(3)]=trimf(ss3,[ssmin 1 ssmx]);

taps=sum(tap);
tsp=sum(tsp);
ssps=sum(ssp);
% Ponderação da função aptidão;
alfa1=0.10;

```

```
alfa2=0.20;
```

```
alfa3=0.30;
```

```
beta1=0.40;
```

```
% Função aptidão;
```

```
uD=alfa1*(taps)+alfa2*(tsps)+alfa3*(ssps)+beta1*(1/gfin);
```

```
PI=uD;
```

A5

Programa responsável pela decodificação dos parâmetros e geração das funções de ponderação W_S e W_C para o H_{INFINITO}/AG simplificado.

```
% Programa elaborado por Cárdenas, 2005;
```

```
% Adaptado por Santos, 2015;
```

```
% Função Ponderação.m
```

```
% esta função decodifica parâmetros e gera a ponderação;
```

```
function[nsa,dsa,nsv,dsv,nco,dco]=ponderacao(p)
```

```
%Função de ponderação de sensibilidade Atitude(3);
```

```
% Parâmetros: Ms, ws, deltas, Mc,wc,deltac p/ adaptação proposta
```

```
nsa=[p(1)*p(2) 1];
```

```
dsa=[p(1) 0.5];
```

```
nsv=conv([p(1)*p(2) 1],[p(1)*p(2) 1]);
```

```
dsv=conv([p(1) 0.5],[p(1) 0.5]);
```

```
nco=[p(4) p(5)];
```

```
dco=[p(5)*0.5 1];
```

APÊNDICE B

Tabela B1: Parâmetros e variáveis do sistema haste flexível (SRV02)

Símbolo	Descrição
L	Comprimento da haste flexível, m
dm_i	Elemento genérico de massa da haste flexível
ω_c	Frequência natural obtida experimentalmente, $\omega_c=3\text{Hz}$
θ	Ângulo de deslocamento do cubo em radiano
$y \approx \alpha$	Deslocamento linear de um ponto genérico da haste, em m
K_{stiff}	Matriz de rigidez da haste flexível, $K_{\text{stiff}}=1,1175$
$K_{\text{gage}}=K_g$	Relação de transformação de engrenagem 1:1 N/A
J_{eq}	Momento de inércia equivalente, $J_{\text{eq}}=K_g^2(J_{\text{eq}}+J_{\text{tach}})=0,0023\text{ Kg m}^2$
$J_{\text{arm}}=J_h$	Momento de inércia da haste, $J_{\text{arm}}=0,0031452$
η_m	Eficiência do motor, $\eta_m=0.87$
η_g	Eficiência do sistema de engrenagens, $\eta_g=0.85$
K_t	Constante de conjugado do motor, $K_t=0,00767\text{ N-m/A}$
K_g	Relação de transmissão das engrenagens, $K_g=(14).(15)$
B_{eq}	Equivalente do atrito viscoso, $B_{\text{eq}}=4 \times 10^{-3}\text{ N m/(rad/s)}$
$R_m=R_a$	Resistência de armadura do motor, $R_m=2.6\ \Omega$
V_m	Fator de calibração do "strain gage" $K_{\text{gage}}=1\text{ volt/polegada}$
m	Massa da viga 0,50 kgr
K_m	Constante de fcm do motor, $K_m=0,00767\text{V-s/rad}$
J_m	Momento de inércia da armadura do motor, $J_m=3,87 \times 10^{-7}\text{ Kg m}^2$
J_{tach}	Momento de inércia do tacometro, $J_{\text{tach}}=0,7 \times 10^{-7}\text{ kg m}^2$

Fonte: Manual EE-479 da Quanser®, 2009)

B2

Dados de inércia da plataforma, torque externo e inércias das rodas de reação (SI):

Momento de inércia da plataforma

$$I_{xx} = 1,1667 \text{ kg.m}^2$$

$$I_{yy} = 1,1671 \text{ ,,}$$

$$I_{zz} = 2,2129 \text{ ,,}$$

$$I_{xy} = 0,0107 \text{ ,,}$$

$$I_{xz} = -0,0185 \text{ ,,}$$

$$I_{yz} = 0,0159 \text{ ,,}$$

Momento de inércia das rodas de reação

$$I_1 = 1,792 \times 10^{-3} \text{ Kg.m}^2$$

$$I_2 = 1,792 \times 10^{-3} \text{ ,,}$$

$$I_3 = 1,792 \times 10^{-3} \text{ ,,}$$