



MINISTERIO DA CIÊNCIA E TECNOLOGIA

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

INPE-6384-TDI/603

**REDES NEURAIS APLICADAS AO CONTROLE DE ATITUDE
DE SATÉLITES COM GEOMETRIA VARIÁVEL**

Valdemir Carrara

**Tese de Doutorado em Engenharia e Tecnologia Espaciais/Mecânica Espacial e
Controle, orientada pelo Dr. Atair Rios Neto, aprovada em junho de 1997.**

**INPE
São José dos Campos
Junho de 1997**

Publicado por:

**Coordenação de Ensino, Documentação e
Programas Especiais - CEP**

Instituto Nacional de Pesquisas Espaciais - INPE

Caixa Postal 515

12201-970 - São José dos Campos - SP - Brasil

Fone: (012) 345.6911

Fax: (012) 345.6919

E-Mail: marciana@sid.inpe.br

- Solicita-se intercâmbio
- We ask for exchange
- Si sollecita intercambio
- On demande l'échange
- Mann bittet un Austausch
- Pidese canje
- Просим обмена
- 歡迎著作交換
- 出版物交換不取酬

Publicação Externa - É permitida sua reprodução para interessados.

INPE-6384-TDI/603

**REDES NEURAIS APLICADAS AO CONTROLE DE ATITUDE
DE SATÉLITES COM GEOMETRIA VARIÁVEL**

Valdemir Carrara

**Tese de Doutorado em Engenharia e Tecnologia Espaciais/Mecânica Espacial e
Controle, orientada pelo Dr. Atair Rios Neto, aprovada em junho de 1997.**

**INPE
São José dos Campos
Junho de 1997**

**MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

INPE-6384-TDI/603

**REDES NEURAIS APLICADAS AO CONTROLE DE ATITUDE
DE SATÉLITES COM GEOMETRIA VARIÁVEL**

Valdemir Carrara

**Tese de Doutorado em Engenharia e Tecnologia Espaciais/Mecânica Espacial e
Controle, orientada pelo Dr. Atair Rios Neto, aprovada em junho de 1997.**

**INPE
São José dos Campos
Junho de 1997**

629.7.062.2:681.3.019

CARRARA, V.

Redes neurais aplicadas ao controle de atitude de satélites com geometria variável / V. Carrara.

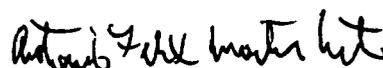
- José dos Campos: INPE, 1997.

202 p. - (INPE-6384-TDI/603)

1. Controle de atitude. 2. Redes neurais. 3. Controle de sistemas. 4. Satelites artificiais.
I. Título.

Aprovada pela Banca Examinadora em cumprimento a requisito exigido para a obtenção do Título de Doutor em Ciência Espacial

Dr. Antônio Felix Martins Neto



Presidente

Dr. Atair Rios Neto



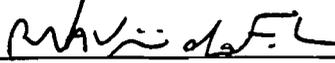
Orientador

Dr. Marcelo Lopes de Oliveira e Souza



Membro da Banca

Dr. Roberto Vieira da Fonseca Lopes



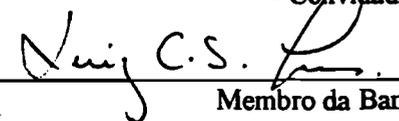
Membro da Banca

Dr. Cairo Lúcio Nascimento Júnior



Membro da Banca
- Convidado -

Dr. Luiz Carlos Sandoval Góes



Membro da Banca
- Convidado -

Candidato: Valdemir Carrara

São José dos Campos, 17 de junho de 1997

AGRADECIMENTOS

Sou extremamente grato ao Professor Atair Rios Neto, que mais do que um orientador é um obstinado incentivador e incansável doutrinador. Poucos, como ele, merecem o título de professor. Sinto-me honrado por receber sua orientação, que, durante todo o desenrolar do trabalho, sempre carregou a luz, manteve acesa a chama e, principalmente, elevou a motivação ao seu mais elevado grau.

Se, entre as diversas qualidades que possui minha esposa tivesse eu que escolher uma, certamente destacaria a sua infinita abnegação. Aqueles que, como eu, ainda se dedicam a desenvolver a ciência e a tecnologia no Brasil atual devem saber como esta qualidade é importante.

Gostaria de destacar a valiosa contribuição do Dr. Roberto V. F. Lopes, que muito auxiliou com sugestões acerca da formulação empregada na modelagem dinâmica do satélite, e também com esclarecedoras discussões sobre o ajuste do método de mínimos quadrados às características do problema.

NEURAL NETWORKS BASED CONTROL OF A SATELLITE
ATTITUDE WITH VARYING DYNAMICS

ABSTRACT

The use of neural networks for satellite attitude control is addressed in this work. In order to validate this application, a spacecraft with a variable dynamic behavior due to articulated appendages fixed to the body was chosen. The differential equations therefore show the nonlinear dynamic effects to be identified by neural nets. In this work some of the main expressions that allow system modeling through neural nets as well as two different training procedures - back-propagation and least squares - are presented. A general method for obtaining the inertia tensor and center of mass of an articulated space device, is also explained, as well as the dynamic and cinematic differential equations. These formulations were used in attitude simulation for neural network system identification and control training. The solar array deployment of the MECB's remote sensing satellite was used as an example of attitude control by means of neural nets. Three solar arrays are articulated in the satellite body and are deployed after orbit injection by a trigger device. Initially, a direct model of the satellite by means of a neural net was obtained. Afterwards, several arrangements and training procedures were tested in order to achieve the inverse model of the dynamics. The best results were obtained with the inverse training through feedback error. In order to validate the control procedure, a parameter variation method (inertia tensor and mass) together with sensor noise were employed after accomplishing the training phase, so as to verify control robustness against to parameter variation. The results show that the neural net is tolerant to sensor noise and has a relatively large capacity to compensate the parameter uncertainty.

RESUMO

Este trabalho investiga o uso de redes neurais em controle de atitude de satélites artificiais. Com a intenção de validar esta aplicação, optou-se por um satélite que não fosse constituído por um corpo rígido, mas que possuísse um comportamento dinâmico variável em função de apêndices articulados. A dinâmica assim gerada tem o carácter não-linear típico para utilização de redes neurais. São apresentados neste trabalho as principais relações que permitem a modelagem de um sistema via rede neural, bem como duas possibilidades de treinamento: retro-propagação e mínimos quadrados. Foram obtidas também as relações dinâmicas e cinemáticas do movimento de um corpo no espaço com apêndices articulados, levando-se em conta a posição do centro de massa e a variação do momento de inércia do conjunto, necessários para efetuar a simulação do movimento do satélite. Para validar o controlador de rede neural, foi utilizado como exemplo a geometria do satélite de sensoriamento remoto da MECB, durante a fase de abertura dos painéis solares, que fazem as vezes dos apêndices articulados. Inicialmente obteve-se o modelo de identificação, contendo a dinâmica direta do satélite. Posteriormente testaram-se várias formas de obtenção do modelo dinâmico inverso através da rede neural, sendo que o treinamento com realimentação do erro mostrou os melhores resultados. Para validar o controle, promoveu-se uma variação de parâmetros do satélite (momentos de inércia, massa, empuxo dos motores) e incluíram-se de ruídos nos sensores, sem entretanto refazer o treinamento da rede. Comprovou-se, assim, que a rede possui capacidade de compensação, capaz de assegurar robustez ao controle proposto.

SUMÁRIO

	<u>Pag.</u>
LISTA DE FIGURAS	xv
LISTA DE TABELAS	xix
LISTA DE SÍMBOLOS	xxi
<u>CAPÍTULO 1 - INTRODUÇÃO</u>	1
<u>CAPÍTULO 2 - REDES NEURAIIS ARTIFICIAIS</u>	7
2.1 REDES NEURAIIS.....	7
2.1.1 O Modelo de Neurônio	9
2.1.2 Conexões entre Neurônios	12
2.1.3 O Algoritmo de Treinamento.....	16
2.2 REDES NEURAIIS DE ALIMENTAÇÃO DIRETA.....	17
2.3 TREINAMENTO DE REDES NEURAIIS COM ALIMENTAÇÃO DIRETA.....	21
2.3.1 O Algoritmo de Retro-Propagação	22
2.3.2 Treinamento com o Algoritmo do Gradiente.....	26
2.3.3 Treinamento com o Algoritmo dos Mínimos Quadrados	30
2.3.3.1 A Matriz de Derivadas Parciais	34
<u>CAPÍTULO 3 - IDENTIFICAÇÃO E CONTROLE DE SISTEMAS POR REDES</u> <u>NEURAIIS</u>	39
3.1 CONTROLE DE ATITUDE DE SATÉLITES.....	39
3.2 IDENTIFICAÇÃO DE SISTEMAS DINÂMICOS	43
3.3 IDENTIFICAÇÃO DO MODELO INVERSO	46
3.4 CONTROLE ATRAVÉS DO MODELO INVERSO.....	51

	<u>Pag.</u>
<u>CAPÍTULO 4 - MODELO DINÂMICO E CINEMÁTICO DO SATÉLITE</u>	57
4.1 EQUAÇÕES DINÂMICAS DE UM CORPO RÍGIDO.....	57
4.2 EQUAÇÕES PARA UM CORPO ARTICULADO.....	61
4.3 EQUAÇÕES CINEMÁTICAS DO MOVIMENTO	67
4.4 DINÂMICA DE ABERTURA DOS PAINÉIS	69
<u>CAPÍTULO 5 - SIMULAÇÃO E RESULTADOS</u>	75
5.1 O SATÉLITE DE SENSOREAMENTO REMOTO (SSR)	75
5.2 O MODELO DINÂMICO DO SATÉLITE SSR.....	81
5.3 A IDENTIFICAÇÃO DO MODELO DIRETO.....	84
5.4 O TREINAMENTO DA REDE NEURAL DE CONTROLE.....	95
5.5 CONTROLADOR COM INCERTEZA NOS PARÂMETROS	117
5.6 CONTROLADOR COM ABERTURA NORMAL DOS PAINÉIS	120
5.7 CONTROLADOR SEM A SUPERVISÃO DO PID.....	127
<u>CAPÍTULO 6 - COMENTÁRIOS E CONCLUSÕES</u>	133
<u>REFERÊNCIAS BIBLIOGRÁFICAS</u>	137
<u>APÊNDICE A - REPRESENTAÇÃO DE VETORES EM SISTEMAS NÃO</u>	
<u>INERCIAIS</u>	143
A.1 TRANSFORMAÇÕES DE SISTEMAS DE COORDENADAS	143
A.2 SISTEMAS GIRANTES	144
A.3 SISTEMAS COMPOSTOS POR DOIS SISTEMAS GIRANTES	149

	<u>Pag.</u>
<u>APÊNDICE B - MATRIZ DE INÉRCIA DE CORPOS RÍGIDOS</u>	153
B.1 MATRIZ DE INÉRCIA	153
<u>APÊNDICE C - MODELO DO SATÉLITE SSR</u>	155
C.1 MATRIZES DE INÉRCIA DO SATÉLITE SSR	155
C.2 POSIÇÃO DAS JUNTAS	158
C.2.1 Corpo Principal	159
C.2.2 Paine Solar 1 (Frontal)	161
C.2.3 Paine Solar 2 (Traseiro)	162
C.2.4 Paine Solar 3 (Superior)	163
<u>APÊNDICE D - ANÁLISE PARAMÉTRICA DO TREINAMENTO DE UMA</u> <u>REDE NEURAL</u>	165
D.1 O SISTEMA DINÂMICO UTILIZADO COMO EXEMPLO	165
D.2 A ANÁLISE PARAMÉTRICA DO TREINAMENTO	169
D.3 A ANÁLISE DOS ERROS DE TREINAMENTO	175

LISTA DE FIGURAS

2.1 - Entradas e saídas de um neurônio artificial.	8
2.2 - Função de ativação tangente hiperbólica e derivada	11
2.3 - Rede estática ou com alimentação direta	13
2.4 - Rede com realimentação da saída	13
2.5 - Rede com realimentação entre camadas, conexões laterais e recorrência	14
2.6 - Rede com neurônios completamente conectados	14
2.7 - Representação de uma rede neural de alimentação direta com duas camadas	18
2.8 - Entradas e saídas da camada k de uma rede neural	18
3.1 - Controle de um sistema dinâmico	40
3.2 - Rede estática com realimentação da saída	44
3.3 - Identificação de modelo direto	46
3.4 - Identificação do modelo inverso pelo treinamento inverso generalizado	47
3.5 - Identificação do modelo inverso pelo método indireto	49
3.6 - Modelo especializado inverso	50
3.7 - Controlador IMC	52
3.8 - Treinamento do modelo inverso pela realimentação do erro	54
4.1 - Satélite com articulações	62
4.2 - Representação das juntas e elos do satélite	64
4.3 - Sistemas de coordenadas no satélite com geometria variável	64
4.4 - Ângulo de abertura dos painéis em função do tempo	73
4.5 - Velocidade angular dos painéis em função do tempo	73
4.6 - Aceleração angular dos painéis em função do tempo	74
5.1 - Satélite de sensoriamento remoto da MECB	76
5.2 - Modelo do impulso gerado pelo motor de hidrazina	79
5.3 - Dinâmica de abertura dos painéis do satélite SSR	83
5.4 - Velocidades angulares durante a abertura dos painéis do SSR	83
5.5 - Atitude do satélite quando o painel se desprende do corpo.	84
5.6 - Treinamento da rede de identificação da atitude	90

5.7 - Teste da RN com o modelo direto - direção do Sol no eixo x^o	90
5.8 - Erro apresentado pela rede neural na leitura do sensor solar σ_x	90
5.9 - Teste da RN com o modelo direto - direção do Sol no eixo y^o	91
5.10 - Erro apresentado pela rede neural na leitura do sensor solar σ_y	91
5.11 - Teste da RN com o modelo direto - direção do Sol no eixo z^o	91
5.12 - Erro apresentado pela rede neural na leitura do sensor solar σ_z	92
5.13 - Teste da RN com o modelo direto - magnetômetro no eixo x^o	92
5.14 - Erro apresentado pela rede neural na leitura do magnetômetro B_x	92
5.15 - Teste da RN com o modelo direto - magnetômetro no eixo y^o	93
5.16 - Erro apresentado pela rede neural na leitura do magnetômetro B_y	93
5.17 - Teste da RN com o modelo direto - magnetômetro no eixo z^o	93
5.18 - Erro apresentado pela rede neural na leitura do magnetômetro B_z	94
5.19 - Perfil de torque aplicado no eixo x^o no treinamento do modelo direto	94
5.20 - Perfil de torque aplicado no eixo y^o no treinamento do modelo direto	94
5.21 - Perfil de torque aplicado no eixo z^o no treinamento do modelo direto	95
5.22 - Treinamento do modelo direto com 12 neurônios na camada oculta	100
5.23 - Erro apresentado pela rede numa simulação após o treinamento	100
5.24 - Erro da rede com sinal nulo nas entradas referentes aos propulsores	101
5.25 - Treinamento com realimentação do erro e controle na aceleração da rede	104
5.26 - Erro apresentado pela rede de controle durante o treinamento	109
5.27 - Variação dos pesos $W_{1,1}^1$ e $W_{1,1}^2$ durante o treinamento	110
5.28 - Evolução da atitude utilizando somente o controlador PID	111
5.29 - Evolução da atitude utilizando o controlador PID e a rede	112
5.30 - Torque no eixo x^o do satélite utilizando somente o controlador PID	112
5.31 - Torque no eixo y^o do satélite utilizando somente o controlador PID	113
5.32 - Torque no eixo z^o do satélite utilizando somente o controlador PID	113
5.33 - Torque no eixo x^o do satélite utilizando a rede neural e o controlador PID	114
5.34 - Torque no eixo y^o do satélite utilizando a rede neural e o controlador PID	114
5.35 - Torque no eixo z^o do satélite utilizando a rede neural e o controlador PID	115
5.36 - Erro quadrático do controlador PID	116

5.37 - Erro quadrático do rede neural e controlador PID	116
5.38 - Erro quadrático com incerteza nos parâmetros do satélite	118
5.39 - Evolução da atitude com incerteza nos parâmetros e ruído nos sensores	119
5.40 - Atitude com abertura normal dos painéis, controle PID e rede.	122
5.41 - Torques no eixo x^o , com abertura normal dos painéis.	122
5.42 - Torques no eixo y^o , com abertura normal dos painéis.	123
5.43 - Torques no eixo z^o , com abertura normal dos painéis.	123
5.44 - Atitude com abertura normal dos painéis, somente controle PID.	124
5.45 - Torque no eixo x^o , com abertura normal dos painéis, controle PID.	124
5.46 - Torque no eixo y^o , com abertura normal dos painéis, controle PID.	125
5.47 - Torque no eixo z^o , com abertura normal dos painéis, controle PID.	125
5.48 - Erro quadrático com abertura normal, e controle efetuado pela rede e PID.	126
5.49 - Erro quadrático com abertura normal e controle exercido apenas pelo PID.	126
5.50 - Controlador com realimentação do erro, sem supervisão do PID.	130
5.51 - Controle com realimentação do erro sem supervisão do PID.	130
5.52 - Controle efetuado com o sinal do PID.	131
5.53 - Controle baseado no método especializado inverso com sinais do erro.	132
A.1 - Transformação de coordenadas	144
A.2 - Sistema de coordenadas girante	145
A.3 - Sistema de coordenadas c fixado nos apêndices do satélite	149
C.1 - Configuração do satélite SSR com painéis abertos	156
C.2 - Dimensões do satélite SSR na configuração de lançamento	156
C.3 - Sistema de coordenadas do corpo principal do satélite	159
C.4 - Sistemas de coordenadas do painel frontal	161
C.5 - Sistemas de coordenadas do painel traseiro	163
C.6 - Sistemas de coordenadas do painel superior	164
D.1 - Evolução do erro obtido durante o treinamento da rede neural	168
D.2 - Valores de dois pesos da rede neural durante o treinamento	168
D.3 - Erro obtido pela rede após uma análise paramétrica	169
D.4 - Erro obtido pela rede em função do número de neurônios da camada oculta	170

D.5 - Erro apresentado pela rede em função do número de entradas atrasadas	171
D.6 - Erro apresentado pela rede em diferentes condições iniciais para os pesos	172
D.7 - Erro apresentado pela rede sob diferentes fatores de esquecimento	173
D.8 - Impacto da matriz de covariância inicial no aprendizado da rede	174
D.9 - Erros obtidos com quantidades diferentes de pontos treinados	175
D.10 - Erro ao final do treinamento nos pontos amostrados	176
D.11 - Erro de interpolação entre os pontos amostrados	177
D.12 - Erro de propagação da rede neural	178

LISTA DE TABELAS

5.1 - SINAIS DE ENTRADA DA REDE DE IDENTIFICAÇÃO	87
5.2 - SINAIS DE SAÍDA DA REDE DE IDENTIFICAÇÃO	87
5.3 - SINAIS DE ENTRADA DA REDE DO MODELO INVERSO	98
5.4 - SINAIS DE SAÍDA DA REDE DO MODELO INVERSO	98

LISTA DE SÍMBOLOS

- a_{ik} Vetor de posição da articulação do sistema k com relação ao sistema i .
- $A_{k,o}$ Matriz de rotação que leva a representação de um vetor no apêndice k ao sistema do centro de massa do satélite.
- $A_{k,i}$ Matriz de rotação que leva a representação de um vetor no sistema k ao sistema i .
- b_i^k Patamar (*bias*) de ativação do neurônio i da camada k
- B Vetor campo magnético terrestre (leitura do magnetômetro).
- $E[.]$ Operador média ou esperança.
- $f^k(\cdot)$ Função de ativação utilizada na camada k .
- $f_r(\cdot)$ Funcional que resume as operações praticadas pela rede neural.
- F^k Matriz diagonal das derivadas da função de ativação da camada k .
- h Momento angular embarcado (resultante das rodas e volantes).
- $H(t)$ Matriz das derivadas parciais do vetor de saídas da rede com relação aos pesos.
- H_i^k Matriz de derivadas parciais do vetor de saída da rede com relação aos pesos associados ao neurônio i da camada k .
- I_k Matriz de inércia do elo k .
- I_o Matriz de inércia do corpo principal do satélite
- $K(t)$ Ganho de Kalman no instante t .
- l Número de camadas da rede neural.
- L Momento angular do satélite.
- m_k Massa do elo k .
- m_o Massa do corpo principal do satélite.
- n_k Número de neurônios da camada k .
- N_{cont} Resultante dos torques de controle de atitude aplicados ao satélite.
- N_{pert} Resultante dos torques ambientais aplicados ao satélite.
- N_w Torque interno aplicado pelo sistema de controle nas rodas ou volantes.
- $P(t)$ Matriz de covariância do erro de estimação no instante t .
- $\bar{P}(t)$ Propagação da matriz de covariância do erro de estimação.

q	Vetor de quatérnions da atitude.
r_{cm}	Vetor de posição do centro de massa do satélite.
$R(t)$	Matriz de covariância do erro $\varepsilon(t)$.
$u(t)$	Vetor de controle do sistema dinâmico no instante t .
w_i^k	Vetor dos pesos associados ao neurônio i da camada k .
w_{ij}^k	Peso entre a saída do neurônio j da camada $k-1$ com neurônio i da camada k .
$\hat{w}(t)$	Estimativa do vetor de pesos de um dado neurônio.
W^k	Matriz dos pesos entre as saídas da camada $k-1$ e a camada k .
x^k	Vetor das variáveis de saída da camada k .
x_i^k	Saída apresentada pelo neurônio i da camada k .
\tilde{x}^k	Vetor de argumentos da função de ativação da camada k .
$y(t)$	Saída apresentada pela rede no instante t .
$y^d(t)$	Saída apresentada pelo sistema (desejada para a rede) no instante t .
$y^m(t)$	Saída apresentada pela rede neural do modelo de identificação.
$y^p(t)$	Saída apresentada pelo sistema (leitura dos sensores)
$z(t)$	Resultado da aplicação da dinâmica não linear pelo neurônio.
$z(t)$	Vetor de observações na aplicação do método de mínimos quadrados.
δ_i^k	Vetor correspondente à i -ésima coluna da matriz Δ^k .
$\delta_{m,i}^k$	Elemento m, i da matriz Δ^k .
Δ^k	Matriz de retro-propagação do erro na saída da rede neural relativo à camada k .
ΔW^k	Atualização da matriz de pesos da camada k .
$\nabla(\cdot)$	Operador gradiente.
$\varepsilon(t)$	Vetor de erros na saída da rede neural no instante t .
λ	Fator de esquecimento na propagação da matriz de covariância do erro.
λ	Parâmetro de convergência do treinamento no método do gradiente.
λ_m	Parâmetro de momento do treinamento pelo método do gradiente.
μ_k	Massa reduzida do elo k .
$\theta_k(t)$	Ângulo de abertura do painel solar k no instante t .
ρ_k	Vetor posição de um elemento de massa do elo k .

- ρ_o Vetor posição de um elemento de massa do corpo principal.
- σ Vetor posição do sol (leitura do sensor solar).
- τ Vetor de magnitudes do empuxo requerido pelo controle.
- ω Vetor de velocidades angulares do satélite.
- $\Omega(\cdot)$ Operador produto vetorial.

CAPÍTULO 1

INTRODUÇÃO

Em anos recentes, as redes neurais artificiais vêm encontrando diversos campos de aplicação dentro da atividade humana e, progressivamente, têm-se aperfeiçoado. Foram utilizadas, por exemplo, em sistemas de guiagem automática de veículos motores, análise de aplicações financeiras, leitura de cheques e endereços de cartas, disposição de elementos elétricos em placas de circuito impresso, síntese de voz, visão artificial, reconhecimento de padrões em imagens, previsão de variação cambial, controle de processos em manufatura, sistemas visuais de inspeção de qualidade, controle de manipuladores robóticos, reconhecimento de linguagem falada e tradução instantânea, entre outros (Demuth e Beale, 1992). A área aeronáutica também encontrou aplicações para as redes neurais, principalmente no campo de pilotos automáticos de aeronaves, controle e detecção de falhas de sistemas de voo.

Redes neurais artificiais vêm sendo empregadas com sucesso na identificação e controle de sistemas dinâmicos (químicos, mecânicos, eletrônicos e térmicos). A identificação do sistema, isto é, a representação numérica de sua dinâmica através de uma rede neural é o primeiro passo na obtenção de controladores também baseados em redes quando não se dispõe, a princípio, de informações que permitam estabelecer a correspondência entre uma resposta desejada e a ação do controle (Chen et al., 1990; Chen e Billings, 1989). É comum o emprego de diversas configurações de rede, funções de ativação e processos de treinamento tanto na identificação quanto no controle, de forma a adequá-los às características do sistema (Chen et al., 1990a, 1990b, 1992).

Por sua vez, controladores implementados com o uso de métodos convencionais, em geral usando sistemas lineares parametrizados como modelo de trabalho, necessitam do modelo matemático do sistema a fim de extrair informações

sobre seu comportamento e também para ajustar os diversos parâmetros da lei de controle. Esta modelagem requer o conhecimento do fenômeno estudado, principalmente quando a dinâmica for não-linear. Dependendo do grau de precisão que se deseja controlar o sistema, a modelagem pode se tornar mais complexa, e seu estudo mais demorado. Existe portanto um compromisso entre a sofisticação e a qualidade de ambos: modelo e controlador. Um outro aspecto que freqüentemente dificulta a modelagem de sistemas dinâmicos é a grande quantidade de parâmetros (não necessariamente variáveis de estado) que se alteram em função da posição e do tempo. Controladores adaptativos conseguem contornar este problema produzindo uma alteração na forma de atuação do controle sempre que ocorrer uma mudança nas condições de operação do sistema. Sistemas que utilizam controle robusto, se por um lado possuem a vantagem de uma modelagem relativamente simples da dinâmica (nem sempre possível de ser implementada), por outro tornam o próprio modelo do controlador bastante sofisticado em certos casos. Em vista destes aspectos, as redes neurais vêm ganhando espaço quer na modelagem dinâmica, quer na identificação do controle que atua no sistema. No primeiro caso, as redes apresentam como maior vantagem a praticamente inexistente necessidade de um conhecimento prévio da dinâmica. No segundo caso, embora não dispensem um modelo de controle em certas aplicações (em outras este modelo pode não existir), freqüentemente podem substituí-lo com vantagens na implementação computacional.

Alguns trabalhos utilizam redes neurais juntamente com procedimentos convencionais de controle (Kawato et al., 1988; Ahmed e Tasadduq, 1994; Nascimento Jr., 1994). Pode-se, por exemplo, utilizar uma rede treinada para representar o modelo direto da planta de modo a obter uma linearização da dinâmica do sistema no ponto de operação, a cada instante. A partir daí, um controlador convencional tipo PID pode ser implementado, com ganhos calculados com base na linearização da planta previamente obtida com a rede. As vantagens deste procedimento devem-se sobretudo à teoria já bastante desenvolvida de controladores robustos aplicados a sistemas lineares.

Na área espacial o uso de redes neurais ainda se encontra bastante limitado. Poucas são as referências, porém são bastante promissoras as possibilidades: predição, detecção, análise e recuperação de falhas em subsistemas, controle de atitude em circunstâncias normais e adversas, controle autônomo de órbita e roteamento de canais de comunicação, entre outros. Vadali et al. (1993), computam trajetórias otimizadas em malha aberta para manobrar um satélite sujeito a torques de gradiente de gravidade. A seguir, treinam uma rede neural RBF (*Radial Basis Function* – função de base radial) com os pontos gerados pelo sinal de controle ótimo, e, portanto, utilizam a rede como um aproximador de funções e não propriamente como controlador da dinâmica. Rios Neto e Rao (1996) apresentam um interessante propagador de órbitas baseado em redes neurais, sem aplicar, contudo, ao problema de controle autônomo de órbitas. Já Yen (1995) serve-se também da RBF para propor um controlador de vibração de estruturas espaciais, descartando no entanto o processo de controle da atitude.

O movimento em atitude de um satélite artificial, isto é, o movimento de rotação que este executa em torno de seu centro de massa, assemelha-se ao de um corpo rígido movendo-se na ausência de torques externos, regido pelas equações de Euler (Wertz, 1978). Nestas equações não-lineares aparecem termos cruzados nas velocidades angulares, isto é, produtos entre duas velocidades ortogonais entre si. Estes termos impedem uma solução analítica trivial da equação diferencial, exceto em casos bastante particulares. Leis de controle baseadas em linearizações das equações de movimento em torno do ponto de operação são bastante conhecidas e largamente utilizadas para controlar a atitude (Wertz, 1978). Podem falhar, entretanto, quando são efetuadas manobras que levam o satélite para regiões afastadas do ponto de operação. A busca por outras formas de controle capazes de lidar com as não-linearidades do sistema é uma das motivações para o desenvolvimento deste trabalho.

Propõe-se analisar e viabilizar um controlador de atitude de satélites através de redes neurais, onde estas assumem o conhecimento da dinâmica não-linear do movimento. O desenvolvimento do controlador envolve o treinamento de uma rede

neural para efetuar a identificação do modelo e outra para controlar a dinâmica. Visto que redes neurais possuem propriedades de generalização (Antsaklis, 1990; Hunt et al., 1992), é imposto que a rede deve ser treinada para levar o satélite de uma atitude qualquer para condições finais fixas. Para validar o controle e diferenciá-lo das implementações tradicionais, baseadas em linearizações da dinâmica nas regiões próximas à referência (ou do ponto de equilíbrio), adota-se um satélite de geometria variável (com momento de inércia variante no tempo) de forma a gerar equações dinâmicas não-lineares, onde este último tipo de controle não consegue bom desempenho. Como exemplo da implementação, optou-se por realizar uma simulação utilizando as características do satélite de sensoriamento remoto (SSR) da MECB, durante a fase de aquisição de atitude, onde grandes alterações na dinâmica da planta ocorrem rapidamente devido ao processo de abertura dos painéis solares.

O treinamento da rede de controle é realizado sem a presença física do sistema dinâmico - o próprio satélite, no caso - visto que a geração de torques erráticos durante o aprendizado poderia colocá-lo numa situação crítica (com os painéis apontando na direção contrária à do Sol, por exemplo). Tampouco é possível efetuar o treinamento antes do lançamento, sob condições controladas, visto ser impossível reproduzir na superfície da Terra o ambiente livre de torques externos que ocorre no espaço. Por isso não só o comportamento dinâmico da atitude deve ser efetuado através de simulação computacional, como também as medições efetuadas pelos sensores e as atuações dos propulsores de bordo. O desenvolvimento do trabalho envolve o equacionamento e implementação em computador da dinâmica do satélite, composto por um corpo principal rígido e apêndices (painéis solares) também rígidos articulados a este corpo. As relações dinâmicas para corpos com apêndices articulados no espaço são deduzidas para um caso geral, já que não se encontram na literatura pesquisada. As possíveis geometrias utilizadas nas redes neurais e os métodos de treinamento são codificados com base na solução adotada para o problema. Finalmente, os esquemas de controle através de redes neurais são selecionados, simulados e testados, buscando-se

adequá-los e adaptá-los ao problema específico de controle de atitude de satélites com geometria variável.

Estas tarefas estão expostas como se segue. No Capítulo 2 serão descritas as redes neurais, variações de arquitetura e métodos de treinamento. Os principais métodos de controle baseados em redes neurais são apresentados no Capítulo 3, com ênfase nos métodos de treinamento da rede utilizada como controlador do sistema dinâmico. As equações da dinâmica de corpo rígido com apêndices articulados são deduzidas no Capítulo 4, junto com um método que sistematiza a obtenção da matriz de inércia de um corpo qualquer com apêndices articulados. No Capítulo 5 descreve-se a implementação do controle no satélite exemplo, e os principais resultados obtidos com o emprego da rede neural no controle da atitude deste satélite. As conclusões extraídas deste trabalho são então resumidas no Capítulo 6.

CAPÍTULO 2

REDES NEURAIIS ARTIFICIAIS

2.1 REDES NEURAIIS

Redes neurais artificiais são estruturas idealizadas, capazes, por exemplo, de aproximar qualquer função contínua com grau de precisão previamente escolhido. Sua aplicação em diferentes áreas como reconhecimento de padrões, identificadores (separadores), memorizadores, simuladores e emuladores (normalmente associados com o caráter cognitivo do sistema nervoso humano), fez com que encontrassem cada vez mais espaço como um método eficiente e rápido de aproximação de funções, e também na identificação e controle de sistemas dinâmicos. A literatura na área é bastante vasta, compreendendo aspectos gerais (Bavarian, 1988; Narendra, 1991; Werbos, 1991; Shoureshi, 1991; Astrom, 1991), teoria (Psaltis et al., 1988; Chu et al., 1990; Sartori e Antsaklis, 1992) e aplicações em controle e modelagem (Kawato et al., 1988; Anderson, 1989; Nagata et al., 1990; Nguyen e Widrow, 1990; Narendra e Parthasarathy, 1990; Naidu et al., 1990; Miller et al., 1990; Kyung et al., 1994).

A organização do cérebro humano motivou os primeiros trabalhos significativos em redes neurais artificiais, no fim da década de 60 (Antsaklis, 1990; Rosenblatt, 1961). Estes trabalhos pioneiros ficaram esquecidos durante os anos 70, para depois ressurgir na década de 80. O principal motivo do novo interesse causado pelas redes neurais artificiais deveu-se, principalmente, ao desenvolvimento de novas expressões que permitiram o aprendizado de rede multi-camadas, com a adoção de algoritmos como a retro-propagação (*back-propagation*) e o já tradicional mínimos quadrados. Além disso ficou provado que redes neurais de apenas duas camadas (uma camada oculta e outra de saída) podem aproximar com uma acuidade pré estabelecida qualquer função não-linear contínua e limitada (Cybenko, 1989; Hornik et al., 1989).

Com as redes realimentadas de Hopfield (Hopfield, 1982) passou-se a contar com uma poderosa ferramenta para identificação e simulação de trajetórias e sistemas dinâmicos.

A principal vantagem das redes reside sobretudo na sua capacidade de aprendizagem, o que elimina a necessidade de modelos matemáticos sofisticados na resolução do problema. A possibilidade de processamento paralelo é sem dúvida um fator que, no futuro, quando for possível dispor de microprocessadores com elevado grau de paralelismo, terá um impacto decisivo na implantação de um número cada vez maior de sistemas modelados e controlados por redes neurais. Outra vantagem é a relativa tolerância das redes a falhas de alguns poucos neurônios, sem que com isso seja comprometido o desempenho da rede, principalmente para sistemas com elevado número de neurônios (Antsaklis, 1990).

Uma rede neural ou neuronal é uma estrutura composta por um conjunto de unidades básicas denominadas *neurônios*, conectados entre si. Cada neurônio artificial (para distinguí-los dos neurônios dos cérebros dos animais) é um operador de uma única saída e diversas entradas conectadas nas saídas de outros neurônios que compõem a rede. O neurônio pondera as entradas através de parâmetros denominados *pesos* para, em seguida, aplicar uma transformação no somatório das entradas ponderadas. Esta transformação, na sua forma mais geral, consiste na aplicação de uma *dinâmica linear*, seguida de uma *função de ativação*, f , conforme esquematizado na Figura 2.1.

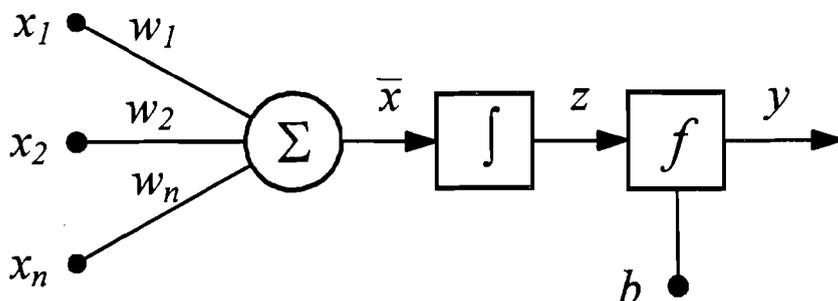


Fig. 2.1 - Entradas e saídas de um neurônio artificial.

Os pesos podem ser fixados a priori a partir de métodos de otimização (Sartori e Antsaklis, 1992), ou ainda ser determinados através de processos de *aprendizagem* ou *treinamento*. No treinamento, os pesos são ajustados por um algoritmo iterativo com o uso de uma função objetivo a ser minimizada. Durante a fase de minimização, na qual os pesos são ajustados, a cada passo a rede aprende o comportamento do sistema ao redor dos pontos treinados.

As redes neurais diferenciam-se quanto ao tipo de neurônio empregado, quanto ao arranjo das conexões entre os neurônios e quanto ao processo de treinamento. Segue uma explicação de algumas implementações distintas.

2.1.1 O Modelo de Neurônio

Na formulação computacional de uma rede neural, admite-se que a informação flui através dos neurônios seqüencialmente e instantaneamente, isto é, as saídas dos diversos neurônios estão sincronizadas entre si, e não há atraso temporal significativo entre o sinal contendo os valores de entrada e a disponibilidade na saída do neurônio. Esta idealização, na verdade, não ocorre em um neurônio natural, pois além de não haver sincronismo nas conexões entre eles, existe um atraso entre a excitação e a resposta. Na tentativa de aproximar o funcionamento do neurônio artificial de seu equivalente biológico, é introduzida uma dinâmica linear na operação efetuada pelo neurônio (Hopfield, 1982; Hunt et al., 1992). Esta dinâmica produz um atraso na resposta e, portanto, faz com que o neurônio atue como um elemento armazenador de informação. Na implementação computacional, este atraso é responsável pelo sincronismo nas saídas dos neurônios em redes com grande número de conexões entre eles, embora, na proposição de Hopfield, a cada intervalo de tempo apenas um neurônio qualquer da rede (escolhido aleatoriamente) tem sua saída avaliada, o que a torna completamente assíncrona.

A dinâmica linear, nem sempre presente na modelagem do neurônio, aplica uma integral de convolução no valor de entrada, na forma:

$$z(t) = \int_{-\infty}^t h(t-t') \bar{x}(t') dt' \quad (2.1)$$

onde \bar{x} constitui o somatório das entradas ponderadas no neurônio, dado por:

$$\bar{x}(t) = \sum_{i=1}^n w_i x_i(t) \quad (2.2)$$

sendo que x_i , $i = 1, \dots, n$, constituem as entradas do neurônio combinadas linearmente no somatório através dos pesos w_1, w_2, \dots, w_n .

Se a transformação $h(t)$ representar uma função exponencial, a convolução resulta em:

$$\alpha_0 \dot{z}(t) + \alpha_1 z(t) = \bar{x}(t), \quad (2.3)$$

válida para sistemas contínuos, com α_0 e α_1 constantes. No caso discreto, a integral resulta:

$$\alpha_0 z(t+1) = \bar{x}(t) - \alpha_1 z(t), \quad (2.4)$$

onde o atraso introduzido pela dinâmica linear fica evidente.

Ao resultado da dinâmica linear, acrescido de um limiar de ativação ou patamar b (*bias*), é aplicada uma função de ativação, $f(\cdot)$, geralmente uma função não-linear limitada. A saída do neurônio é dada então por:

$$y(t) = f(z(t) + b) . \quad (2.5)$$

O desempenho característico de uma rede neural é em grande parte influenciado pelo tipo de função de ativação escolhido. Em problemas de classificação, reconhecimento de padrões e memorização, onde as respostas geralmente assumem poucos valores (sim ou não, 0 ou 1, etc.) empregam-se funções de ativação descontínuas do tipo degrau, linear truncada ou a função sinal (Hunt et al., 1992; Demuth e Beale, 1992). Sistemas contínuos não-lineares utilizam funções contínuas e não-lineares nas camadas intermediárias, e agrupam-nas linearmente na saída. Entre estas funções, destacam-se a função sigmóide:

$$f(x) = \frac{1}{1 + e^{-ax}} , \quad (2.6)$$

e suas variações como a função tangente hiperbólica, cujo gráfico é mostrado na Figura 2.2 (juntamente com sua derivada):

$$f(x) = \frac{1 - e^{-ax}}{1 + e^{ax}} . \quad (2.7)$$

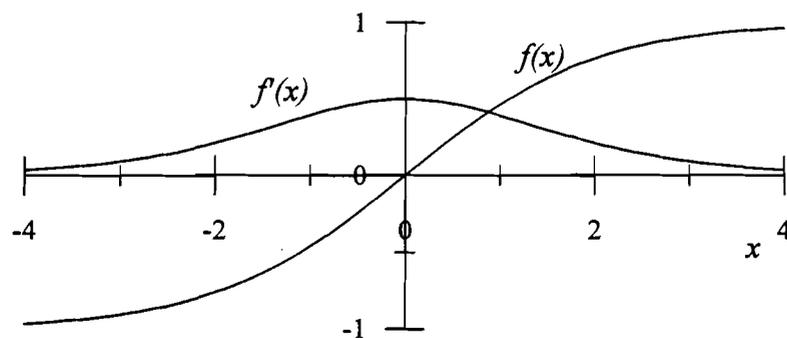


Fig. 2.2 - Função de ativação tangente hiperbólica e derivada.

O parâmetro α , positivo, estabelece o grau de transição entre os limites inferior e superior da função sigmóide. Para valores grandes de α a tangente hiperbólica se aproxima da função sinal.

Também bastante utilizada como função de ativação é a gaussiana dada por:

$$f(x) = e^{-x^2/\sigma^2} \quad (2.8)$$

A função de ativação irá definir as formas de treinamento possíveis para a rede. Certos procedimentos de determinação dos pesos requerem, por exemplo, que esta função seja limitada e diferenciável em todos seus pontos.

2.1.2 Conexões entre Neurônios

Dependendo da geometria como são realizadas as conexões entre os neurônios, tanto o seu comportamento como também sua forma de treinamento são afetados. Uma possibilidade freqüentemente empregada é a disposição dos neurônios em camadas, com as saídas de uma determinada camada conectadas nas entradas de todos os neurônios da camada seguinte, e assim por diante. As Figuras 2.3 a 2.6 mostram diversas arquiteturas possíveis de conexões de neurônios em redes neurais. Por conveniência, as redes mostradas compõem-se somente de duas camadas com dois neurônios cada uma. Na Figura 2.3 está representada uma rede *estática* ou de *alimentação direta (feedforward)*, enquanto que as demais mostram implementações diferentes de redes *dinâmicas*. Na Figura 2.4, as saídas da rede são realimentadas nas entradas, constituindo uma configuração bastante utilizada na identificação de sistemas, denominado de modelo em paralelo (Narendra e Parthasarathy, 1990). Na Figura 2.5, há uma realimentação entre camadas, isto é, as saídas de uma determinada camada são realimentadas na camada anterior. Além disso, ocorrem também conexões laterais (a saída de um neurônio de uma determinada camada alimenta os neurônios da mesma camada) e recorrência (auto-realimentação dos neurônios) neste tipo de rede (Jin et al.,

1994). Finalmente, numa rede com conexão completa, mostrada na Figura 2.6, a saída de cada neurônio alimenta todos os outros, inclusive a si próprio. Como as entradas da rede também excitam todos os neurônios, nesta rede a noção de camadas perde o sentido. Ainda assim, pode-se dizer que nesta configuração a rede apresenta uma camada de saída, onde estão os neurônios conectados diretamente aos sinais de saída, e uma camada interna, não conectada a estes sinais. A saída dos neurônios internos é algumas vezes denominada de *hiperestado* da rede (Żbicowski e Gawthrop, 1992).

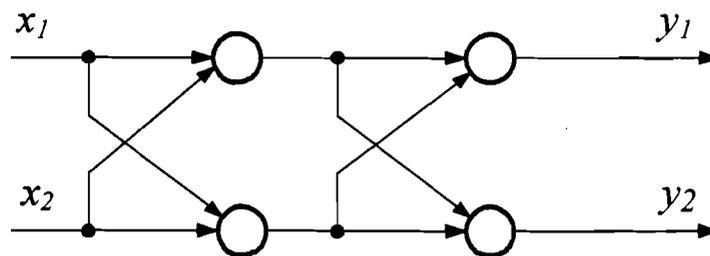


Fig. 2.3 - Rede estática ou com alimentação direta.

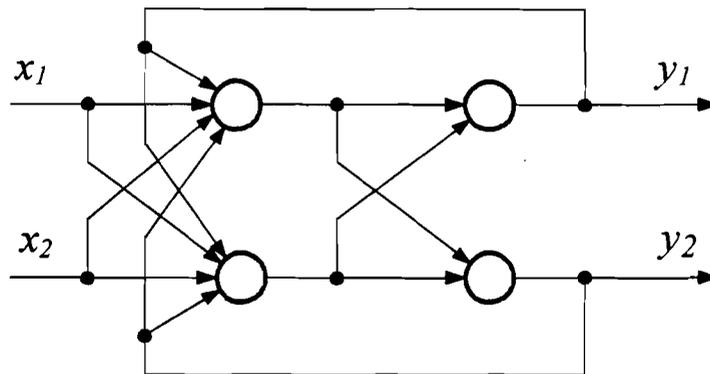


Fig. 2.4 - Rede com realimentação da saída.

A saída de uma rede neural estática pode ser colocada em termos de suas entradas na forma:

$$y = f(x), \quad (2.9)$$

que não reflete um comportamento dinâmico. Justamente em virtude deste fato, redes estáticas tem uso limitado na identificação de sistemas dinâmicos, podendo ocasionar instabilidade e um aprendizado deficiente. Entretanto, o uso de redes estáticas para controlar sistemas é um processo viável, caso seja possível atribuir uma forma não dinâmica ao controlador, como por exemplo para prover um ganho não-linear num sistema de realimentação de erro (Żbicowski e Gawghrop, 1992).

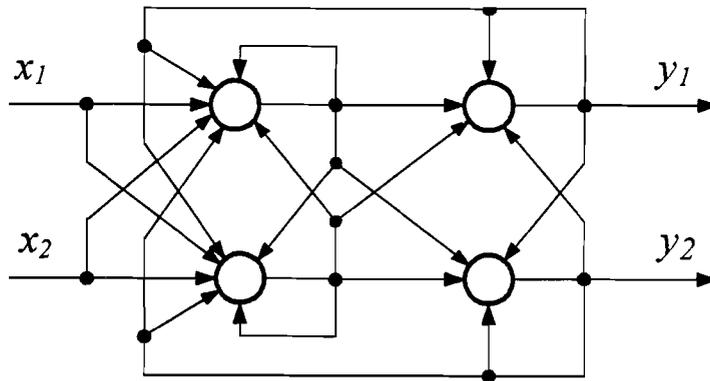


Fig. 2.5 - Rede com realimentação entre camadas, conexões laterais e recorrência.

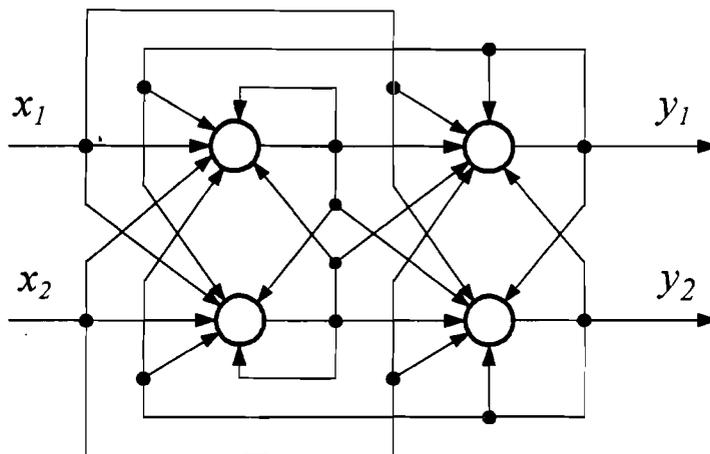


Fig. 2.6 - Rede com neurônios completamente conectados.

Numa rede realimentada, a sincronia ou o atraso na realimentação exerce papel fundamental. Note-se na Figura 2.4, por exemplo, que a rede fornece a saída no instante $t+\delta t$, e esta informação é realimentada no instante t . Para tornar este atraso evidente, adota-se um neurônio dinâmico, ou, no caso em que o neurônio não possua um

comportamento dinâmico, pode-se introduzir na malha de realimentação um elemento de atraso: um integrador ou um armazenador. Assim, a rede discretizada fornece na saída uma relação na forma:

$$y(t+1) = f[y(t), x(t)]. \quad (2.10)$$

Este processo, infelizmente, é impraticável nas redes contendo conexões laterais e recorrência, conforme as Figuras 2.5 e 2.6. Torna-se necessário, então, admitir que cada neurônio possua uma dinâmica própria, capaz de prover por si só o atraso necessário. A saída de cada neurônio, no instante $k+1$, fica determinado pelo estado que os demais neurônios a ele conectados possuíram no instante k , quer provenham da camada anterior ou posterior. O estado assumido pela camada k da rede da Figura 2.5, por exemplo, no instante $t+1$ pode possuir a forma:

$$x^k(t+1) = f^k [W^{k-1,k} x^{k-1}(t) + W^{k,k} x^k(t) + W^{k+1,k} x^{k+1}(t)] \quad (2.11)$$

onde W^{ij} é a matriz de peso que relaciona a saída dos neurônios da camada i à entrada dos neurônios da camada j , e f^k é a função de ativação da camada k . O primeiro termo do argumento da função de ativação traduz a alimentação direta (*feedforward*), o segundo compreende as conexões laterais (*crosstalk*) e recorrência (*recurrency*), e o terceiro termo é a realimentação (*feedback*) da camada posterior. É claro que a expressão acima deve ser modificada convenientemente quando se tratar da primeira e última camadas, visto que nestes casos não deve haver, respectivamente, alimentação direta e realimentação. Embora o treinamento possa ser dificultado face às interconexões existentes, o poder de representação de sistemas dinâmicos dessas redes é aumentado pelo mesmo motivo (Jin et al., 1994). Esta formulação, embora simplificada, é análoga à Relação 2.4, com a constante α_1 nula.

A estabilidade das redes neurais realimentadas é um problema bastante abordado na literatura. Embora ainda careçam de prova abrangente, alguns resultados

parciais garantem a estabilidade da rede neural sob certas hipóteses simplificadas (Hunt et al., 1992, Etxebarria, 1994). Uma das possibilidades para a prova de estabilidade consiste em se admitir que o sistema dinâmico possa ser representado por uma equação diferencial linear invariante no tempo. Prova-se, a seguir, segundo algum critério já estabelecido (Liapunov, localização dos pólos e zeros da função de transferência), a estabilidade da rede de identificação agindo sobre esta classe de equações.

2.1.3 O Algoritmo de Treinamento

É na forma de treinamento onde ocorrem as maiores diferenças nas diversas implementações de redes neurais. O treinamento pode diferir, de início, quer a rede seja estática ou dinâmica, e busca prover a rede de características que melhor se adaptem ao problema em questão. Embora seja possível, em determinadas situações, obter os pesos da rede diretamente a partir de um processo algébrico (Sartori e Antsaklis, 1992), os algoritmos iterativos são mais difundidos, em virtude de diversos aspectos. Estes algoritmos, geralmente bastante simples na sua formulação, utilizam critérios de busca por aproximações sucessivas, onde a função a ser minimizada depende da diferença entre o valor desejado na saída da rede e o valor realmente apresentado. Nas redes estáticas, algoritmos do tipo método do gradiente, com retro-propagação do erro, são bastante eficazes no treinamento. Este mesmo processo pode ser empregado nas redes dinâmicas, embora existam alternativas no caso de utilização de neurônios dinâmicos (Hopfield, 1982). Os algoritmos de treinamento são bastante variados em sua forma, mas raramente dispensam o uso do algoritmo de retro-propagação na avaliação do erro nas camadas intermediárias. Existem algumas ferramentas já implementadas em MATLAB que permitem a construção e o treinamento de redes em diversas configurações (Demuth e Beale, 1992; Norgaard, 1995, 1996).

Redes realimentadas como aquelas mostradas nas Figuras 2.4, 2.5 e 2.6 podem ser treinadas para assumir dois tipos de comportamentos distintos: treinamento de ponto fixo e treinamento para trajetória. No primeiro caso, a saída apresentada pela

rede deve ser igual à sua entrada, e o intuito é fazer com que a rede atinja o regime permanente num ponto de equilíbrio predefinido, com eventuais oscilações totalmente amortecidas. A rede é treinada com padrões não corrompidos por ruídos, que se tornam pontos de equilíbrio. A seguir, são apresentados para a rede alguns destes padrões contendo ruídos. A cada nova iteração, a saída da rede se aproxima do ponto de equilíbrio associado ao padrão apresentado e, como esta saída é realimentada na própria rede, ao fim de algumas iterações obtém-se o resultado final. Este tipo de treinamento é utilizado em problemas de reconhecimento de padrões (Hopfield, 1982; Hunt, 1992), embora, dado o seu caráter dinâmico, são também usualmente empregadas na identificação, isto é, na emulação de sistemas físicos. No treinamento para trajetória, procura-se treinar a rede de forma a seguir uma trajetória especificada, e, portanto, esta assume um comportamento dinâmico. Neste tipo de problema, torna-se necessário prover a rede com informações passadas do estado do sistema, o que faz com que a rede atue como um integrador numérico, propagando o estado de forma a prevê-lo no instante seguinte.

Embora, como foi dito, uma rede estática não possua as características de comportamento dinâmico, a realimentação do estado do sistema (ou da própria rede, como na Figura 2.6), permite que ela adquira tal particularidade, mesmo na ausência da dinâmica linear embutida no modelo dos neurônios. Esta peculiaridade das redes estáticas, aliada da relativa facilidade de treinamento fornecida pelo algoritmo de retropropagação do erro, faz com que ela se torne uma poderosa ferramenta na identificação e controle de sistemas dinâmicos. Em vista disso, ela foi adotada como candidata natural ao problema estudado aqui, e será mostrada com mais detalhes nas seções seguintes.

2.2 REDES NEURAIS DE ALIMENTAÇÃO DIRETA

Uma rede neural de alimentação direta (*feedforward*) consiste de uma transformação de mapeamento que leva n_0 elementos de sua entrada para uma saída composta de n_l elementos. Na sua forma típica, esta rede apresenta várias camadas de

neurônios artificiais onde cada camada é composta por vários neurônios não dinâmicos, conforme esquematizado na Figura 2.7. As diferentes camadas da rede ponderam suas respectivas entradas com pesos e aplicam a função ativação no somatório das entradas ponderadas, para cada uma de suas saídas.

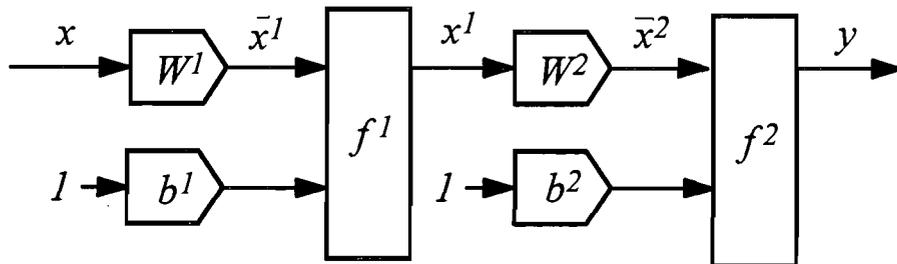


Fig. 2.7 - Representação de uma rede neural de alimentação direta com duas camadas.

Cada camada de uma rede é formada por vários neurônios, e a forma usual de representação de uma camada é dada na Figura 2.8. A camada k representada na figura, possui n_{k-1} argumentos na entrada e n_k parâmetros de saída. O expoente dos vetores e matrizes se refere ao número da camada em questão.

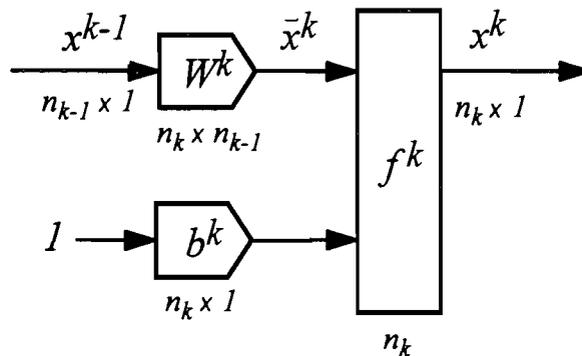


Fig. 2.8 - Entradas e saídas da camada k de uma rede neural.

A saída x_i^k apresentada pelo neurônio i da camada k é então dada pela relação:

$$x_i^k = f^k \left(\sum_{j=1}^{n_{k-1}} w_{ij}^k x_j^{k-1} + b_i^k \right). \quad (2.12)$$

O patamar da função de ativação, b^k , pode ser eliminado se for incluído mais uma componente no conjunto dos argumentos de entrada, de valor unitário. O processo de determinação do patamar fica automaticamente transferido para a estimação de mais um conjunto de pesos associados à entrada unitária. É claro que nem sempre este artifício pode ser empregado, particularmente em redes cujas funções de ativação incluem o patamar de forma não-linear. Nestes casos, uma formulação convencional pode ser realizada. O processo de cálculo efetuado pela rede neural pode ser colocado numa forma algébrica utilizando vetores e matrizes, confirmando assim a característica de paralelismo apresentada pelas redes. Nesta notação, sendo W^k a matriz de pesos da camada k , o vetor de saída desta camada fica:

$$x^k = f^k(W^k x^{k-1}). \quad (2.13)$$

onde o vetor x^k e a matriz de pesos W^k são compostos por:

$$x^k = \begin{bmatrix} f^k(\bar{x}_1^k) \\ \vdots \\ f^k(\bar{x}_{n_k}^k) \\ 1 \end{bmatrix} \quad (2.14)$$

$$W^k = \begin{bmatrix} w_{11}^k & \cdots & w_{1n_{k-1}}^k & b_1^k \\ w_{21}^k & \cdots & w_{2n_{k-1}}^k & b_2^k \\ \vdots & & \vdots & \vdots \\ w_{n_k 1}^k & \cdots & w_{n_k n_{k-1}}^k & b_{n_k 1}^k \end{bmatrix}. \quad (2.15)$$

e portanto, com a introdução do patamar dentro da matriz de pesos, as dimensões de x^k e W^k estendem-se para n_k+1 e $n_k \times n_{k-1}+1$, respectivamente.

Diversas camadas podem ser agrupadas para formar uma rede neural estática. Geralmente as funções de ativação são comuns para as camadas intermediárias, mas diferem da última. Redes estáticas com duas camadas (uma camada oculta e outra de saída) aplicam primeiro uma função de ativação não-linear (sigmóide ou tangente hiperbólica) nos parâmetros de entrada, para em seguida combiná-los linearmente na saída. Se n_k for o número de neurônios da camada k de uma rede composta por l camadas, então o vetor de saídas da rede, x^l , é função do vetor de entrada x^0 na forma:

$$x^l = f^l(W^l f^{l-1}(W^{l-1} \dots f^1(W^1 x^0) \dots)) \quad (2.16)$$

O número de camadas influi na qualidade dos resultados da rede, e pode ser mostrado que uma rede com um número maior de camadas geralmente apresenta resultados melhores quando comparados com uma rede com menos camadas e mesma quantidade de neurônios (Nguyen e Widrow, 1990; Chen e Billings, 1992). No entanto, a capacidade de generalização da rede, isto é, o grau de aproximação nos pontos onde a rede não foi treinada, é mais pronunciada nas redes com poucas ou apenas uma camada oculta (Baffes et al., 1991). Por outro lado, redes com mais camadas conseguem uma maior precisão nos pontos onde ela foi treinada. De qualquer forma, pode ser mostrado que uma rede com duas camadas sendo a segunda uma combinação linear das saídas não-lineares da primeira camada, pode representar com um dado grau de aproximação qualquer função não-linear contínua e limitada. Sistemas dinâmicos estáveis, representados por equações diferenciais ou a diferenças em termos das variáveis de estado e de ação, possuem característica de continuidade e confinamento e podem, por isso, serem aproximadas por redes neurais (Chen e Billings, 1992).

Portanto quando a dinâmica do sistema a ser modelado não for complexa, uma rede estática de duas camadas com função de transferência do tipo sigmóide na primeira camada e linear na camada de saída irá produzir resultados satisfatórios na maior parte das aplicações. O número de neurônios nas camadas intermediárias (ou

ocultas) é importante do ponto de vista da aproximação dos resultados. Poucos neurônios levam à instabilidade ou má aproximação. Muitos neurônios, por sua vez, além de sobrecarregar o tempo de aprendizado, podem acarretar grandes oscilações da saída nos pontos em que a rede não foi treinada (Billings et al., 1992). Na verdade, utilizam-se redes com mais de duas camadas somente em sistemas muito complexos, principalmente para reduzir o número de neurônios total da rede e aumentar o desempenho computacional.

O treinamento de redes estáticas consiste em métodos de obtenção de seus parâmetros. Estes parâmetros consistem nos pesos associados às conexões ligando os diversos neurônios e seus respectivos patamares (*bias*). Os algoritmos de treinamento normalmente partem da minimização do erro apresentado na saída da rede como forma de obtenção dos pesos e variam também conforme a forma de implementação: podem ser estimadores por lote, estimação estocástica (Rios Neto, 1994) ou por recursão, sendo que estes últimos são utilizados mais frequentemente, devido principalmente à facilidade de uso do algoritmo recursivo de retro-propagação do erro. Neste capítulo serão abordados dois algoritmos: os mínimos quadrados recursivos e o algoritmo do gradiente.

Cabe ainda salientar que, uma vez concluído o treinamento de uma rede estática, a diferença entre o sistema real e o resultado da rede é função exclusiva do número de neurônios empregado, do número de camadas e da forma de treinamento. O número de neurônios e o número de camadas devem ser ajustados empiricamente, já que ainda não se dispõe de regras estabelecidas matematicamente para relacionar estas variáveis com o grau de representação desejado para o modelo.

2.3 TREINAMENTO DE REDES NEURAIS COM ALIMENTAÇÃO DIRETA

O treinamento de uma rede neural consiste num procedimento numérico de estimação dos pesos, geralmente baseado na minimização de uma função que depende

do erro apresentado na saída da rede. Diferentes processos de obtenção dos pesos, e diferentes funções de minimização levam a diferentes algoritmos de treinamento, com características particulares que se prestam melhor para resolver determinados problemas. Quase todos os algoritmos, entretanto, utilizam a retro-propagação do erro (Rumelhart et al., 1986) na estimativa dos pesos das camadas ocultas. Um dos algoritmos mais utilizados no treinamento de uma rede é o algoritmo do gradiente, que, por estar fortemente associado ao algoritmo de retro-propagação, é confundido com este em alguns trabalhos.

O algoritmo de retro-propagação consiste, na verdade, de uma fórmula recorrente para obter a variação apresentada na saída de um dado neurônio, quando um dos pesos da rede é alterado. Esta derivada, como será visto, pode ser colocada em termos das derivadas dos neurônios da camada imediatamente anterior, o que caracteriza uma relação recorrente. Com isso a derivada é obtida para a saída apresentada na última camada, a seguir para a penúltima e assim por diante, de onde procede o nome de retro-propagação. O algoritmo do gradiente utiliza esta forma de retro-propagação para propagar o erro na saída da rede até as camadas ocultas, para com isso efetuar um ajuste nas matrizes de pesos destas camadas. Já o algoritmo de mínimos quadrados, também mostrado a seguir, utiliza a retro-propagação para obter a variação de cada sinal na saída com relação a todos os pesos que constituem a rede. Uma vez obtida esta variação, efetua-se um ajuste nos pesos de forma a minimizar o erro final da rede.

2.3.1 O Algoritmo de Retro-Propagação

Como foi dito, o algoritmo de retro-propagação consiste na obtenção da derivada parcial do valor apresentado na saída de um determinado neurônio com relação a um determinado peso de uma camada anterior. Considere então a componente i do vetor x^k apresentado na saída da camada k . A derivada desta componente com relação ao peso j associado a este neurônio é obtida derivando-se a Expressão 2.13, e resulta:

$$\frac{\partial x_i^k}{\partial w_{ij}^k} = f^{k'}(\bar{x}_i^k) x_j^{k-1} \quad (2.17)$$

onde $f^{k'}(\cdot)$ é a derivada da função de ativação da camada k com relação ao seu argumento. É claro que se fosse escolhido uma outra componente do vetor x^k (por exemplo, m), esta derivada seria nula, já que esta componente não depende do peso w_{ij} . A derivada pode ser posta em termos vetoriais, resultando:

$$\frac{\partial x^k}{\partial w_{ij}^k} = F^k \begin{bmatrix} 0 \\ \vdots \\ x_j^{k-1} \\ \vdots \\ 0 \end{bmatrix} \quad (2.18)$$

onde todas as componentes da última parcela são nulas, exceto a i -ésima componente, e F^k é uma matriz diagonal contendo as derivadas da função de ativação:

$$F^k = \frac{df^k(\bar{x}^k)}{d\bar{x}^k} = \begin{bmatrix} f^{k'}(\bar{x}_1^k) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & f^{k'}(\bar{x}_{n_k}^k) \end{bmatrix} \quad (2.19)$$

Sendo a função de ativação conhecida e contínua, a derivada pode ser avaliada em todos os valores de \bar{x}^k . Considere agora a variação da componente m com relação ao peso w_{ij} da camada anterior, $k-1$:

$$\frac{\partial x_m^k}{\partial w_{ij}^{k-1}} = f^{k'}(\bar{x}_m^k) \sum_{s=1}^{n_{k-1}} w_{ms}^k \frac{\partial x_s^{k-1}}{\partial w_{ij}^{k-1}}. \quad (2.20)$$

Veja que a derivada de x_s da camada $k-1$ é análoga à derivada obtida na Relação 2.17, desde que s seja igual a i ; para outros valores de s a derivada resulta nula e portanto:

$$\frac{\partial x_m^k}{\partial w_{ij}^{k-1}} = f^{k'}(\bar{x}_m^k) w_{mi}^k f^{k-1'}(\bar{x}_i^{k-1}) x_j^{k-2} \quad (2.21)$$

ou, na forma vetorial:

$$\frac{\partial x^k}{\partial w_{ij}^{k-1}} = F^k W^k F^{k-1} \begin{bmatrix} 0 \\ \vdots \\ x_j^{k-2} \\ \vdots \\ 0 \end{bmatrix} \quad (2.22)$$

onde, novamente, todas as componentes do vetor do segundo membro são nulas, exceto a i -ésima.

Estas expressões permitem estabelecer formas de recorrência, onde a derivada com relação ao peso de uma dada camada é obtida a partir de valores calculados durante a avaliação da derivada das camadas superiores, caracterizando com isso a retro-propagação. A derivada do elemento m da camada l (suposta como a última camada da rede), com relação ao peso j associado ao neurônio i da camada k fica então:

$$\frac{\partial x_m^l}{\partial w_{ij}^k} = \delta_{m,i}^k x_j^{k-1} \quad (2.23)$$

onde os escalares $\delta_{m,i}^k$ são obtidos de forma recorrente:

$$\delta_{m,i}^k = f^{k'}(\bar{x}_i^k) \sum_{s=1}^{n_{k+1}} \delta_{m,s}^{k+1} w_{si}^{k+1}, \quad (2.24)$$

com condição inicial dada por:

$$\delta_{m,i}^l = \begin{cases} f^{l'}(\bar{x}_i^l), & \text{se } i = m \\ 0, & \text{se } i \neq m \end{cases} \quad (2.25)$$

As relações obtidas acima podem ser traduzidas na forma de operações entre matrizes, onde a derivada do vetor de saídas da rede fica sendo dada por:

$$\frac{\partial x^l}{\partial w_{ij}^k} = \Delta^k \begin{bmatrix} 0 \\ \vdots \\ x_j^{k-1} \\ \vdots \\ 0 \end{bmatrix}, \quad (2.26)$$

e a matriz de retro-propagação, Δ^k vale:

$$\Delta^k = F^l W^l F^{l-1} W^{l-1} \dots F^{k+1} W^{k+1} F^k \quad (2.27)$$

ou, na forma recorrente:

$$\Delta^k = \Delta^{k+1} W^{k+1} F^k \quad (2.28)$$

desde que conheça-se, a princípio, o valor inicial da matriz Δ^l , que para a camada de saída, l , vale $\Delta^l = F^l$. As componentes da matriz de retro-propagação são dadas pelos $\delta_{m,i}^k$, fornecidos na Relação 2.24. Note que a matriz Δ^k possui dimensão de $n_l \times n_k$. Quando multiplicada pelo erro na saída, esta matriz constitui uma forma de retro-

propagação do erro, o que permite o ajuste dos pesos nas camadas ocultas da rede. Vale ainda salientar que a dimensão da matriz de pesos da camada k é de $n_k \times n_{k-1}+1$, já incluindo os pesos referentes ao patamar de ativação na última coluna. Neste caso, a matriz de derivadas da função de ativação, F^* , deveria possuir dimensão $n_{k-1}+1 \times n_{k-1}+1$, mas como o último elemento da diagonal é sempre nulo (por representar a derivada de uma constante), pode-se reduzir a dimensão da matriz de pesos (e também da matriz F) durante as multiplicações.

2.3.2 Treinamento com o Algoritmo do Gradiente

O algoritmo de treinamento conhecido como gradiente do erro (algumas vezes também chamado de retro-propagação) é sem dúvida o mais difundido e utilizado entre todos os outros algoritmos (Hunt et al., 1992). A sua principal vantagem reside no alto grau de paralelismo alcançado por este método na atualização dos pesos da rede. Sua origem remonta ao início da década de 60, quando Widrow e Hoff propuseram um circuito para uso em filtragem de sinal, o qual denominaram Adaline (Nguyen e Widrow, 1990). A saída única deste circuito era obtida pela adição das entradas ponderadas por pesos e, a seguir, submetendo este resultado a uma função não-linear (originalmente uma função sinal, ou degrau). Os pesos eram ajustados através de um processo de minimização do erro apresentado na saída, o que levava a uma solução de mínimos quadrados. Desta forma, uma rede Adaline pode ser visualizada como uma rede neural de uma única camada. O aprendizado via algoritmo do gradiente foi adaptado da generalização do método de Widrow-Hoff para redes de multi-camadas e funções de ativação diferenciáveis e não-lineares. Da mesma forma que na rede Adaline, o algoritmo do gradiente usa um critério de minimização do erro da rede no ajuste dos pesos durante a fase de aprendizagem. Como resultado, o algoritmo de ajuste caminha na direção do gradiente da função a ser minimizada, provocando uma convergência segura. Os pesos começam a ser ajustados a partir da última camada de neurônios, e com isso fornecem informações suficientes para o ajuste do peso na camada anterior e assim sucessivamente, de onde decorre o nome alternativo de retro-propagação

(Rumelhart, Hilton e Williams, 1986). A função a ser minimizada neste caso é o erro quadrático na saída da rede, ou seja:

$$J(t) = \frac{1}{2} \varepsilon(t)^T \varepsilon(t) \quad (2.29)$$

onde $\varepsilon(t)$ é o erro na saída da rede, no instante t , e vale:

$$\varepsilon(t) = y^d(t) - y(t) , \quad (2.30)$$

onde $y^d(t)$ é o vetor de saídas desejadas para a rede. O gradiente desta função com relação aos pesos da camada k de neurônios, já incluindo o patamar de ativação na última coluna, é definido pela matriz:

$$\nabla J^k = \begin{bmatrix} \frac{\partial J}{\partial w_{11}^k} & \cdots & \frac{\partial J}{\partial w_{1n_{k-1}+1}^k} \\ \vdots & \ddots & \vdots \\ \frac{\partial J}{\partial w_{n_k 1}^k} & \cdots & \frac{\partial J}{\partial w_{n_k n_{k-1}+1}^k} \end{bmatrix} . \quad (2.31)$$

O ajuste dos pesos se faz na direção do gradiente da função, conforme a equação:

$$W^k(t+1) = W^k(t) - \lambda \nabla J^k \quad (2.32)$$

e onde a matriz de pesos W^k é dada por:

$$W^k = \begin{bmatrix} w_{11}^k & \cdots & w_{1n_{k-1}+1}^k \\ \vdots & \ddots & \vdots \\ w_{n_k 1}^k & \cdots & w_{n_k n_{k-1}+1}^k \end{bmatrix} . \quad (2.33)$$

sendo que os patamares de ativação da camada k estão representados pelos pesos da última coluna, $w_{in_{k-1}+1}^k$, e λ é o parâmetro de convergência do aprendizado, compreendido normalmente entre 0 e 1. Quanto maior o valor de λ mais rápido o aprendizado, porém não se consegue boa estabilidade dos coeficientes quando a rede adquire conhecimento do sistema. Por outro lado, valores pequenos de λ acarretam tempos de aprendizado extremamente longos e freqüentemente podem fazer com que o processo de treinamento fique preso em um mínimo local.

A derivada da função com relação ao peso j associado ao neurônio i da camada k , é obtida derivando-se a Equação 2.29, e como o vetor de saída da rede, $y(t)$, é dado por:

$$y(t) = x^l(t) , \quad (2.34)$$

obtém-se a relação:

$$\frac{\partial J}{\partial w_{ij}^k} = -\varepsilon^T \frac{\partial x^l}{\partial w_{ij}^k} . \quad (2.35)$$

Aplicando, a seguir, a Relação 2.26, chega-se a:

$$\frac{\partial J}{\partial w_{ij}^k} = -\varepsilon^T \Delta^k \begin{bmatrix} 0 \\ \vdots \\ x_j^{k-1} \\ \vdots \\ 0 \end{bmatrix} , \quad (2.36)$$

que pode ser simplificado, resultando:

$$\frac{\partial J}{\partial w_{ij}^k} = -x_j^{k-1} \sum_{m=1}^{n_l} \varepsilon_m \delta_{m,i}^k . \quad (2.37)$$

Note que este resultado indica que a derivada da função depende exclusivamente da componente j do vetor de entrada da camada $k-1$ e da coluna i da matriz Δ^k . Isto sugere que o gradiente pode ser colocado em termos de operações entre matrizes, que resulta, após algumas manipulações:

$$\nabla J^k = -\Delta^{kT} \varepsilon x^{k-1T} . \quad (2.38)$$

A formulação obtida aqui, para o algoritmo do gradiente, difere em certos aspectos daquela difundida correntemente na literatura. A principal diferença se deve à incorporação, na formulação tradicional, do vetor de erros ε na matriz de retro-propagação, e, desta forma, propaga-se um vetor de erros e não, como aqui, uma matriz de variações, da seguinte forma:

$$\delta^{kT} = \delta^{k+1T} W^{k+1} F^k \quad (2.39)$$

onde δ^k , o erro retro-propagado, é um vetor de n_k colunas, e com condição inicial δ^l dada por:

$$\delta^{lT} = \varepsilon^T F^l . \quad (2.40)$$

A derivada da função, neste caso, resulta em:

$$\frac{\partial J}{\partial w_{ij}^k} = -\delta_i^k x_j^{k-1} \quad (2.41)$$

onde o escalar δ_i^k é a i -ésima componente do vetor δ^k . Por sua vez, o gradiente fica expresso pelo produto do vetor δ^k pelo vetor x^{k-1} , ou seja:

$$\nabla J^k = -\delta^k x^{k-1T}. \quad (2.42)$$

Embora o equacionamento anterior pareça, a princípio, mais trabalhoso pelo fato de ser necessário efetuar uma retro-propagação de uma matriz contra uma retro-propagação de um vetor, no entanto ele irá facilitar a implementação do algoritmo de treinamento por mínimos quadrados, como ficará claro mais adiante.

O algoritmo de retro-propagação resulta de um processo de otimização, e como todo processo deste tipo, ele pode levar o resultado para um mínimo local. Parâmetros de convergência elevados podem levar o processo de aprendizagem para uma situação instável, enquanto que valores pequenos resultam numa razão de convergência muito baixa. Para contornar este problema e também para acelerar o processo de aprendizagem, foram desenvolvidos métodos de ajuste de λ , de forma a se ter valores grandes no início do aprendizado e reduzi-lo à medida que a rede caminha para a solução absoluta. Ainda assim, e dada a grande quantidade de variáveis envolvidas, o processo de aprendizado de uma rede multi-camadas via retro-propagação é demasiadamente lento (Chen e Billings, 1992), exceto nos casos onde o treinamento é realizado num conjunto reduzido de valores.

2.3.3 Treinamento com o Algoritmo dos Mínimos Quadrados

Considere $y(t)$ o vetor (dimensão n_l) de saída da rede neural no instante t , função dos pesos dos neurônios $w(t)$ e do vetor de entrada $x(t)$, (dimensão n_0) no mesmo instante:

$$y(t) = f(x(t), w(t)) . \quad (2.43)$$

Neste caso, por simplicidade, optou-se por representar todos os pesos dos diversos neurônios que compõem a rede em um único vetor $w(t)$. Os algoritmos de estimação de parâmetros foram desenvolvidos para sistemas lineares e, como a função $f(\cdot)$ é não-linear, efetua-se uma linearização da mesma em torno do instante $t-1$, onde por hipótese são conhecidos os pesos. Desta forma, como o vetor de entradas $x(t)$ não varia durante o processo de atualização dos pesos e desprezando os termos de segunda ordem, tem-se:

$$y(t) = y(t-1) + \left. \frac{\partial y}{\partial w} \right|_{t-1} [w(t) - w(t-1)] . \quad (2.44)$$

Esta expressão pode ser posta em termos do erro apresentado na saída da rede, bastando para isso substituir o valor de $y(t)$ obtido da Equação 2.30. Reagrupando-se os termos, chega-se a:

$$z(t) = H(t) w(t) + \varepsilon(t) \quad (2.45)$$

onde a matriz de derivadas da saída da rede, $H(t)$, vale:

$$H(t) = \left. \frac{\partial y}{\partial w} \right|_{t-1} \quad (2.46)$$

e o vetor $z(t)$ é dado por:

$$z(t) = y^d(t) - y(t-1) + H(t) w(t-1) \quad (2.47)$$

A Equação 2.45 se encontra na forma de um sistema linear, e, neste caso, pode-se aplicar um método de estimação de parâmetros com informação a priori de forma a se obter o vetor de pesos w . O algoritmo dos mínimos quadrados é obtido a

partir da minimização de uma função objetivo representada por uma ponderação da informação a priori e do erro quadrático médio na saída, ou seja:

$$J(t) = \frac{1}{2} \left\{ \hat{w}^T(t-1) \bar{P}^{-1}(t) \hat{w}(t-1) + \varepsilon^T(t) R^{-1}(t) \varepsilon(t) \right\} \quad (2.48)$$

onde $\hat{w}(t)$ é a estimativa do vetor de pesos, a matriz $\bar{P}(t)$ pode ser associada à propagação da matriz de covariâncias $P(t)$ do erro de estimação de $w(t)$, e $R(t)$ pode ser associada com uma matriz diagonal das variâncias dos erros tolerados nas respectivas componentes de $\varepsilon(t)$ ao se tentar anulá-las durante a minimização da função objetivo (Gelb, 1974). A estimativa do vetor de pesos no instante t , $\hat{w}(t)$, é obtida recursivamente através da relação

$$\hat{w}(t) = (\underline{1} - K(t)H(t)) \hat{w}(t-1) + K(t) z(t) \quad (2.49)$$

onde $\underline{1}$ é a matriz identidade e $K(t)$ é o chamado ganho de Kalman, e vale:

$$K(t) = \bar{P}(t) H(t)^T [H(t) \bar{P}(t) H(t)^T + R(t)]^{-1} \quad (2.50)$$

Como não existe propagação de equações dinâmicas neste processo, adota-se um *fator de esquecimento*, λ , de forma a garantir que a covariância não diminua em demasia, conforme o sistema adquira conhecimento do vetor de pesos $w(t)$:

$$\bar{P}(t) = \frac{P(t-1)}{\lambda}, \quad 0 < \lambda \leq 1. \quad (2.51)$$

A atualização da matriz de covariância se faz na forma:

$$P(t) = [\underline{1} - K(t)H(t)] \bar{P}(t). \quad (2.52)$$

Reagrupando os termos da Expressão 2.49, chega-se a:

$$\hat{w}(t) = \hat{w}(t-1) + K(t) [z(t) - H(t) \hat{w}(t-1)] , \quad (2.53)$$

ou, mais resumidamente:

$$\Delta \hat{w}(t) = K(t) \varepsilon(t) , \quad (2.54)$$

que traduz a implementação da forma recursiva para o treinamento da rede neural. Em algumas referências (Chen e Billings, 1992), as Equações 2.51 e 2.52 são substituídas na Relação 2.50 e, após alguma manipulação algébrica, chega-se ao seguinte resultado:

$$K(t) = P(t) H(t)^T R(t)^{-1} \quad (2.55)$$

Embora mais simples que a Expressão 2.50, esta última não se encontra numa forma recursiva, tornando-se com isso impraticável sua implementação na forma de algoritmo.

Resta ainda informar as condições iniciais para a formulação recursiva, isto é, os valores dos pesos $w(t)$ e da matriz de covariância $P(t)$ no instante inicial. Normalmente adota-se para os pesos uma distribuição aleatória uniforme entre -1 e 1, e, para a matriz de covariância do erro de estimação o valor:

$$P(0) = \sigma_0^2 \mathbf{1} \quad (2.56)$$

cuja variância quadrática média, σ_0^2 , deve representar o desconhecimento acerca dos pesos no início do treinamento. É geralmente fixada entre 10 e 10000 (Norgaard; 1995, 1996). Por sua vez, a matriz de covariância do erro $\varepsilon(t)$, $R(t)$, geralmente é admitida como uma matriz identidade.

2.3.3.1 A Matriz de Derivadas Parciais

Foi visto que a identificação dos pesos da rede neural utilizando mínimos quadrados requer a avaliação da matriz de derivadas parciais do vetor de saída $y(t)$ com relação aos pesos da rede, conforme a Equação 2.44. Este problema é de fácil solução em redes de apenas uma camada, mas a complexidade e a quantidade de operações aumentam com o número de camadas. Neste caso, são usualmente empregados algoritmos recursivos para obter a matriz de derivadas. Considere-se então uma rede neural composta por l camadas, contendo n_k neurônios na camada k . Seja também x^k o vetor apresentado na saída da camada k , dado por:

$$x^k = \begin{bmatrix} f^k(\bar{x}^k) \\ 1 \end{bmatrix}, \quad (2.57)$$

onde $f^k(\cdot)$ é a função de ativação dos neurônios da camada k . O vetor de argumento é ponderado pelos pesos associados a esta camada, na forma:

$$\bar{x}^k = W^k x^{k-1}, \quad (2.58)$$

e onde a matriz de pesos W^k é dada por:

$$W^k = \begin{bmatrix} w_{11}^k & \cdots & w_{1n_{k-1}+1}^k \\ \vdots & \ddots & \vdots \\ w_{n_k}^k & \cdots & w_{n_k n_{k-1}+1}^k \end{bmatrix}. \quad (2.59)$$

onde os patamares de ativação da camada k estão representados pelos pesos da última coluna, $w_{in_{k-1}+1}^k$.

Derivando-se a Relação 2.34 e após substituir as Equações 2.57 e 2.58 obtém-se uma relação de recorrência entre as entradas e saídas das diversas camadas de neurônios, e assim pode-se obter a derivada dos parâmetros de saída da rede:

$$H_i^k = \frac{\partial x^l}{\partial w_i^k} = F^l W^l \frac{\partial x^{l-1}}{\partial w_i^k} \quad (2.60)$$

onde F^k é a matriz de derivadas da função de ativação da camada k com relação ao argumento \bar{x}^k , e é dada pela Expressão 2.19. Efetuando novamente a derivada na camada $l-1$, e assim sucessivamente, a Relação 2.60 resulta

$$H_i^k = F^l W^l F^{l-1} W^{l-1} \dots F^{k+1} W^{k+1} F^k \frac{\partial \bar{x}^k}{\partial w_i^k} . \quad (2.61)$$

Esta expressão pode ser colocada em termos de relações recorrentes, de tal forma que as matrizes de derivadas parciais vão sendo obtidas a partir da última camada até atingir a primeira camada:

$$H_i^k = \Delta^k \frac{\partial \bar{x}^k}{\partial w_i^k} \quad (2.62)$$

onde as matrizes Δ^k são obtidas da fórmula recorrente apresentada na Relação 2.28, com a condição inicial dada por $\Delta^l = F^l$. A derivada do vetor de argumentos da função de ativação da camada k com relação aos pesos w_i resulta na matriz de n_k linhas:

$$\frac{\partial \bar{x}^k}{\partial w_i^k} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{k-1} & x_2^{k-1} & \dots & x_{n_{k-1}}^{k-1} \\ 0 & 0 & \dots & 0 \end{bmatrix} \quad (2.63)$$

com apenas a linha i preenchida e todas as outras linhas nulas, já que as derivadas dos pesos dos demais neurônios da camada k não dependem dos pesos w_i . Se a matriz Δ^k for colocada na forma de suas componentes vetoriais, ou seja:

$$\Delta^k = \begin{bmatrix} \delta_1^k & \delta_2^k & \cdots & \delta_{n_k}^k \end{bmatrix} \quad (2.64)$$

onde cada componente possui n_l linhas, então a matriz H_i^k pode ser escrita como

$$H_i^k = \delta_i^k x^{k-1T}. \quad (2.65)$$

A matriz de derivadas parciais considerando todos os pesos da rede pode agora ser composta a partir das matrizes de derivadas dos neurônios, na forma:

$$H = \begin{bmatrix} H_1^1 & \cdots & H_{n_1}^1 & H_1^2 & \cdots & H_{n_2}^2 & H_1^3 & \cdots & \cdots & H_{n_l}^l \end{bmatrix}. \quad (2.66)$$

Esta matriz possui n_l linhas (correspondente ao número de variáveis na saída da rede neural) e cada coluna corresponde a um peso (ou limiar de ativação), que totalizam:

$$n_w = \sum_{m=1}^l n_m (n_{m-1} + 1). \quad (2.67)$$

As matrizes de peso são atualizadas conforme a relação:

$$W^k(t+1) = W^k(t) + \Delta W^k(t) \quad (2.68)$$

cuja matriz de incremento dos pesos, $\Delta W^k(t)$, é composta pelos elementos do vetor $\Delta \hat{w}(t)$, obtido da Relação 2.54:

$$\Delta W^k(t) = \begin{bmatrix} \Delta \hat{w}_{p+1} & \Delta \hat{w}_{p+2} & \vdots & \Delta \hat{w}_{p+n_{k-1}+1} \\ \Delta \hat{w}_{p+n_{k-1}+2} & \Delta \hat{w}_{p+n_{k-1}+3} & \vdots & \Delta \hat{w}_{p+2(n_{k-1}+1)} \\ \dots & \dots & \ddots & \vdots \\ \Delta \hat{w}_{p+(n_k-1)(n_{k-1}+1)+1} & \Delta \hat{w}_{p+(n_k-1)(n_{k-1}+1)+2} & \dots & \Delta \hat{w}_{p+n_k(n_{k-1}+1)} \end{bmatrix} \quad (2.69)$$

onde o índice p depende da camada considerada:

$$p = \sum_{m=1}^{k-1} n_m (n_{m-1} + 1) \quad (2.70)$$

Durante o processo de desenvolvimento do algoritmo computacional deve-se estar precavido com as diversas operações de matrizes cujas componentes são em grande parte nulas, como por exemplo a matriz F^k . Sendo o treinamento de uma rede neural um processo consideravelmente lento, deve-se procurar meios de forma a evitar cálculos desnecessários.

Veja que o algoritmo do gradiente é bastante semelhante ao algoritmo de mínimos quadrados recursivos, e as expressões resultantes são muito parecidas. A principal diferença entre os algoritmos se refere à função que leva ao método de otimização. Nos mínimos quadrados recursivos, o erro em cada saída é considerado individualmente no processo de minimização, enquanto que no algoritmo do gradiente o erro quadrático de todas as saídas é minimizado nas iterações. Se, por um lado, o esforço computacional é significativamente maior nos mínimos quadrados, por outro o treinamento é conseguido mais rapidamente.

CAPÍTULO 3

IDENTIFICAÇÃO E CONTROLE DE SISTEMAS POR REDES NEURAIAS

3.1 CONTROLE DE ATITUDE DE SATÉLITES

Um satélite em órbita da Terra consiste de um corpo livre de forças e torques, isto é, ele se encontra numa situação de imponderabilidade. Na verdade, a força gravitacional terrestre mantém o satélite numa órbita fechada, como se estivesse em queda livre, e, porisso, esta força não é sentida pelo objeto. Justamente por se encontrar num ambiente livre de torques, a orientação que este assume no espaço pode ser qualquer. Muitas vezes esta orientação, denominada de *atitude* do satélite, necessita ser controlada de forma a manter um determinado equipamento apontado numa dada direção. Por exemplo, o satélite pode estar provido de câmeras para efetuar imagens da Terra, e o eixo ótico deste equipamento precisa apontar para algum lugar da superfície do planeta. Eventualmente, o satélite precisa mover-se para apontar as células solares para o Sol, a fim de carregar as baterias. Estes movimentos na orientação do satélite são proporcionados pelo sistema de controle de atitude.

Controlar a atitude significa, portanto, gerar torques no satélite através de atuadores, segundo regras previamente estabelecidas. Visto que no ambiente espacial as perturbações que afetam a orientação do satélite possuem magnitude reduzida (não ultrapassam 10^{-4} Nm num satélite de comprimento típico da ordem de 1 m), também os atuadores devem gerar pequenos torques (máximo 10^{-2} Nm) de forma a não interferir em demasia na estabilidade da atitude. As informações que fornecem ao controlador meios para que este possa avaliar o grau de desvio da atitude com relação a uma referência são obtidas dos sensores que integram o sistema de controle. A Figura 3.1 mostra um esquema de controle freqüentemente empregado em satélites. Podem ser empregados tanto sensores de posição (magnetômetros, sensores de horizonte, sensores de estrelas, sensores solares, etc.) quanto sensores de velocidade (girômetros) e até

mesmo de acelerações (acelerômetros). As medidas de angulares de posição y e velocidade angular w efetuadas pelos sensores são processadas pelo estimador de estado de forma a obter estimativas do estado (atitude x e velocidade angular v) do satélite. O controlador compara, em seguida, estes valores com uma referência previamente fornecida do estado x^d e v^d , e comanda os atuadores de forma a reduzir ou eliminar o erro entre as estimativas e a referência. O estimador de estados pode não estar presente no controlador, dependendo dos tipos de sensores empregados, ou ainda pode estar reduzido a um simples elemento de filtragem dos ruídos dos sensores. Caso não se disponha de sensores de velocidade, e se o controlador necessita desta informação, ainda assim pode-se derivar as medidas de posição e utilizá-las no controle. Em essência, o controlador obtém o controle u a partir do erro de atitude e de regras de atuação previamente estabelecidas. A correspondência entre este erro e a magnitude do controle é função das características do sistema dinâmico, e por isso apresenta problemas de não-linearidade, ajustes do modelo matemático e identificação de parâmetros.

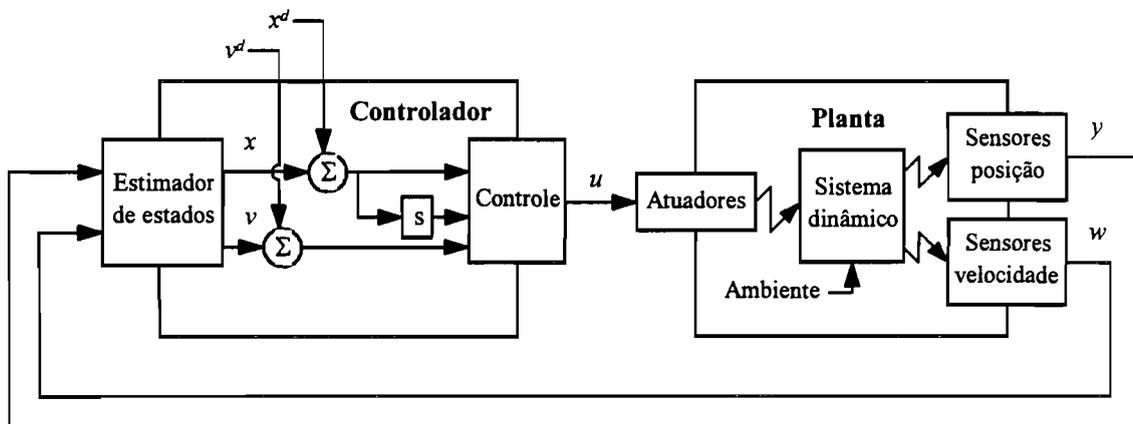


Fig. 3.1 - Controle de um sistema dinâmico

Uma forma bastante utilizada para representar sistemas dinâmicos consiste na formulação da equação diferencial que governa seu comportamento, em termos das variáveis de estado x , do vetor de saída (ou de observações efetuadas pelos sensores) y , e do termo forçante, ou controle u , todos funções do tempo t :

$$\dot{x} = \Phi[x(t), u(t)] \quad (3.1)$$

$$y = \Psi[x(t)] \quad (3.2)$$

As funções Φ e Ψ são mapeamentos não-lineares que traduzem a dinâmica do sistema e a relação entre as variáveis de estado e a saída, respectivamente. No caso da atitude de um satélite, as variáveis de estado são compostas pela posição (orientação) em termos de ângulos de Euler, ou então de quatérnions, e por sua velocidade angular. As equações dinâmicas do satélite proposto, no entanto, não são invariantes no tempo, devido à abertura dos painéis, que modifica a matriz de inércia. Para evitar este inconveniente, promove-se uma extensão do estado, que passa agora a incorporar o ângulo de abertura dos painéis em suas variáveis. O vetor de observações depende dos sensores utilizados para medir a atitude, e consistem genericamente de sensores de Sol, magnetômetros, sensores de horizonte ou infra-vermelhos, sensores de estrela, giroscópios, entre outros (Wertz, 1978).

Sistemas lineares e invariantes no tempo podem ser representados por:

$$\dot{x} = A x(t) + B u(t) \quad (3.3)$$

$$y = C x(t) \quad (3.4)$$

onde as matrizes A , B e C , são compostas por elementos constantes. A teoria hoje existente para controlar sistemas lineares está solidamente estabelecida, porém não garante a eficácia do controlador no caso de sistemas não-lineares. A solução encontrada foi utilizar a teoria de controle linear em sistemas linearizados, isto é, sistemas não-lineares onde foram aplicadas técnicas de linearização em torno de um ponto ou trajetória de operação e, a partir daí, usar métodos consagrados de estabilidade e obtenção de ganho. Ainda assim, o controlador pode falhar na presença de perturbações

não consideradas no modelo ou quando o estado do sistema encontrar-se longe da região linearizada.

Na tentativa de sobrepujar mais este inconveniente, investiu-se, em épocas recentes, no desenvolvimento de controladores adaptativos, que compensam variações na dinâmica do sistema (devido a envelhecimento ou perturbações ambientais) através de adaptações nos parâmetros de controle, de forma a garantir a robustez do controlador. Alguns sistemas, entretanto, resistem a esta abordagem simplista, por possuírem dinâmica altamente não-linear, e geram problemas devido a falhas na modelagem, na identificação das variáveis de estado, e na imprecisão do modelo linearizado (Lightbody et al., 1992). Estes fatores provocam a busca por controladores que empreguem técnicas não-lineares como forma de aumentar a confiabilidade e robustez.

São principalmente as não-linearidades do sistema dinâmico que tornam difícil de ser obtida a correspondência matemática entre o erro e o controle. Neste aspecto, justamente por proporcionarem excelente modelagem de sistemas não-lineares, redes neurais vêm apresentando bons resultados quando empregadas na identificação do controlador.

As redes são, de fato, versáteis ao lidar com grandes quantidades de informações provindas de elementos sensoriais. Pela formulação clássica, é muito difícil para um controlador absorver eficientemente e fazer uso de informações oriundas de diversos sensores, pois precisa de modelos que associem os sinais dos sensores com as variáveis de estado. Em contrapartida, a própria arquitetura paralela das redes facilita seu emprego onde essa gama de informações provindas de sensores é necessária. Em resumo, uma rede neural artificial apresenta como características principais a faculdade de aprendizagem ou adaptação, a possibilidade de processamento paralelo, e a capacidade de manipulação de grande quantidade de informações.

Foi sem dúvida o desenvolvimento de equipamentos computacionais com grande poder de cálculo que impulsionou e viabilizou a implementação de redes neurais para uso em controle de sistemas em tempo real. A velocidade de cálculo está diretamente relacionada com o número de neurônios (e também com o número de camadas) possível de ser empregado por uma rede neural. Este, por sua vez, relaciona-se com o grau de precisão na representação do sistema físico ou com a sofisticação da lei de controle. Portanto, quanto maior for a velocidade de cálculo, mais semelhante será a representação do sistema pela rede neural, e maior será a capacidade da rede de controlar sistemas mais complexos. Se, por um lado, esta complexidade tem aumentado nos últimos anos, por outro a demanda por controladores com requisitos mais acentuados de desempenho tem também se acentuado, não só para fazer frente à estes sistemas, mas também para torná-los mais robustos, inteligentes, e tolerantes a falhas. A solução parece apontar de forma definitiva a modelos e controles baseados em redes neurais.

3.2 IDENTIFICAÇÃO DE SISTEMAS DINÂMICOS

Quando se utilizam redes do tipo estática, o comportamento dinâmico do sistema é fornecido através de informações “atrasadas” de seu estado. Estas informações atrasadas podem ser visualizadas como uma realimentação realizada com valores provindos do sistema dinâmico ou ainda da própria rede neural (como mostrado na Figura 3.2), e, desta forma, a rede adquire um comportamento dinâmico semelhante a uma rede de Hopfield. O número destas informações também determina o grau de aproximação da rede, mas, intuitivamente, a quantidade destes parâmetros se relaciona com a ordem do sistema (o grau da equação diferencial que rege seu comportamento). Na verdade, a rede interpreta as informações atrasadas do estado como avaliações das derivadas do estado naquele instante. Billings et al., (1992), fazem importantes conclusões sobre a relação entre o número de entradas atrasadas e a ordem do sistema dinâmico (no caso, o grau da função não-linear utilizada). Sua principal conclusão é que o erro apresentado pela rede após o treinamento diminui conforme se aumenta o número

de entradas atrasadas, até atingir um valor mínimo. Este mínimo ocorre quando o número de entradas atrasadas for suficiente para satisfazer a ordem do sistema. A inclusão de mais valores atrasados, a partir daí, pode aumentar o erro, mas lentamente.

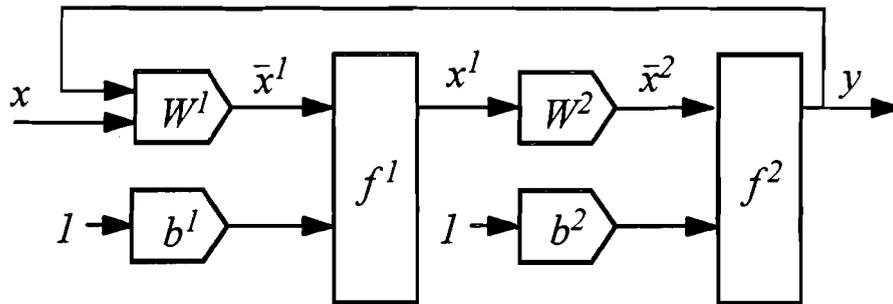


Fig. 3.2 - Rede estática com realimentação da saída.

Uma rede neural que aproxima tal sistema dinâmico deve ter como entradas não somente o controle, mas também as observações e as derivadas de ambos: controle e observações. Na falta destes últimos, valores atrasados destes parâmetros devem ser fornecidos: n valores atrasados das observações (ou do estado) e k valores do controle. Se a rede for realimentada, como mostrado na Figura 3.2, a saída $y^m(t)$ será uma função f_h , que aproxima o comportamento dinâmico do sistema:

$$y^m(t+1) = f_h(y^m(t), y^m(t-1), \dots, y^m(t-n)) \quad (3.5)$$

Hunt et al., 1992, (também em Warwick et al., 1992) fazem algumas distinções com relação à forma de utilização de redes neurais na identificação de sistemas, considerando a *modelagem direta*, e a *modelagem inversa*.

Na modelagem direta, a rede é treinada para se comportar como o sistema real, com entradas e saídas idênticas ao modelo físico. Nesta modelagem, rede e sistema são postos em paralelo, e utiliza-se o erro entre as suas saídas como estímulo do treinamento, conforme a Figura 3.3. A introdução da dinâmica do sistema pode, neste caso, ser realizada utilizando-se uma rede recorrente (realimentada) do tipo Hopfield, ou

ainda alimentar a rede com valores passados do estado e da entrada u , como discutido anteriormente. A rede apresenta na saída, neste caso:

$$y^m(t+1) = f_m(y^p(t), y^p(t-1), \dots, y^p(t-n), \\ u(t), u(t-1), \dots, u(t-k)). \quad (3.6)$$

Note-se que a rede está sendo alimentada com valores passados do sistema, e isso significa que é necessário alguma forma de armazenagem de informação. Além disso, a arquitetura do processo não é exatamente paralela devido à realimentação (representada pela linha tracejada na Figura 3.3), e por isso este modelo é algumas vezes chamado de série-paralelo. É claro que após o treinamento o erro cometido pela rede será bastante pequeno, o que viabiliza a realimentação da saída $y^p(t)$ do sistema pela saída obtida pela própria rede $y^m(t)$, tornando-a independente do sistema. Neste caso, a função fica:

$$y^m(t+1) = f_m(y^m(t), y^m(t-1), \dots, y^m(t-n), \\ u(t), u(t-1), \dots, u(t-k)) \quad (3.7)$$

No entanto, a utilização da saída da rede na realimentação só poderá ser realizada depois que a rede estiver suficientemente treinada, quando o erro entre a rede e o sistema for relativamente pequeno. Esta estrutura é preferível, segundo Hunt, quando o sistema apresentar considerável ruído na saída, já que este poderia provocar um deslocamento no sinal de saída da rede. Em certas situações, na verdade, a presença de ruído pode auxiliar a modelagem e identificação de sistema, conforme Billings et al., 1992. Neste trabalho, eles concluíram que a análise da correlação entre os valores de entrada e o erro apresentado pela rede neural fornece informações sobre a adequação dos parâmetros de entrada na representação do sistema. A correlação permanece limitada a valores pequenos se as entradas atrasadas da rede neural estiverem com a mesma ordem do sistema. Se estiverem subdimensionadas, a correlação resulta maior que os limites.

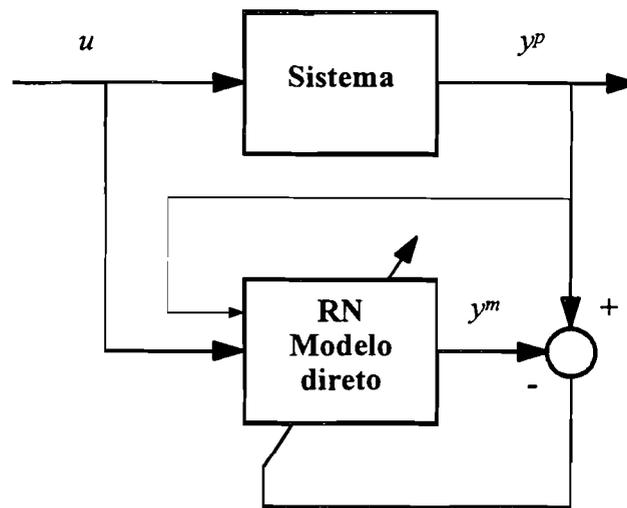


Fig. 3.3 - Identificação de modelo direto.

3.3 IDENTIFICAÇÃO DO MODELO INVERSO

O modelo da dinâmica inversa, ou simplesmente modelo inverso, pode ser obtido empregando-se diferentes estratégias de treinamento. O primeiro destes modelos, denominado método de treinamento inverso generalizado, é apresentado na Figura 3.4. Nele utiliza-se um sinal de referência u para excitar o sistema, e a sua resposta é usada como entrada na rede neural. A saída da rede é então comparada com o sinal de referência e a diferença é empregada no treinamento da rede, que é forçada, desta forma, a se comportar de forma inversa ao comportamento do sistema. O principal problema nesta abordagem é garantir que a rede seja treinada numa região bastante ampla dos valores possíveis da saída y^p , o que pode ser difícil de se conseguir sem um modelo matemático do sistema que realize um mapeamento da referência u na saída y^p . Infelizmente, até agora ainda não surgiu um método adequado que garanta a excitação do sistema na região de interesse.

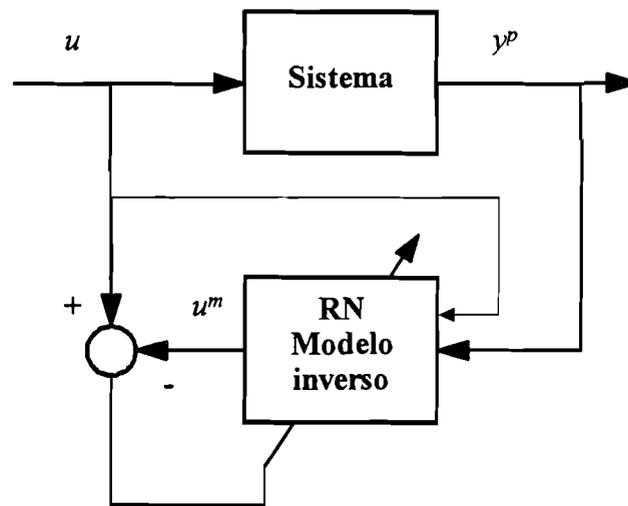


Fig. 3.4 - Identificação do modelo inverso pelo treinamento inverso generalizado.

Caso o comportamento dinâmico do sistema seja conhecido, mesmo que de forma aproximada, pode-se utilizar esta modelagem para realizar o treinamento da rede do modelo inverso, conforme mostrado por Zhang et al., 1995. Neste caso, utiliza-se somente o sinal da derivada da saída da planta com relação à sua entrada no procedimento de treinamento da rede. Visto que a avaliação da derivada através da própria planta poderia ser corrompida por ruídos e perturbações, justifica-se a utilização do modelo matemático simplificado. Caso não se disponha deste modelo, ou caso a saída apresentada pela planta não seja muito ruidosa, a variação da própria saída da planta com relação aos parâmetros de entrada poderá ser utilizada. A idéia de considerar apenas o sinal, e não a magnitude da derivada deve-se a eventuais erros de modelagem matemática ou ainda a presença de ruídos. De qualquer forma, o treinamento não é comprometido (exceto, talvez, em termos de velocidade de aprendizagem) já que o parâmetro λ de convergência do aprendizado, conforme a Equação 2.32, pode ser ajustado de forma a compensar o desconhecimento do módulo da derivada.

Outro problema ocorre quando o sistema apresenta mais de uma saída para a mesma entrada. Neste caso, é quase certo que a rede não irá apresentar a resposta correta, ou o processo de aprendizagem não irá convergir. Veja que é necessário um

prévio conhecimento do controle u de forma a garantir o aprendizado da rede. Em sistemas não-lineares, porém, nem sempre se dispõe de uma lei de controle válida para todo o domínio de atuação do sistema. Como o comportamento dinâmico do sistema depende do histórico dos sinais de controle, é natural supor que o modelo inverso também dependa destas informações. Em consequência, deve-se garantir que a rede receba como entrada k valores atrasados do controle, e a função de transferência, neste caso, assume a forma:

$$u(t) = f_u(y^p(t), y^p(t-1), \dots, y^p(t-n), \\ u(t-1), u(t-2), \dots, u(t-k)). \quad (3.8)$$

A rede com o modelo inverso, na forma com que foi obtida, pode ser utilizada como um controlador não-linear em malha aberta, cuja entrada é o valor de referência desejado na saída do sistema. No entanto, como foi dito, não há como garantir que o treinamento desta rede seja efetuado em todo o domínio de valores possíveis para a saída da planta, ou mesmo nas imediações da referência a ser mantida, pois desconhece-se como o sistema se comporta com relação à uma dada atuação do controle.

A segunda forma de obter o modelo inverso é através do método indireto, que consiste na utilização do modelo inverso simultaneamente para identificar e controlar o sistema. Como mostrado na Figura 3.4, elimina-se assim a necessidade de calcular previamente o sinal de controle u , pois este é obtido diretamente da rede que está sendo treinada. Este método apresenta como inconveniente o fato de eventualmente obter a mesma saída u^d e u , para entradas distintas entre y^d e y^p . A rede não mais consegue prosseguir no treinamento, embora o erro entre a referência e a saída do sistema ainda esteja elevado.

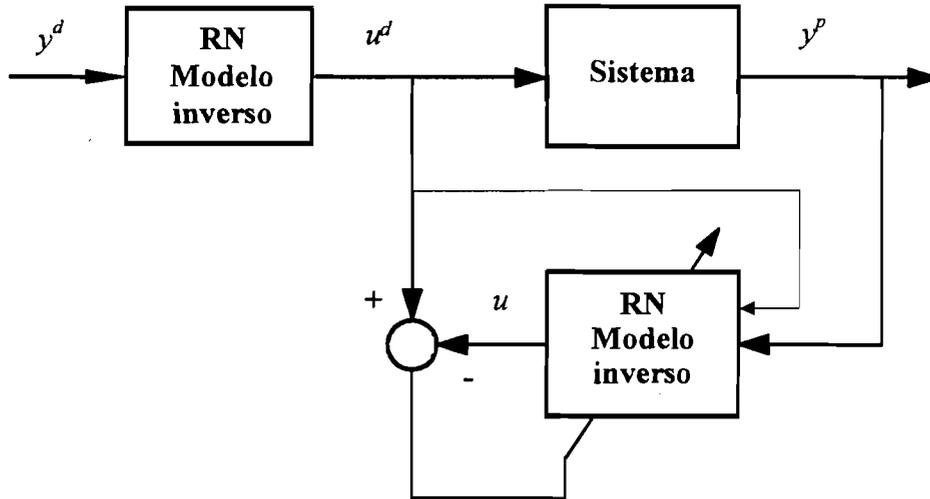


Fig. 3.5 - Identificação do modelo inverso pelo método indireto.

Uma forma de contornar estes problemas consiste no chamado aprendizado especializado inverso (Psaltis et al., 1988). Como mostra a Figura 3.6, o método possui duas redes neurais: a primeira recebe um sinal de referência igual ao estado desejado para o sistema. A segunda rede é essencialmente uma rede de modelo direto, em paralelo com o sistema. O treinamento é realizado com a diferença entre o sinal de referência y^d e a saída y^p do sistema, e utilizando a rede direta para retro-propagar o erro até a rede inversa. Este procedimento é análogo à determinação do Jacobiano do sistema, sem contudo necessitar do modelo matemático da planta para tal. Pode também ser utilizada a saída do modelo direto y^m no lugar de y^p , o que pode ser conveniente quando o treinamento não puder ser realizado com a própria planta, ou quando a saída do sistema contiver muito ruído. Deve-se tomar cuidado, neste caso, pois a representação do modelo inverso só será exata se o modelo direto também o for. Erros no modelo direto podem se propagar pelo modelo inverso durante a aprendizagem deste, levando a rede a se comportar de forma diferente do sistema real.

As vantagens do modelo especializado inverso com relação ao modelo inverso reside principalmente no sinal de referência, mais fácil de ser obtido do que o sinal de controle da planta, pois consiste no próprio estado do sistema. Hunt (Hunt et al.,

1992) não faz menção, no entanto, às dificuldades de se produzir uma trajetória de referência que não implique em respostas tipo degrau na saída da planta, o que em sistemas dinâmicos reais implicaria em controladores com ganhos infinitos. Por outro lado se o sistema não responder à entrada no devido tempo (resposta atrasada), o erro pode levar a um treinamento inadequado. Não se pode esquecer ainda que tanto a rede como o sistema de controle e a amostragem da planta são discretizados, e qualquer sinal discretizado apresenta saltos caso não se empreguem filtros adequados.

Há ainda um outro problema a ser resolvido: o modelo inverso, descrito pelo inverso da equação do modelo direto (Equação 3.6), necessita do conhecimento prévio do estado no instante $t+1$. Como normalmente a referência é conhecida antecipadamente, sugere-se a substituição do estado no instante $t+1$ pela referência, visto que ambos estarão próximos, no caso de representação correta do modelo inverso pela rede neural, ou seja:

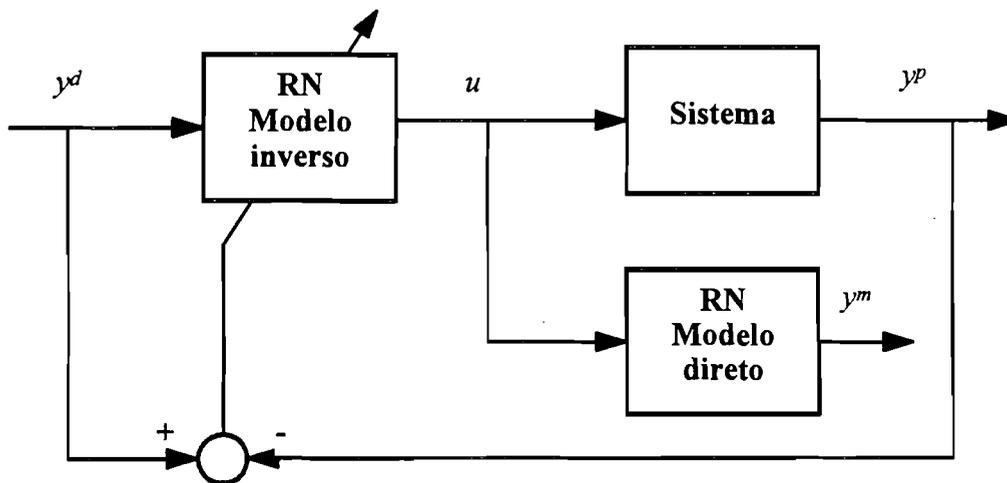


Fig. 3.6 - Modelo especializado inverso.

$$u(t) = f_m^{-1}(y^d(t+1), y^p(t), y^p(t-1), \dots, y^p(t-n), \\ u(t-1), u(t-2), \dots, u(t-k)) \quad (3.9)$$

3.4 CONTROLE ATRAVÉS DO MODELO INVERSO

Deve-se realçar que os modelos inversos são essencialmente controladores em malha aberta. A menos que o treinamento seja realizado continuamente, alterações nas características dinâmicas da planta não serão compensadas pelo controle. Para contornar esta desvantagem, vários modelos de realimentação do erro surgiram nos últimos anos, como o modelo de controle por referência (Narendra e Parthasarathy, 1990); modelo de controle interno (Garcia e Morari, 1982) e o controle preditivo (Hunt et al., 1992), entre outros. No modelo de controle interno, IMC, um modelo do sistema (RN Modelo na Figura 3.7) é treinado para se comportar como o sistema dinâmico direto da planta. Uma segunda rede, a rede de controle (RN Controle) é a seguir treinada segundo o modelo inverso, usando no treinamento o erro retro-propagado do modelo direto. Como as redes não são alimentadas com informações que permitem a modelagem das perturbações d que afetam o comportamento do sistema, elas não conseguem eliminar os erros na trajetória devido aos efeitos destas perturbações. Na verdade, o conhecimento adquirido pela rede retrata o comportamento dinâmico “médio” da planta, ou seja, o comportamento que o sistema teria na ausência de perturbações. A diferença entre a trajetória real da planta e a trajetória fornecida pela rede neural direta será então utilizada na realimentação de forma a corrigir o estado e compensar os efeitos das perturbações. Visto sob um outro ângulo, este tipo de controle obriga o sistema a assumir o comportamento dinâmico da rede de modelo.

Os procedimentos de obtenção do modelo inverso nem sempre apresentam bons resultados no que se refere ao grau de aproximação, generalidade e abrangência da solução. Além das dificuldades já mencionadas de existência ou não da dinâmica inversa, da possibilidade de haver mais de uma solução para uma dada trajetória, a própria complexidade do sistema pode levar a uma inversa não representativa, ou mesmo à sua inexistência. De fato, na literatura ainda não existem trabalhos sobre aplicações de redes neurais em controle de sistemas cuja dinâmica possua uma complexidade elevada (o que pode ser traduzido por um número maior de graus de

liberdade, maior número de parâmetros de entrada e saída, algoritmo matemático mais extenso, etc.). No decorrer deste trabalho pôde-se constatar que os métodos citados para treinamento do modelo inverso ficaram freqüentemente sujeitos à instabilidades introduzidas pela realimentação de um sinal de controle errático. As instabilidades acabam por levar o sistema a uma região no qual o modelo direto é pouco representativo (por não ter sido treinado convenientemente) e, como consequência, o treinamento do controlador não mais consegue convergir.

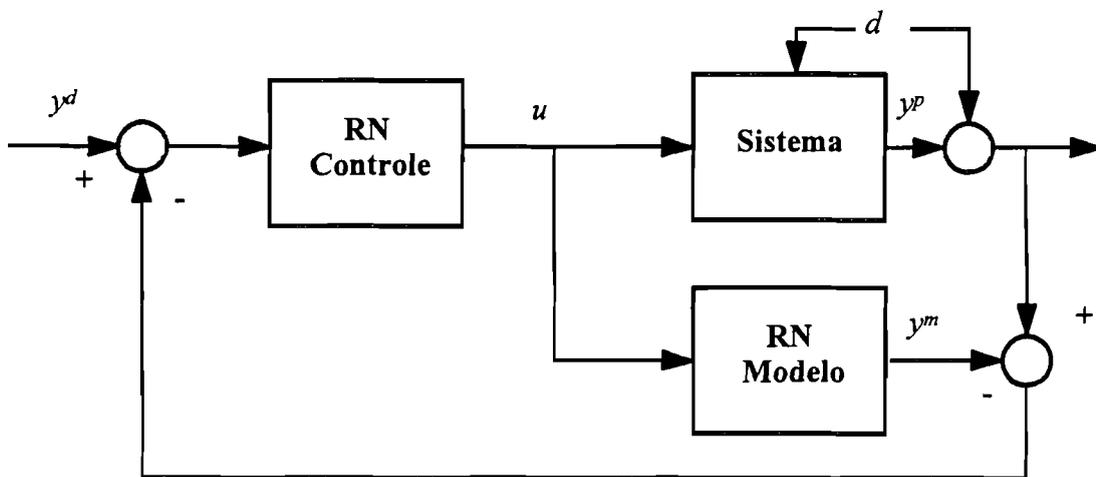


Fig. 3.7 - Controlador IMC.

A solução apresentada por Kawato et al., 1988 (também em Kawato, 1990 e Oliveira e Yoneyama, 1994) denominada de aprendizado com realimentação do erro, contorna, pelo menos em princípio, o problema da incontrolabilidade durante o aprendizado. Conforme mostrado na Figura 3.8, o erro da planta com relação à trajetória de referência é realimentado na malha de controle, mantendo, assim, o sistema numa situação estável durante o processo de aprendizagem. A rede neural do modelo inverso ainda desempenha a função de um controlador direto (*feedforward*), mas o sinal do controle da rede, u^c , é adicionado ao sinal do controle realimentado, u^d (geralmente um controlador proporcional, derivativo e integral - PID). A informação de treinamento é dada pelo próprio sinal do controle realimentado, garantindo assim a minimização do

erro de realimentação tão logo a rede adquira o conhecimento da dinâmica inversa. A rede, desta forma, ao invés de se comportar como um mero controlador PID faz mais do que isso, pois na verdade se adapta de forma a minimizar a realimentação do erro, conforme ponderou Nascimento Jr. (Nascimento Jr., 1994). Durante o processo de aprendizado o controle da planta passa gradativamente de realimentado para direto. As informações sobre a dinâmica do sistema podem fazer parte do conjunto de parâmetros de entrada da rede, juntamente com valores atrasados no tempo. O mesmo é válido também para a trajetória de referência, y^d .

Ainda que as vantagens deste procedimento sejam evidentes, alguns problemas podem surgir durante o treinamento. A rede pode, por exemplo, competir com o controlador PID, de tal forma que o sinal resultante u se anule, contrabalançando a realimentação do erro entre o sistema y^p e o sinal de referência y^d .

Nas aplicações de identificação e controle, é comum o emprego de redes neurais estáticas com multi-camadas, com realimentação da planta somente durante a fase de treinamento (Antsaklis, 1992). É também freqüente a utilização do método do gradiente (com retro-propagação do erro) para o aprendizado, embora normalmente este seja um processo demasiadamente lento, demandando bastante tempo de processamento para convergir. Utiliza-se normalmente a função sigmóide neste tipo de aplicação, mas podem ser encontradas também as funções gaussiana e a sinal. Como a maioria das implementações são em sistemas não-lineares (dificultando com isso saber a priori qual deva ser o sinal de controle), treina-se a rede numa determinada classe de pares de entrada-saída com diferentes condições iniciais. Após o treinamento, a precisão de modelagem atingida pela rede depende, é claro, do número de neurônios e da dinâmica do sistema. O número de neurônios e o número de camadas são normalmente escolhidos de forma empírica, já que ainda não existe método que estabeleça a geometria da rede em função da precisão estabelecida. Quanto maior o número de camadas, melhor a representação do sistema modelado, maior a complexidade da rede e seu treinamento, e

menor a sua capacidade de generalização. Portanto, há um compromisso entre a fidelidade do modelo e o tamanho da rede (Nguyen e Widrow, 1990).

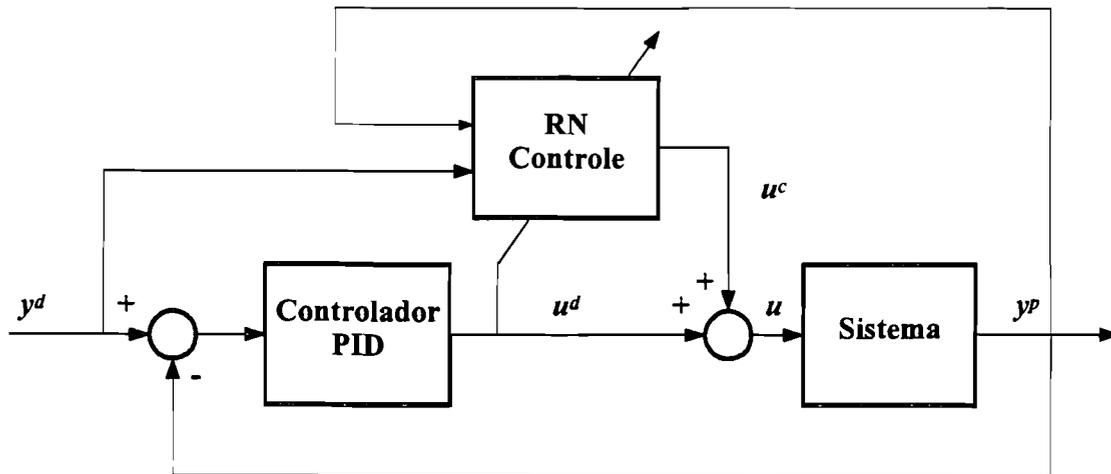


Fig. 3.8 - Treinamento do modelo inverso pela realimentação do erro.

Os requisitos impostos ao controle de atitude de satélites são, com frequência, bastante exigentes no que se refere à precisão de apontamento, estabilidade da atitude e tolerância aos erros. Apontamentos da ordem de 1 segundo de arco devem ser conseguidos em certos satélites, como por exemplo nos telescópios espaciais (Hubble, ISO, ROSAT, etc.), embora a grande maioria não tenha requisitos tão rígidos, necessitando precisões ao redor de 1 grau. Mesmo nestas condições, o apontamento e estabilidade só podem ser conseguidos com controladores em malha fechada, com realimentação do erro. Tanto o modelo IMC quanto o modelo com realimentação do erro apresentam características de compensação do erro, embora tanto em um quanto no outro a rede neural não trabalhe diretamente como erro cometido pela planta. O modelo IMC, pelo menos teoricamente, consegue compensar os efeitos das diversas perturbações ambientais, principais causas no desvio da atitude de satélites com órbita baixa (abaixo de 1000 km de altitude). Decidiu-se, assim, testar estes métodos no controle de atitude de um satélite com geometria variável. Como o sistema físico da planta não pode ser obtido em Terra, as equações dinâmicas do sistema, necessárias na fase de treinamento e simulação do controle devem também ser simuladas

computacionalmente. O capítulo seguinte irá descrever o modelo dinâmico e cinemático do satélite, incluindo a dinâmica de abertura dos painéis. A seguir, será mostrada a implementação dos modelos direto e inverso no computador e os principais resultados obtidos.

CAPÍTULO 4

MODELO DINÂMICO E CINEMÁTICO DO SATÉLITE

4.1 EQUAÇÕES DINÂMICAS DE UM CORPO RÍGIDO

Um satélite em órbita terrestre fica sujeito a ação de forças gravitacionais, ambientais e próprias, isto é, geradas pelos equipamentos do próprio satélite. A força dominante é a atração gravitacional exercida pela Terra, responsável por manter o satélite numa órbita elíptica, ou kepleriana, com o centro terrestre em um dos focos da elipse. As demais forças, com pelo menos três ordens de magnitude menor do que aquela exercida pela Terra, modificam a órbita de tal forma que esta agora somente poderá ser considerada elíptica em termos médios. Entre estas forças, consideradas perturbações ao movimento kepleriano, incluem-se também algumas de origem gravitacional, como por exemplo aquelas oriundas da distribuição não uniforme da massa da Terra, as atrações exercidas pelo Sol, pela Lua e também pelas marés terrestres (embora em menor escala). Já as forças ambientais originam-se de perturbações causadas no movimento da espaçonave por fatores do ambiente espacial, como por exemplo o arrasto atmosférico, a pressão de radiação solar, forças e torques de origem eletromagnética, etc. (Wertz, 1978; Carrara, 1982). Estas forças e torques podem ainda ser classificados quanto ao tipo em internos e externos. Forças e torques internos são exercidos por uma parte do satélite em outra parte do mesmo e, portanto, compensam-se entre si, com efeito nulo. Já as forças externas não são compensadas e podem por isso ocasionar mudanças no movimento. Entre aquelas de origem externa destacam-se as já citadas forças ambientais e gravitacionais. Também são externos os esforços que têm origem nas atuações do sistema de controle de órbita e atitude, justamente para compensar os efeitos das perturbações ambientais. Na ausência de esforços externos, um corpo livre no espaço tende a conservar sua quantidade de movimento angular, ou momento angular L . A lei da conservação do momento angular, além disso, afirma que a variação temporal do vetor momento angular em relação a um ponto fixo ou o centro de

massa é idêntico à atuação dos torques externos N_{ext} ao corpo (e, neste caso, inclui-se aqui também os torques de origem ambiental e gravitacional), na forma (Crandall et al., 1968; Wertz, 1978).

$$\frac{dL}{dt} = \sum N_{ext} . \quad (4.1)$$

No caso de um satélite artificial, a resultante dos torques externos pode ser decomposta, conforme foi dito anteriormente, na parcela devida aos torques ambientais e gravitacionais, ou de perturbações, N_{pert} e a parcela devida aos torques externos oriundos do sistema de controle de atitude, N_{cont} .

As equações dinâmicas dependem do sistema de referência no qual são expressas. Por isso é importante estabelecer uma sistemática de representação de vetores, no qual fique evidente de forma clara em qual sistema o vetor é expresso. Para não sobrecarregar nos índices, visto que será necessário reservá-los para identificar os diferentes corpos que compõem o satélite com articulações, serão utilizados índices superiores (ou sobrescritos) para realçar o sistema de coordenadas no qual estão representadas as componentes de um dado vetor. Assim, o vetor r_k^o indica, por exemplo, a posição de um dado ponto do elo k do satélite (como será visto adiante), medido em relação ao sistema k do próprio elo, porém expresso no sistema de coordenadas o (fixo ao corpo principal do satélite).

O momento angular é dado pelo produto vetorial entre o vetor posição em relação a um dado ponto, e o vetor quantidade de movimento de um ponto de massa. No caso de um corpo com dimensões finitas, o momento angular deve ser integrado em elementos infinitesimais de massa. Devido ao movimento do corpo em relação a um sistema de coordenadas inerciais, esta integral pode ser bastante complexa. Por isso, é mais conveniente expressar o momento angular num sistema de coordenadas fixo ao corpo, no qual são conhecidos os momentos de inércia. Considere-se, então, um sistema

de eixos inerciais i com origem no centro de massa de um corpo rígido, e um outro sistema o , fixo ao corpo, também com origem no centro de massa. O momento angular L^o com relação ao centro de massa, no sistema fixo ao corpo é dado por (Wertz, 1978; Crandall et al., 1968):

$$L^o = \int_V r^o \times v^o dm_o = \int_V r^o \times \dot{r}^o dm_o . \quad (4.2)$$

onde r^o e \dot{r}^o (ou v^o) representam a posição e a velocidade (com relação a um sistema inercial), do elemento de massa dm_o . O momento angular deve ser integrado em todo o volume V ocupado pelo corpo. A derivada desta integral fornece, já utilizando a notação para o produto vetorial apresentada no Apêndice A:

$$\dot{L}^o = \int_V \Omega(r^o) \ddot{r}^o dm_o . \quad (4.3)$$

onde $\Omega(\cdot)$ representa a matriz do produto vetorial. Como a aceleração \ddot{r}^o é tomada num referencial girante, deve-se então substituí-la pela Equação A.19 (Apêndice A). No caso do corpo rígido, após a aplicação de algumas propriedades do produto vetorial, obtém-se a conhecida formulação de Euler:

$$\dot{L}^o = I_o \dot{\omega}_o - \Omega(\omega_o) I_o \omega_o . \quad (4.4)$$

sendo que I_o é a matriz de inércia do corpo, e ω_o a sua velocidade angular, expressa no próprio referencial fixo ao corpo. Utilizou-se, aqui, uma notação diferente da empregada no Apêndice A. Se, por um lado, a compreensão do significado das variáveis é fundamental no citado apêndice, por outro a introdução de muitos índices pode dificultar o entendimento do equacionamento obtido aqui. No entanto, para que possa ser feita uma analogia, a velocidade angular ω_o expressa a velocidade do corpo (ou do sistema de coordenadas o fixo ao corpo) em relação ao sistema inercial i .

Alguns satélites são equipados com rodas de reação ou volantes de inércia, dispositivos compostos por um motor que aciona uma roda girante. Ao aplicar um torque (de aceleração ou frenagem) na roda, o motor sofre um torque de mesma intensidade e sentido contrário. O sistema de controle usa então este torque para realizar mudanças na atitude do satélite. Note-se porém que sendo este um torque interno, não há alteração da quantidade de momento angular total do satélite como um todo. Há sim, uma troca entre a parcela de momento angular devido ao corpo do satélite e aquela devido ao volante. Como os instrumentos de medida e atuação do sistema de controle estão fixados na carcaça do satélite, o resultado desta troca de momento angular é idêntico ao ocasionado por um torque externo. Já que o momento angular total do satélite não se altera com a reação provocada pela roda, perturbações tendenciosas, isto é, que mantêm com o tempo uma mesma orientação, provocam um acúmulo de momento angular, de tal forma que a roda ou volante acabam se saturando. Neste ponto, quando atingem a velocidade máxima suportada pelo equipamento, torna-se necessário a atuação de um dispositivo que produza um torque externo que reduza ou anule o momento angular total. A vantagem de dispositivos como as rodas de reação ou os volantes de inércia deve-se à facilidade de gerar pequenos torques (da ordem de 10^{-4} Nm) comparativamente com os grandes torques causados pelos motores à reação ou motores de hidrazina, praticamente incapazes de gerar torques abaixo de 10^{-2} Nm. Desta forma, o sistema de controle utiliza as rodas para efetuar pequenos ajustes na atitude do satélite e os motores à reação para, ocasionalmente, provocar a dessaturação das rodas de reação. Um satélite que possua rodas ou volantes já não pode ser considerado um corpo rígido embora suas partes constituintes o sejam. A inclusão do efeito das rodas na equação do momento angular é equivalente a aumentar os graus de liberdade ou a ordem do sistema, que fica então:

$$\dot{L}^o = I_o \dot{\omega}_o^o - \Omega(\omega_o^o) I_o \omega_o^o - (\Omega(\omega_o^o) h^o + N_w) \quad (4.5)$$

$$\dot{h}^o = N_w \quad (4.6)$$

onde L^o agora representa o momento angular da carcaça, h^o é o momento angular resultante das rodas e volantes e N_w é o torque aplicado nas rodas pelo sistema de controle.

4.2 EQUAÇÕES PARA UM CORPO ARTICULADO

No entanto, se o satélite não for rígido (possuir apêndices flexíveis ou geometria variável), as equações acima não são mais válidas, e torna-se necessário obter uma nova formulação para este caso, o que pode ser bastante complexo em virtude da geometria do satélite. Neste trabalho, será feita a suposição de que o satélite é formado por diversas estruturas rígidas unidas por juntas articuladas com um corpo principal, ou seja, um corpo com geometria variável.

Considere-se então um corpo unido a outro através de uma junta de articulação, conforme a Figura 4.1. Normalmente, a velocidade das articulações é conhecida pois existe um modelo de atuador que fornece a posição $\theta(t)$, sua velocidade ou mesmo o torque na junta. Por exemplo, suponha um satélite com apontamento geocêntrico (que possua uma face sempre voltada para a Terra) e que tenha um painel solar que acompanhe a direção do Sol conforme o satélite movimenta-se em sua órbita. Neste caso, a velocidade angular do painel com relação ao satélite é aproximadamente igual à velocidade angular orbital do satélite, porém em sentido contrário. Como a velocidade angular orbital é praticamente constante em órbitas de baixa excentricidade, pode-se supor que a velocidade angular do painel $\dot{\theta}(t)$ com relação ao corpo do satélite é também constante. Veja no entanto que esta mesma velocidade angular com relação a um referencial inercial é quase nula.

Para obter a equação da dinâmica do satélite, será suposto aqui que o mesmo é formado por diversos corpos rígidos, denominados elos, articulados com relação ao corpo principal do satélite através de juntas rotativas. Em robótica, costuma-se utilizar a notação de Denavit-Hartenberg (Asada e Slotine, 1985), que estabelece regras para se

compor os diversos sistemas de coordenadas das juntas. Por esta notação, cada junta recebe um sistema de coordenadas retangulares, cuja orientação dos eixos é realizada de forma a facilitar a transposição das diversas matrizes de rotação que compõem o movimento dos braços. Infelizmente, embora esta sistemática seja eficiente para compor movimentos, ela se revela ineficaz para a composição da dinâmica, por não levar em conta a posição do centro de massa de cada elo. Outra desvantagem desta formulação se deve ao fato de que a base de um braço mecânico absorve a reação aos torques aplicados nas juntas, o que não ocorre num corpo livre no espaço. No satélite torna-se imperativo, portanto, que os sistemas de coordenadas intermediários sejam fixados no centro de massa de cada um dos corpos articulados, assegurando desta forma que possam ser computados o centro de massa e matriz de inércia do conjunto.

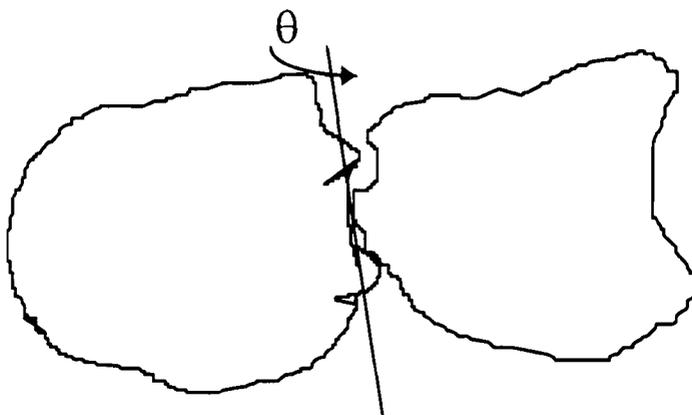


Fig. 4.1 - Satélite com articulações.

Outro fato comum em robótica é o aparecimento de juntas prismáticas (ou telescópicas), onde o deslocamento do braço se faz ao longo do eixo da junta. Embora este tipo de junta seja bastante raro em aplicações espaciais (mesmo nos braços mecânicos espaciais), a formulação obtida aqui pode ser facilmente estendida para acomodar também as juntas prismáticas. Será suposto também que em cada junta unem-se exclusivamente dois corpos (ou elos), sendo um deles chamado elo antecessor da junta e o outro o elo sucessor da junta. No caso estudado aqui, o elo antecessor será sempre o corpo principal do satélite. No entanto, uma formulação mais geral com elos

intermediários (simultaneamente antecessores e sucessores) poderá ser obtida, embora com uma dificuldade algébrica considerável. Tal formulação foge, contudo, do objetivo principal deste trabalho, e assim não será analisada aqui.

Os elos que se unem ao corpo do satélite serão numerados seqüencialmente, a partir da unidade. Será admitido que n elos estão unidos ao corpo principal, conforme a Figura 4.2. Os índices inferiores referem-se ao elo considerado (k para os elos, $k = 1, \dots, n$; o para o corpo principal), exceto no caso de vetores, que empregarão o índice superior para representar o sistema no qual são referenciados. Serão utilizados três sistemas de coordenadas para estabelecer o equacionamento dinâmico, mostrados na Figura 4.3: o sistema de direções inerciais i , com origem no centro de massa do conjunto (corpo principal mais elos); o sistema do satélite, o , fixo ao corpo principal e com origem em seu centro de massa e um sistema de coordenadas para cada elo, k , $k = 1, \dots, n$, com origem no centro de massa do elo correspondente. Admite-se, além disso, que são conhecidas as massas m_o e m_k , e as matrizes de inércia I_o e I_k do corpo do satélite e dos elos, respectivamente, com relação aos seus sistemas de coordenadas. A posição da junta k deve ser fornecida através dos vetores a_{ok}^o e a_{ko}^k , com relação aos sistemas do satélite e do elo k , respectivamente. Finalmente, tem-se como dada a velocidade angular ω_k^o de cada elo k com relação ao sistema o , segundo a direção da junta rotativa, porém medida e expressa no sistema de coordenadas do corpo do satélite. Novamente, para simplificar a notação, empregou-se somente o índice k , quando, de acordo com a regra empregada no Apêndice A, deveria ser k/o .

Em analogia às equações da dinâmica para corpos rígidos, devem-se expressar todos os vetores no sistema do corpo do satélite, sistema este no qual as medidas dos diversos sensores de atitude são referenciadas, e no qual é conhecida a matriz de inércia. Como a origem do sistema de direções inerciais é fixada ao centro de massa r_{cm} do conjunto, necessita-se da posição deste centro com relação ao sistema o a cada instante:

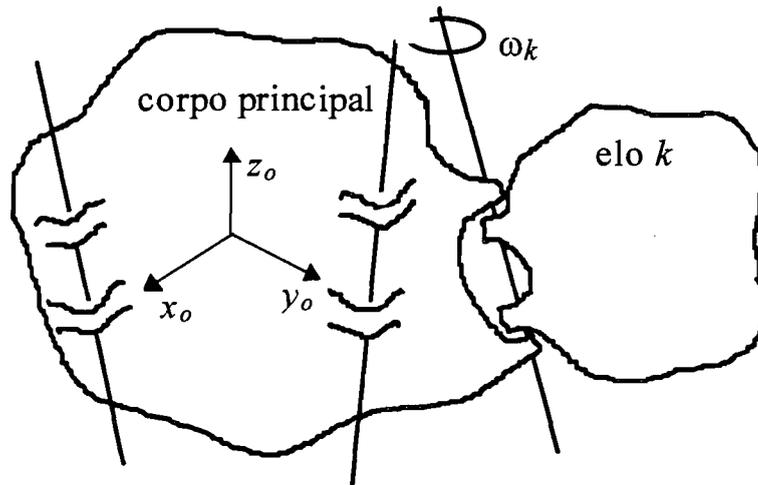


Fig. 4.2 - Representação das juntas e elos do satélite.

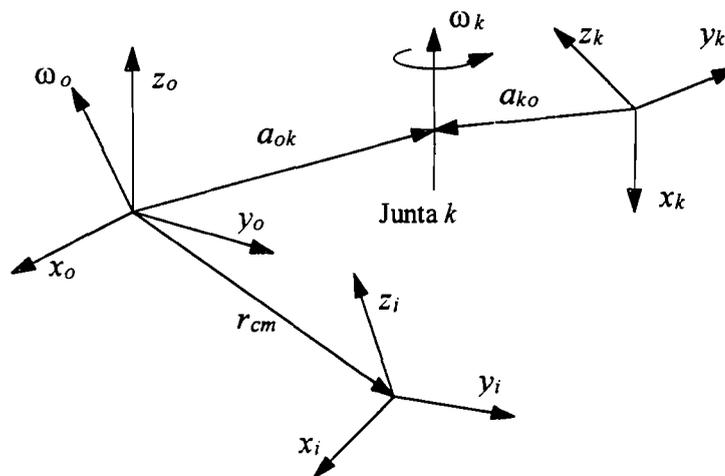


Fig. 4.3 - Sistemas de coordenadas no satélite com geometria variável.

$$r_{cm}^o = \sum_{k=1}^n \mu_k (a_{ok}^o - a_{ko}^o) \quad (4.7)$$

onde a massa reduzida de cada elo, μ_k , vale:

$$\mu_k = \frac{m_k}{m_o + \sum_{k=1}^n m_k} \quad (4.8)$$

A derivada do momento angular, segundo a Relação 4.3 pode agora ser decomposta nos constituintes do satélite, na forma:

$$\dot{L}^o = \int_{V_o} \Omega(r_o^o) \ddot{r}_o^o dm_o + \sum_{k=1}^n \int_{V_k} \Omega(r_k^o) \ddot{r}_k^o dm_k . \quad (4.9)$$

onde os vetores r_o e r_k representam a posição do elemento de massa dm_o e dm_k , do corpo principal do satélite e do elo k , respectivamente, com relação ao centro de massa do conjunto, e expressos no sistema o . As integrais são efetuadas em todo no volume V_o e V_k do satélite. Se ρ_o^o e ρ_k^o representarem, respectivamente, a posição de um elemento de massa do corpo principal e do apêndice k , com relação ao seu próprio sistema de coordenadas, porém expressas ambas no sistema o , então a posição destes elementos com relação ao centro de massa do conjunto, conforme a Figura 4.3, fica:

$$r_o^o = \rho_o^o - r_{cm}^o \quad (4.10)$$

$$r_k^o = \rho_k^o - r_{cm}^o + a_{ok}^o - a_{ko}^o . \quad (4.11)$$

Derivando-se estas expressões duas vezes, e substituindo na equação da variação do momento angular, obtém-se:

$$\begin{aligned} \dot{L}^o = & -M \Omega(r_{cm}^o) \ddot{r}_{cm}^o + \sum_{k=1}^n m_k \Omega(a_{ok}^o - a_{ko}^o) (\ddot{a}_{ok}^o - \ddot{a}_{ko}^o) + \\ & + \int_{V_o} \Omega(\rho_o^o) \ddot{\rho}_o^o dm_o + \sum_{k=1}^n \int_{V_k} \Omega(\rho_k^o) \ddot{\rho}_k^o dm_k \end{aligned} \quad (4.12)$$

onde M é a massa do conjunto (massa do corpo principal mais a massa dos apêndices). É claro que a posição do elemento de massa do elo k depende da posição relativa entre o próprio elo e o corpo principal do satélite. Entretanto, esta posição pode ser obtida em cada instante pois admite-se que haja um dispositivo que atue na junta com uma lei dinâmica conhecida, ou seja, a posição $\theta_k(t)$, a velocidade $\omega_k(t)$, e a aceleração $\dot{\omega}_k(t)$ da junta k em relação ao sistema o são fornecidas. Neste caso, existe uma matriz de transformação de coordenadas, $A_{o,k}(t)$, e sua inversa, $A_{k,o}(t)$, que levam vetores representados no sistema o ao sistema k e vice-versa. Como tanto o sistema de coordenadas o fixo ao corpo quanto o sistema k do elo k são não-inerciais, deve-se utilizar as Equações A.19 e A.26, mostradas no Apêndice A, para obter as acelerações dos elementos de massa. A substituição destes valores na equação da derivada do momento angular leva à uma modificação na equação de Euler, de forma a acomodar o efeito dos apêndices articulados:

$$\dot{L}^o = (I_o + J_n) \dot{\omega}_o^o + \Omega(\omega_o^o) I_o \omega_o^o + H_n \quad (4.13)$$

onde os termos J_n e H_n levam em conta não somente a inércia dos apêndices, mas também o efeito decorrente da mudança do centro de massa do conjunto, devido ao movimento dos elos:

$$\begin{aligned} J_n = & \sum_{k=1}^n \left(A_{k,o} I_k A_{k,o}^T - m_k \Omega(a_{ok}^o - a_{ko}^o) \Omega(a_{ok}^o - a_{ko}^o) \right) + \\ & + \sum_{k=1}^n \left(m_k \Omega(a_{ok}^o - a_{ko}^o) \right) \sum_{k=1}^n \left(\mu_k \Omega(a_{ok}^o - a_{ko}^o) \right) \end{aligned} \quad (4.14)$$

sendo I_k a matriz de inércia do elo k com relação ao sistema k , e:

$$H_n = \sum_{k=1}^n \left[\Omega(\omega_o^o + \omega_k^o) A_{k,o} I_k A_{k,o}^T (\omega_o^o + \omega_k^o) + \right.$$

$$\begin{aligned}
& + A_{k,o} I_k A_{k,o}^T (\dot{\omega}_k^o + \Omega(\omega_o^o) \omega_k^o) \Big] + \sum_{k=1}^n m_k \Omega(a_{ok}^o - a_{ko}^o) \beta_k - \\
& - \left(\sum_{k=1}^n m_k \Omega(a_{ok}^o - a_{ko}^o) \beta_k \right) \sum_{k=1}^n \mu_k \beta_k .
\end{aligned} \tag{4.15}$$

O termo β_k vale:

$$\begin{aligned}
\beta_k = & \Omega(\omega_o^o) \Omega(\omega_o^o) (a_{ok}^o - a_{ko}^o) - \Omega(\omega_o^o) \Omega(\omega_k^o) a_{ko}^o - \\
& - \Omega(\omega_k^o) \Omega(\omega_o^o + \omega_k^o) a_{ko}^o + \Omega(a_{ko}^o) (\Omega(\omega_o^o) \omega_k^o + \dot{\omega}_k^o)
\end{aligned} \tag{4.16}$$

A Equação 4.13 é similar à obtida para corpos rígidos, e como esta última, pode ser invertida e colocada em termos de uma equação diferencial:

$$\dot{\omega}_o^o = (I_o + J_n)^{-1} \left[\sum (N_{cont} + N_{per}) - \Omega(\omega_o^o) I_o \omega_o^o - H_n \right]. \tag{4.17}$$

Caso seja necessário, pode-se incluir nesta última equação os termos devido ao momento angular das rodas de reação, conforme estabelecido nas Relações 4.5 e 4.6.

4.3 EQUAÇÕES CINEMÁTICAS DO MOVIMENTO

Uma vez integrada as equações dinâmicas, (4.17 e 4.6) procede-se a integração das equações cinemáticas. No entanto, a integração do movimento nas variáveis angulares de Euler leva usualmente a singularidades durante o processamento numérico. Por isso, é conveniente efetuar esta integração por meio de quatérnions que, se por um lado elimina o inconveniente das singularidades, por outro dificulta a visualização da atitude (Wertz, 1978):

$$\frac{dq}{dt} = \frac{1}{2} \tilde{\Omega}(\omega_o^o) q \tag{4.18}$$

com

$$\vec{q}^T = [q_1 \quad q_2 \quad q_3 \quad q_4] \quad (4.19)$$

e $\tilde{\Omega}(\cdot)$ representa, agora, a matriz de produto vetorial estendida (e com sinal trocado):

$$\tilde{\Omega}(\omega) = \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix} \quad (4.20)$$

As equações diferenciais dinâmicas e cinemáticas são integradas a partir das condições iniciais conhecidas $\omega_o(t_0)$ e $q(t_0)$, onde t_0 é o instante inicial. Uma vez obtido o vetor de quatérnions, determina-se a matriz de rotação A , que relaciona a posição inicial do satélite com a posição atual:

$$A = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_1q_3 + q_2q_4) & 2(q_2q_3 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \quad (4.21)$$

Fazendo a correspondência entre as componentes da matriz A com

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (4.22)$$

pode-se obter os ângulos de Euler referentes a uma rotação 3-1-3, por exemplo:

$$\phi = \arctan(a_{31} / (-a_{32})), \quad 0 \leq \phi < 2\pi \quad (4.23a)$$

$$\eta = \arccos(a_{33}), 0 \leq \eta < \pi \quad (4.23b)$$

$$\psi = \arctan(a_{13} / a_{23}), 0 \leq \psi < 2\pi. \quad (4.23c)$$

Resta agora obter $q(t_0)$ a partir da atitude inicial do satélite. Seja então a matriz de rotação B que relaciona um sistema de referência inercial com a atitude inicial. Então, as componentes do quatérnio inicial podem ser obtidas, entre outras formas, por:

$$q_4(t_0) = \frac{1}{2} \sqrt{1 + b_{11} + b_{22} + b_{33}} \quad (4.24a)$$

$$q_1(t_0) = \frac{1}{4q_4} (b_{23} - b_{32}) \quad (4.24b)$$

$$q_2(t_0) = \frac{1}{4q_4} (b_{31} - b_{13}) \quad (4.24c)$$

$$q_3(t_0) = \frac{1}{4q_4} (b_{12} - b_{21}) \quad (4.24d)$$

As equações dinâmicas e cinemáticas serão integradas numericamente, utilizando um modelo do Satélite de Sensoriamento Remoto (SSR). A descrição das matrizes de inércia e posição dos elos e juntas para este satélite é feita no Apêndice C. A dinâmica da abertura dos painéis, apresentada na seção seguinte, pode ser integrada analiticamente (com algumas hipóteses) e, embora faça parte do conjunto de variáveis de estado a serem integradas, computacionalmente estão separadas.

4.4 DINÂMICA DE ABERTURA DOS PAINÉIS

A integração das equações dinâmicas do movimento do satélite composto por apêndices articulados necessita do conhecimento da velocidade angular ω_k e aceleração $\dot{\omega}_k$ de cada junta. Estes parâmetros dependem, por sua vez, do tipo de atuador empregado na junta. O movimento nas articulações pode ser causado, por

exemplo, por um motor de torque constante, ou de velocidade angular constante, ou ainda um atuador passivo como molas de torção, entre outros. Normalmente, o histórico do ângulo de abertura, $\theta_k(t)$, só pode ser obtido através da integração (numérica ou analítica) das equações dinâmicas que regem o comportamento da junta. Este é o caso das articulações dos painéis do SSR, cujo sistema de abertura de cada painel é composto por duas dobradiças de baixo atrito, com uma mola de torção pré tensionada em cada uma. Um dispositivo pirotécnico de retenção do painel na posição fechada é acionado de forma a liberar sua abertura logo após a injeção do satélite em sua órbita. Este sistema obedece assim a uma lei dinâmica bastante conhecida, que pode ser integrada analiticamente em certas situações. Como os painéis possuem um comportamento dinâmico, durante a abertura, que depende não somente das condições iniciais mas também das forças e torques aplicados na junta, a integração analítica só é possível se forem admitidas algumas hipóteses, de forma a viabilizar esta integral, sem que haja grande perda de generalidade ou erro excessivo no comportamento dinâmico do sistema. Admite-se, em princípio, que a massa do corpo principal é muito maior que a massa dos painéis, e com isso pode-se desprezar o efeito da reação ao torque da junta no corpo do satélite. Admite-se, em seguida, que as velocidades angulares iniciais dos painéis são muito inferiores às velocidades alcançadas por estes durante a abertura. Desta forma, efeitos de rigidez giroscópica não aparecem nas equações dinâmicas do sistema de abertura, e este desacoplamento permite uma solução analítica. Note-se que estas hipóteses são necessárias apenas para se obter um histórico $\theta_k(t)$ que aproxime a dinâmica verdadeira do torque atuante nas juntas, e não influem no comportamento dinâmico do satélite durante a abertura dos painéis.

O mecanismo de abertura dos painéis possui uma mola de torção que aplica um torque de 0.14 Nm, na posição fechada, e de 0.02 Nm quando o painel estiver aberto. Sabendo-se que durante a abertura o torque é proporcional ao ângulo do painel na articulação, então o torque na junta k pode ser escrito como:

$$T_k = b_k - a_k \theta_k, \quad (4.25)$$

onde $a_k = 0.0764$ Nm/rd e $b_k = 0.14$ Nm. A equação diferencial que rege o comportamento da abertura dos painéis (sistema massa-mola), de acordo com as hipóteses formuladas acima, é dada por:

$$I_k \ddot{\theta}_k + a_k \theta_k - b_k = 0 \quad (4.26)$$

sendo que I_k é o momento de inércia do painel k , com relação ao eixo da articulação. A solução desta equação leva ao resultado:

$$\theta_k(t) = c_k \cos \omega_k t + d_k \sin \omega_k t + \theta_{k0} \quad (4.27)$$

onde c_k , d_k e θ_{k0} são constantes. As condições iniciais de abertura dos painéis permitem a determinação destes coeficientes. Estas condições são: posição θ_k e velocidade angular nulas em $t = 0$, e aceleração angular inicial dada por b_k/I_k . A solução agora pode ser escrita como:

$$\theta_k(t) = \frac{b_k}{a_k} \left(1 - \cos \sqrt{\frac{a_k}{I_k}} t \right) \quad (4.28)$$

O momento de inércia dos painéis com relação ao eixo da junta pode ser obtido a partir da componente I_{yy} da matriz de inércia (que vale 0.290 kg m² para todos os painéis do SSR), e a expressão que permite o cálculo do momento de inércia num eixo trasladado paralelamente, o que resulta $I_k = 1.204$ kg m² ($k = 1, 2, 3$). Substituindo os valores numéricos na solução $\theta_k(t)$, obtém-se:

$$\theta_k(t) = 1.832(1 - \cos 0.252t) \quad (4.29)$$

Quando o ângulo de abertura dos painéis atinge 85° , será suposto que o sistema de batente aplique a ambos: satélite e painéis, um torque constante de forma a cessar o movimento de abertura quando o ângulo atingir 90° . O instante t_1 em que o torque de frenagem é aplicado é obtido invertendo-se a expressão de $\theta_k(t)$, e substituindo-se o valor de $\theta_k = 85^\circ$, o que resulta $t_1 = 5,4739$ s. A aceleração angular necessária para frear o movimento dos painéis de tal forma que a velocidade angular se anule quando $\theta_k = 90^\circ$ pode agora ser obtida, e resulta em $-1,1770$ rd/s. Com isso, a posição $\theta_k(t)$, velocidade $\omega_k(t)$ e aceleração angular $\dot{\omega}_k(t)$ dos painéis em função do tempo ficam:

$$\left. \begin{array}{l} \theta_k(t) = 0 \\ \omega_k(t) = 0 \\ \dot{\omega}_k(t) = 0 \end{array} \right\} \text{para } t \leq t_0 \quad (4.30)$$

$$\left. \begin{array}{l} \theta_k(t) = 1,832 (1 - \cos 0,252t) \\ \omega_k(t) = 0,46166 \text{ sen } 0,252t \\ \dot{\omega}_k(t) = 0,11634 \cos 0,252t \end{array} \right\} \text{para } t \leq t_0 + 5,4739 \text{ s} \quad (4.31)$$

$$\left. \begin{array}{l} \theta_k(t) = 1,4835 + 0,4532 (t - t_1) - \\ \quad - 0,5885 (t - t_1)^2 \\ \omega_k(t) = 0,4532 - 1,1770 (t - t_1) \\ \dot{\omega}_k(t) = -1,1770 \end{array} \right\} \text{para } 5,4739 \text{ s} \leq t - t_0 \leq 5,8590 \text{ s} \quad (4.32)$$

$$\left. \begin{array}{l} \theta_k(t) = 0 \\ \omega_k(t) = 0 \\ \dot{\omega}_k(t) = \pi/2 \end{array} \right\} \text{para } t \leq t_0 + 5,8590 \text{ s} \quad (4.33)$$

onde t_0 é o instante onde se inicia a abertura dos painéis. Os valores de $\theta_k(t)$, $\omega_k(t)$ e $\dot{\omega}_k(t)$ são mostrados nas Figuras 4.4, 4.5 e 4.6, respectivamente, considerando o

instante inicial de abertura $t_0 = 1$ s. Note-se que, embora as funções $\theta_k(t)$ e $\omega_k(t)$ sejam contínuas, a aceleração angular $\dot{\omega}_k(t)$ é descontínua, o que pode trazer problemas durante a integração numérica do movimento do satélite. Este é mais um fator de complexidade a ser sobrepujado pela modelagem das redes neurais.

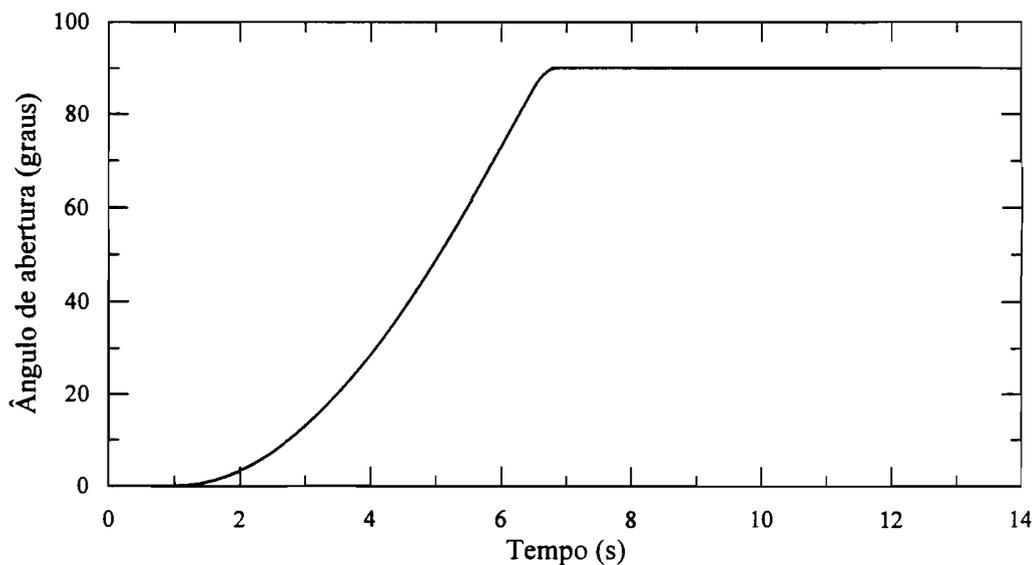


Fig. 4.4 – Ângulo de abertura dos painéis em função do tempo.

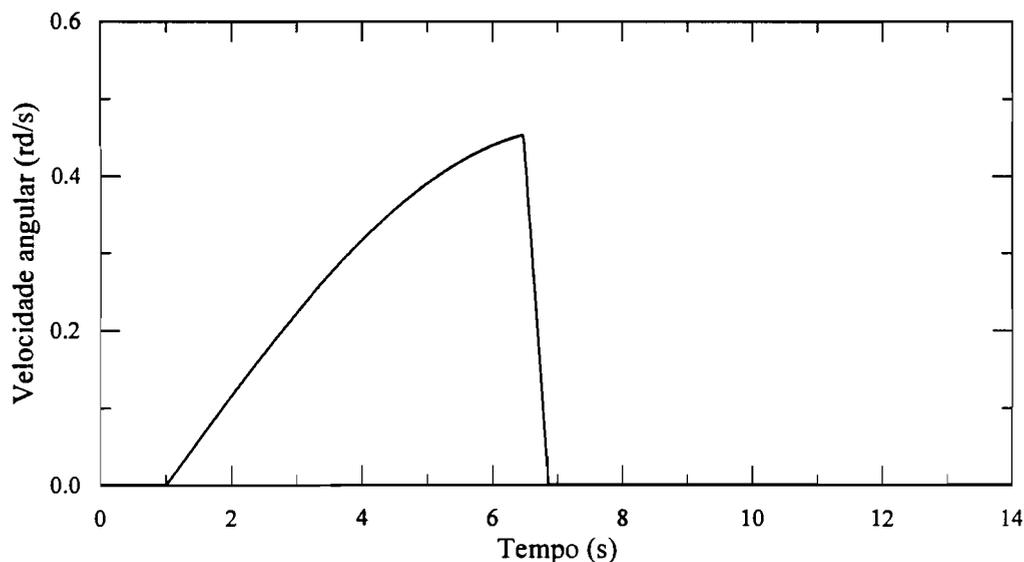


Fig. 4.5 – Velocidade angular dos painéis em função do tempo.

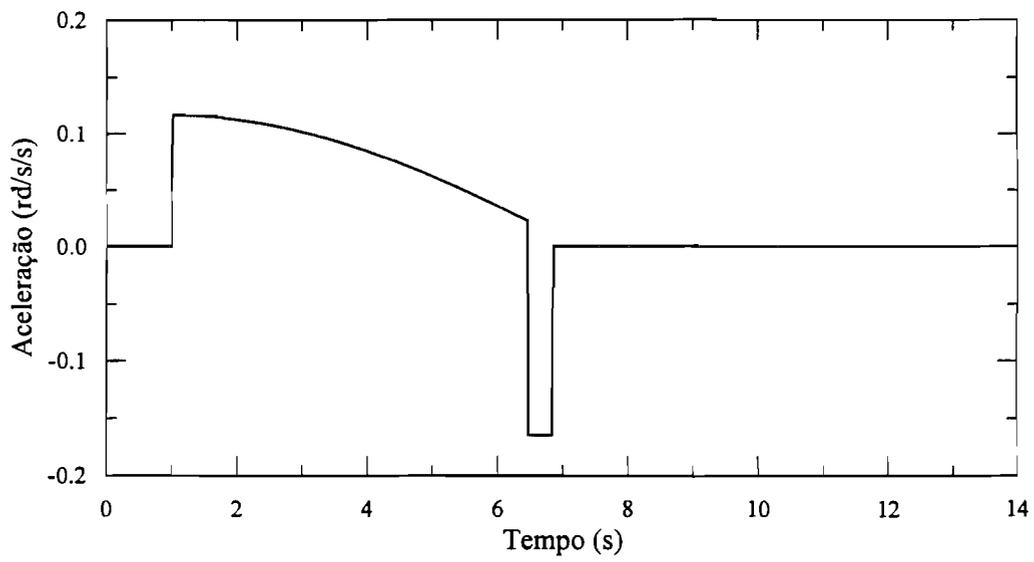


Fig. 4.6 – Aceleração angular dos painéis em função do tempo.

CAPÍTULO 5

SIMULAÇÃO E RESULTADOS

5.1 O SATÉLITE DE SENSOREAMENTO REMOTO (SSR)

O problema de controle de atitude de satélites através da utilização de redes neurais será testado com o satélite de sensoriamento remoto (SSR) da MECB. A geometria deste satélite é dada na Figura 5.1, onde as principais dimensões e sistemas de coordenadas também estão representados. O satélite SSR é formado por um corpo principal, onde se localizam os instrumentos e equipamentos para a operação, e três painéis recobertos por células solares, responsáveis pelo suprimento de energia elétrica para os aparelhos de bordo (ver Apêndice C). O posicionamento dos painéis é concebido visando o melhor aproveitamento da energia disponível na órbita solssíncrona deste satélite. Devido ao tamanho dos painéis e seu posicionamento quando em órbita, estes ficam presos ao corpo do satélite durante o lançamento, e são abertos após a injeção na órbita. Mecanismos de segurança acionados por parafusos pirotécnicos providenciam a liberação dos painéis quando comandados por sinais gerados pelo sistema de controle de atitude. Uma vez liberados, os painéis são impulsionados por molas de torção, e um dispositivo de trava prende-os na posição de operação. Durante o trajeto, a posição e velocidade de abertura são funções exclusivas das características da mola e do atrito fornecido pela junta de articulação.

A abertura dos painéis interfere na atitude porque durante esta operação a geometria do satélite (momentos de inércia e posição do centro de massa) variam com o tempo. Visto que as leis de controle convencionais dependem, para desempenharem sua função, destes parâmetros, fica claro a necessidade de controladores adaptativos ou com grande robustez que superem as variações introduzidas na dinâmica pelo sistema de abertura. Considerando a relativa

capacidade de redes neurais de emularem controladores não-lineares, decidiu-se testar o uso de redes neurais para efetuar o controle de atitude de um satélite com geometria variável no satélite SSR.

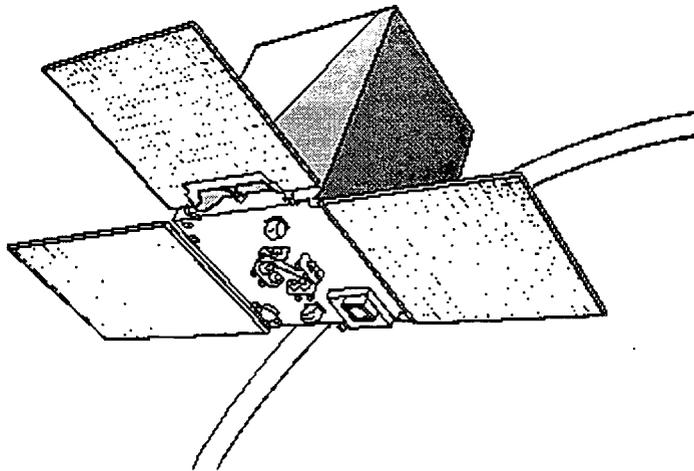


Fig. 5.1 - Satélite de sensoriamento remoto da MECB

Embora o modelo do SSR seja adotado na simulação e testes, usando-se variação de parâmetros para situações críticas, na verdade a implementação deste controlador no satélite sofre diversas dificuldades. A principal delas se refere ao treinamento da rede na fase pós-injeção em órbita. Para ensinar o comportamento dinâmico do satélite à rede, esta deveria excitar o movimento através do acionamento errático dos motores, o que poderia colocá-lo numa situação indesejável, além, é claro, do desperdício de combustível durante o treinamento. Desta forma, todo o aprendizado deverá ser realizado a priori, através de simulação dinâmica. Uma vez treinada, pode-se efetuar um refinamento dos pesos da rede após a colocação em órbita. Este procedimento *on-line* teria a finalidade de ajustar os parâmetros da rede às características reais do satélite. Neste trabalho será considerado, portanto, que o treinamento é realizado antecipadamente ao lançamento através de simulações do comportamento dinâmico do satélite.

Há ainda que se notar que os sistemas de controle de atitude atuais ainda não são voltados para a utilização intensiva de redes neurais, que requerem paralelismo nos cálculos e grande número de sensores para realmente tornarem-se superiores aos controladores convencionais. Há fortes indícios, no entanto, que esta condição será superada pela próxima geração de sistemas de controle, que ganham rapidamente maior poder de cálculo e maior capacidade de memória nos modernos satélites. Pode-se facilmente imaginar outros tipos de satélites com boa chance de utilizar a formulação desenvolvida aqui. Como exemplos, pode-se citar as estações orbitais, onde o acoplamento de naves, agregação de novos módulos ou partes da estrutura e movimentação de massa provocam o efeito de geometria variável. Não tão remotos quanto as estações orbitais, os satélites que requerem um sistema dual de apontamento são também fortes candidatos, como os telescópios espaciais que necessitam manter o conjunto ótico apontado para uma dada direção e o corpo do satélite apontado para Terra, para fins de comunicação. Satélites de retransmissão de dados podem requerer uma parte da estrutura estabilizada e antenas parabólicas móveis para direcionar a retransmissão para diferentes localizações. Não se deve esquecer, ainda, que boa parte dos satélites já em órbita apontam o corpo do satélite para a Terra, enquanto que um sistema rotativo direciona os painéis solares para o Sol. Nestes exemplos a posição do centro de massa e os momentos de inércia variam com o tempo, de acordo com a formulação obtida aqui.

Para empregar rede neural ao controle de atitude do satélite SSR durante a fase de abertura dos painéis solares, será necessário primeiro definir as variáveis de entrada e saída das redes de modelo e controle utilizadas. Considere primeiro a rede neural do modelo do sistema, também chamada de rede de identificação do modelo ou simplesmente rede de identificação. Esta rede tem como entradas o sinal de controle $u(t)$ e a saída do sistema dinâmico $y^p(t)$. Como as variáveis de estado definem como e onde está o sistema em qualquer instante, elas seriam candidatas naturais para exercer a função de variáveis na rede de modelo. No entanto, as redes conseguem extrair a informação sobre o comportamento do sistema, e

conseqüentemente sobre seu estado, utilizando os próprios sinais dos sensores (desde que o sistema seja observável com este conjunto de informações). No caso do SSR, serão empregados dois sensores nesta fase: sensor solar de dois eixos e magnetômetro de três eixos. Será necessário, também, prover à rede as informações atrasadas dos sinais destes sensores, e como as equações dinâmicas linearizadas resultam num sistema de segunda ordem (Wertz, 1978), serão consideradas pelo menos duas entradas atrasadas de cada sinal (solar e magnetômetro).

Também deve ser informada à rede a alteração no comportamento dinâmico do satélite introduzida pela abertura dos painéis. Será suposto que existe um sensor de posição que fornece o ângulo de abertura dos painéis a cada instante. Com a intenção de simplificar o equacionamento, a hipótese de abertura conjunta e simultânea de cada painel deverá ser considerada, o que significa que basta somente um ângulo para fornecer a posição dos três painéis. Será necessário fornecer, também para esta variável, pelo menos um valor atrasado, visto que nas equações do movimento claramente aparece a derivada temporal dos ângulos das articulações. Outra possibilidade seria fornecer à rede o instante inicial de abertura dos painéis, através de um sinal liga-desliga, visto que a própria dinâmica da rede consegue simular o comportamento do satélite durante a abertura.

A rede de identificação deve ter também como entradas os sinais para os atuadores. O SSR deverá utilizar motores de hidrazina como atuadores, na fase de aquisição de atitude, para assegurar a capacidade de geração de torques necessários à estabilização do comportamento dinâmico do satélite. Cada motor consegue fornecer 2 Newtons de empuxo ou, considerando a distância de 95 mm de aplicação do empuxo, 0,19 Nm de torque. Estes motores podem ser acionados continuamente por até 5 minutos, mas em manobras de atitude raramente ultrapassa-se um minuto. Motores de hidrazina sempre apresentam um impulso mínimo, isto é, o torque aplicado no tempo que a válvula de abertura do motor leva para abrir e fechar em seguida. Para o SSR utiliza-se um intervalo de 0,05 s, o que corresponde a um

impulso aproximado de 0,01 Nms. O ciclo de controle, ou seja, o intervalo de tempo entre duas leituras dos sensores e comando dos atuadores será considerado como de 1 segundo. Como durante este intervalo o torque deverá permanecer constante, deve-se prover o sistema de controle com uma eletrônica ou um programa embarcado especial, capaz de transformar o torque médio requerido no ciclo com os limites físicos do atuador. Por exemplo, se o torque médio a ser aplicado no ciclo de 1 segundo for de 0,08 Nm (equivalente a um impulso de 0,08 Nms), então a eletrônica ou programa especial deverá distribuir este impulso numa série de 8 pulsos mínimos de 0,01 Nms igualmente espaçados no ciclo ou então ativar o motor durante 0,4 s. Caso o impulso demandado seja maior que o máximo disponível de 0,19 Nms, então o motor permanece acionado durante todo o ciclo, mas não será capaz de gerar a variação de velocidade angular requerida. O modelo matemático do atuador será portanto um torque limitado em 0,19 Nms, com atuação mínima de 0.01 Nms, como mostrado na Figura 5.2. As informações atrasadas dos sinais dos motores nos três eixos do satélite nos instantes $t-1$ e $t-2$ também fazem parte do conjunto de valores de entrada da rede.

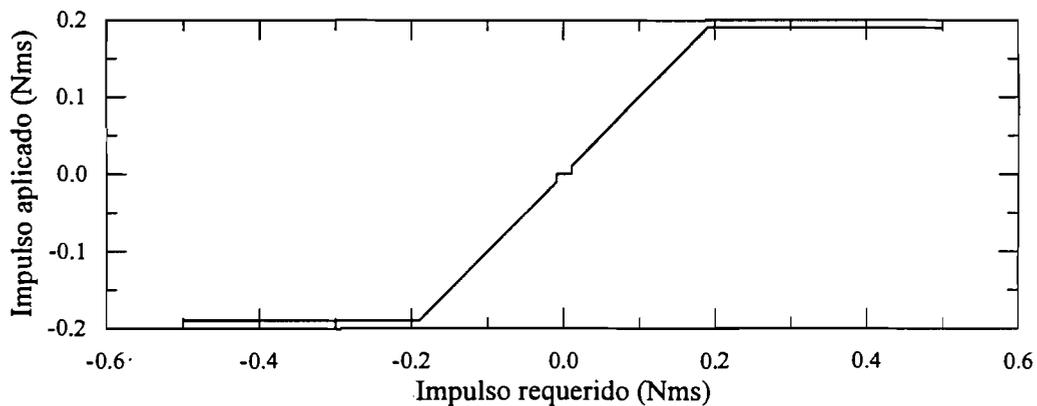


Fig. 5.2 - Modelo do impulso gerado pelo motor de hidrazina.

As saídas da rede de identificação são, naturalmente, estimativas dos próprios sinais dos sensores: magnetômetro de três eixos e sensor solar de dois eixos.

As variáveis da rede de controle dependem da implementação utilizada no seu treinamento. Em geral, elas são uma combinação dos vetores contendo a trajetória de referência $y^d(t)$ do controlador e a saída do sistema $y^p(t)$. O sinal de referência deve ser composto pelas leituras desejadas para os sensores, ou seja, deve fornecer as direções do campo magnético e do Sol em função do tempo. Os seus respectivos valores atrasados nos instantes $t-1$ e $t-2$ devem também ser fornecidos à rede. O sinal dos propulsores nos instantes anteriores podem ou não fazer parte do conjunto de entrada. Embora sejam necessários para compor a dinâmica inversa, conforme as Equações 3.8 e 3.9, algumas implementações, como por exemplo o treinamento pela realimentação do erro dispensam a alimentação deste sinal. A saída da rede controladora deve apresentar apenas o sinal discretizado dos propulsores no instante t .

Em resumo, a rede de identificação possui como entradas 9 valores do campo magnético, 6 valores da posição do Sol, 3 valores do ângulo de abertura dos painéis e 9 valores do sinal dos propulsores, totalizando 27 valores de entrada. Como saídas, esta rede apresenta aproximações dos 3 valores do campo magnético e dos 2 sinais do sensor solar. A rede de controle, por sua vez, apresenta 15 sinais de entrada contendo a soma entre os sinais de referência e a alimentação do erro, e respectivos valores atrasados, e mais 6 valores atrasados do sinal de controle. Na saída apresenta somente os 3 sinais dos propulsores.

A implementação destas redes e seu treinamento certamente irá requerer um processo iterativo de ajuste do número de camadas, número de neurônios em cada camada e funções de ativação adequadas. A princípio, com base em exemplos encontrados na literatura, espera-se que 2 camadas (uma oculta e outra de saída) em cada uma das redes serão suficientes para o grau de aproximação desejado. O número de neurônios, todavia, deve ser ajustado para que o grau de aproximação da rede seja compatível com a precisão necessária para ambos: controle e modelo de identificação.

5.2 O MODELO DINÂMICO DO SATÉLITE SSR

As relações dinâmicas apresentadas no Capítulo 4, que modelam um corpo livre no espaço composto por um ou mais apêndices articulados, foram traduzidas computacionalmente, utilizando-se um integrador Runge-Kutta de ordem 7(8) (Rao e Kuga, 1986; Forsythe et al., 1977; Fehlberg, 1968), com ajuste automático do passo, para integrar as equações do movimento. Todos os processamentos – simulação da dinâmica mais o treinamento das redes - foram realizados em microcomputador IBM-PC, usando linguagem FORTRAN (MicroSoft FORTRAN Power Station 1.0). A posição do satélite na órbita foi obtida a partir de relações clássicas de movimento kepleriano e transformações de coordenadas, com o pacote ORBA (Kuga et al., 1981; Deutsch, 1963; Brower e Clemence, 1961). O vetor campo magnético numa dada posição foi calculado com as rotinas descritas em Lopes et al. (1983), baseados no modelo do IGRF-1990 (International Geomagnetic Reference Field, 1981). A direção do Sol no sistema inercial foi computada com o algoritmo de Flandern e Pulkkinen (1979), contido também no pacote ORBA (Kuga et al., 1981). Devido ao pequeno intervalo de tempo de simulação (da ordem de alguns minutos, no máximo), não se efetuou a propagação de órbita do satélite. Os elementos orbitais (keplerianos) adotados foram: órbita circular com semi-eixo maior de 7128 km (750 km de altitude), inclinação com o equador de 98° e ascensão reta do nodo ascendente, argumento do perigeu e anomalia média nulos. Estes elementos estão referidos na data 1^o de janeiro de 1996, às 0 hs. Uma vez verificado que esta posição não se encontra na sombra da Terra, caso contrário o sensor solar estaria desativado temporariamente, calcula-se a direção do Sol e do campo magnético terrestre, para ser utilizado posteriormente na simulação da leitura dos sensores.

O modelo computacional utilizado para simular a leitura do magnetômetro é bastante simples, e trata-se de uma conversão de escalas. Não foram adicionados ruídos, à princípio, viés (*bias*), ou uma relação de calibração. O campo

magnético calculado no sistema inercial é levado para o sistema de coordenadas da atitude (corpo principal) do satélite, através de multiplicações por matrizes de rotação. Uma vez obtido este vetor no sistema do satélite, aplica-se uma transformação linear de forma a alterar sua escala. O mesmo pode ser dito do sensor solar, que embora na prática forneça uma leitura não linear com o ângulo de medida, cujo modelo foi simplificado para poder ser tratado como entrada na rede neural. Admitiu-se, assim, que este sensor forneça diretamente os co-senos diretores da posição do Sol, referidos ao sistema do satélite, transformados linearmente para compatibilizar as magnitudes de ambos: sensor solar e magnetômetro.

As pequenas perturbações ambientais às quais os satélites estão sujeitos foram omitidas da integração, pois seu efeito acumulado é muito pequeno em virtude do reduzido tempo de simulação (máximo de 10 minutos). As grandezas utilizadas na descrição geométrica e de massa do satélite SSR foram extraídas do Apêndice C.

O comportamento dinâmico do satélite durante a abertura dos painéis é visto na Figura 5.3, com os ângulos de Euler de uma rotação XYZ relativos à atitude inicial dados por φ , η e ψ , respectivamente. O início da abertura dos painéis ocorre após 10 segundos de propagação. O principal efeito da abertura é o movimento introduzido no corpo do satélite, que deriva da energia armazenada na mola do mecanismo de articulação. Ao serem liberados, os painéis introduzem um torque no corpo do satélite, cuja resultante aumenta a velocidade angular preferencialmente no eixo x^o , conforme mostrado na Figura 5.4. Este movimento pode ser visualizado como um efeito perturbador na atitude, a ser compensado pelo sistema de controle do satélite, baseado em propulsores de hidrazina.

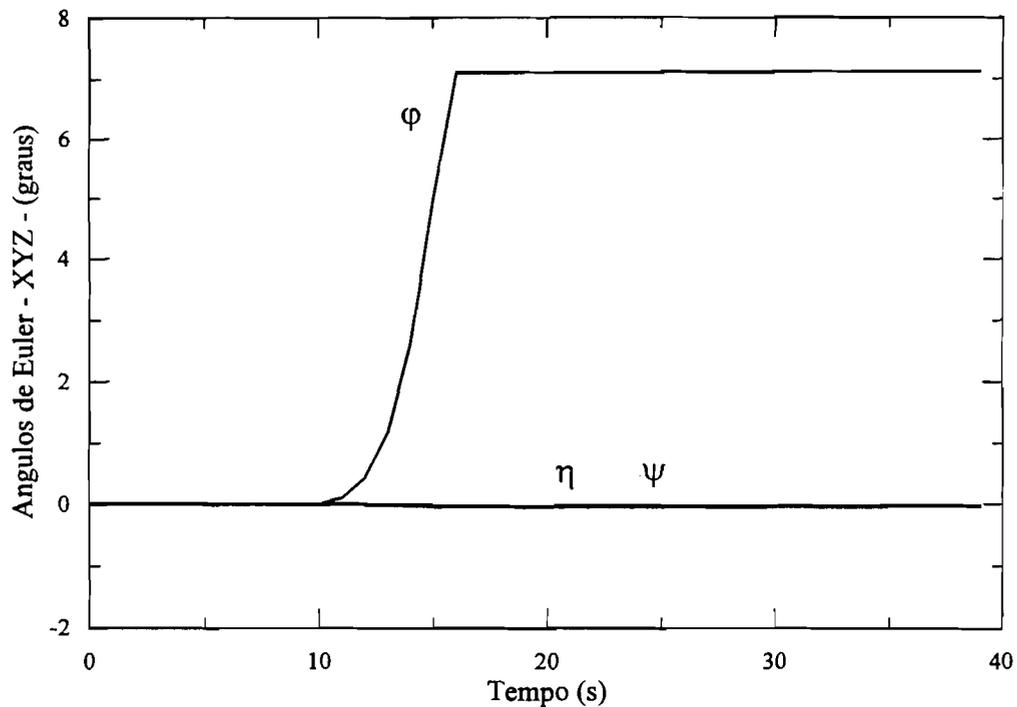


Fig. 5.3 - Dinâmica de abertura dos painéis do satélite SSR.

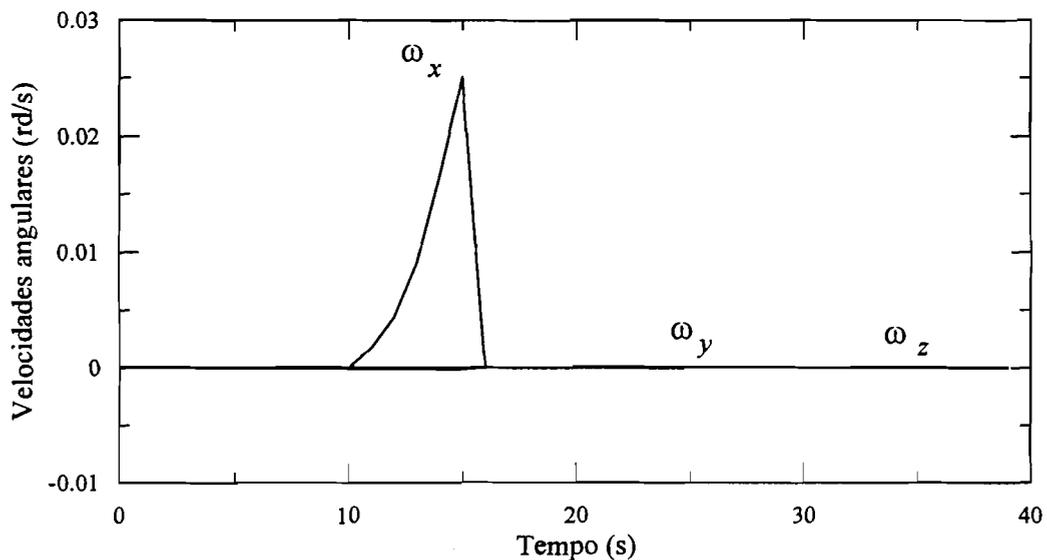


Fig. 5.4 - Velocidades angulares durante a abertura dos painéis do SSR.

O pequeno valor dos momentos de inércia dos painéis, entretanto, provoca um efeito reduzido na atitude – cerca de 7° –, de tal forma que, muito provavelmente, um controlador ótimo baseado na dinâmica linearizada poderia facilmente manter a atitude abaixo de certo erro. Decidiu-se, portanto, investigar uma

situação extrema, na qual supõe-se que o painel superior do satélite (painel 3 da Figura C.1 do Apêndice C) desprenda-se do corpo do satélite ao fim do processo de abertura. Esta situação, embora com probabilidade muito baixa, é ainda assim um caso crítico, pois se a rede conseguir lidar com este problema então mais facilmente conseguirá na situação normal. Caso o painel se desprenda ao fim da abertura, o satélite terá um movimento em atitude dado pela Figura 5.5. Nesta figura, a atitude é descrita em termos de uma rotação ZXZ, com ângulos de Euler dados respectivamente por φ , θ e ψ .

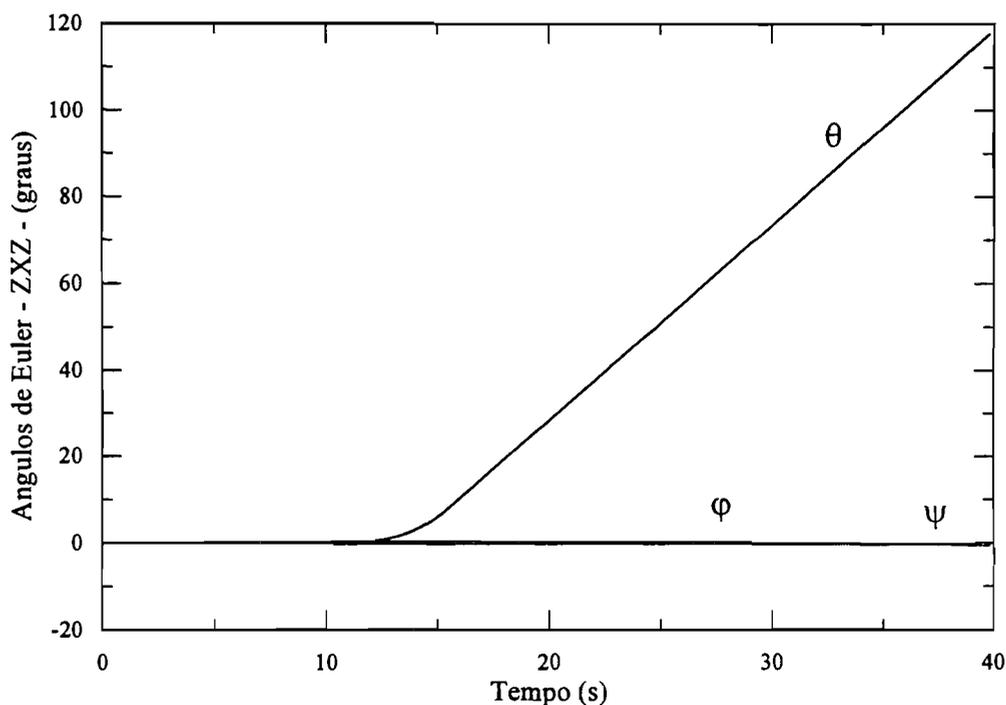


Fig. 5.5 – Atitude do satélite quando o painel se desprende do corpo.

5.3 A IDENTIFICAÇÃO DO MODELO DIRETO

No treinamento do modelo direto, utilizou-se como entradas os sinais do magnetômetro de 3 eixos, os sinais dos sensores solares, o comando de atuação dos propulsores e, finalmente, o ângulo de abertura dos painéis solares. Todos estes sinais foram normalizados, por dois motivos: primeiro, para aproximarem-se dos sinais reais dos sensores, que normalmente apresentam saídas analógicas compreendidas

entre 0 e 5 Volts ou entre -5 e 5 Volts; segundo, porque as componentes do erro apresentado na saída da rede devem ser compatíveis entre si em termos de ordem de magnitude, para que o treinamento não favoreça os sinais de maior amplitude.

Embora empreguem-se em satélites, normalmente, sensores solares de 2 eixos com grande campo de visada - o que permite o recobrimento total com cerca de 2 a 4 sensores - admitiu-se que a direção do Sol fosse fornecida em 3 eixos (co-seno diretores) para simplificar e minimizar a quantidade de sinais processadas pela rede. Isto significa, portanto, que será necessário um prévio processamento do sinal real dos sensores solares antes de dirigi-los à rede. Tal processamento deve incluir a identificação do sensor que está sendo estimulado num dado instante, a transformação de sua saída (normalmente uma função, as vezes não unívoca, da direção do Sol) em valores que especifiquem uma direção em dois eixos e, por último, a transformação desta direção em co-senos diretores. Na simulação, estes sinais foram amplificados para ficarem compreendidos entre -5 e 5, dados por $\sigma_x(t)$, $\sigma_y(t)$ e $\sigma_z(t)$. Por sua vez, as saídas simuladas do magnetômetro foram divididas por 40000 nT (nano Tesla), que é a magnitude máxima do campo magnético terrestre acima de 500 km de altitude, resultando então as medidas $B_x(t)$, $B_y(t)$ e $B_z(t)$ nos eixos x_o , y_o e z_o do satélite.

Como os sinais dos propulsores $\tau_x(t)$, $\tau_y(t)$ e $\tau_z(t)$ não entram na composição do erro na saída da rede de identificação do modelo, permitiu-se mantê-los com magnitude compreendida entre -1 e 1, a despeito do torque de 0,2 Nm utilizado no modelo dos torques gerados pelos propulsores durante a simulação.

A discretização do tempo leva a um problema para o sinal do controlador. Como os sinais são amostrados de 1 em 1 segundo (compatível com controladores embarcados de atitude) o impulso de 0,2 Nms fornecido pelos propulsores neste intervalo é suficientemente grande para provocar um liga-desliga do controle em torno da trajetória de referência. Pior que reduzir a vida útil dos

motores, este procedimento desperdiça o precioso combustível (não renovável) de bordo. A solução para isto será admitir valores intermediários entre -1 e 1 para o sinal do controle. Isto supõe, em adição, uma outra lógica computadorizada ou analógica, capaz de acionar os propulsores em intervalos de tempo menores que 1 segundo. Assim, desprezando o efeito de mínimo impulso que os motores podem fornecer, o modelo do controlador utilizado aqui é essencialmente uma função linear limitada: para valores unitários, os motores permanecem ativos por todo o intervalo de discretização; para valores compreendidos entre -1 e 1, os propulsores são acionados durante um intervalo de tempo proporcional ao valor do sinal de controle.

Foi admitido, também, que o ângulo de abertura dos painéis, $\theta(t)$, fosse fornecido através da medição da tensão aplicada numa resistência variável fixada na articulação de um dos painéis. Após passar por um conversor analógico/digital, este sinal é convertido num valor compreendido entre 0 e 5.

Estes 10 sinais mencionados, juntamente com mais 2 valores obtidos em instantes anteriores, e mais o sinal unitário para o patamar de ativação, totalizando 31 valores, como mostrado na Tabela 5.1, são fornecidos à rede de identificação do modelo, e sua saída é, a seguir, comparada com os sinais dos sensores amostrados no instante seguinte, que constituem a saída da rede, indicada na Tabela 5.2.

Para que o treinamento pudesse ser representativo, e, desta forma, a identificação do modelo direto fosse completa, admitiu-se que a atitude inicial fosse qualquer, escolhida aleatoriamente no início de cada simulação. A velocidade angular inicial é também escolhida aleatoriamente, porém limitada a 5 rotações por minuto, no máximo, em cada um dos eixos do satélite (compatível com a dispersão máxima fornecida por veículos lançadores).

Como não se dispõe de uma lei de controle para efetuar o acionamento dos propulsores, durante a identificação do modelo, admitiu-se aqui que os motores

fossem acionados aleatoriamente, com uma probabilidade de 8% de estarem acionados em cada instante - tanto em um sentido quanto em outro - e com duração dada por uma distribuição de Poisson, com intervalo médio de 1 segundo. Não há, na verdade, motivos outros que não intuitivos para justificar tais valores. No entanto, a atuação prolongada ou freqüente dos motores podem levar o comportamento dinâmico do satélite a uma região afastada daquela de interesse do ponto de vista do controle, enquanto que atuações esporádicas ou muito rápidas dos motores poderão levar a um aprendizado incompleto da rede neural.

TABELA 5.1 - SINAIS DE ENTRADA DA REDE DE IDENTIFICAÇÃO

Sinal	Variável	Instantes	Valor simulado	Valor numérico
Sensor Solar	σ_x	$t, t-1, t-2$	-1 a 1	-5 a 5
	σ_y	$t, t-1, t-2$	-1 a 1	-5 a 5
	σ_z	$t, t-1, t-2$	-1 a 1	-5 a 5
Magnetômetro	B_x	$t, t-1, t-2$	-400 a 400 mG	-5 a 5
	B_y	$t, t-1, t-2$	-400 a 400 mG	-5 a 5
	B_z	$t, t-1, t-2$	-400 a 400 mG	-5 a 5
Propulsores	τ_x	$t, t-1, t-2$	-0,2 a 0,2 Nm	-1 a 1
	τ_y	$t, t-1, t-2$	-0,2 a 0,2 Nm	-1 a 1
	τ_z	$t, t-1, t-2$	-0,2 a 0,2 Nm	-1 a 1
Ângulo de abertura	θ	$t, t-1, t-2$	0° a 90°	-5 a 5
Patamar	b	t	-	1

TABELA 5.2 - SINAIS DE SAÍDA DA REDE DE IDENTIFICAÇÃO

Sinal	Variável	Instantes	Valor numérico
Sensor Solar	σ_x	t+1	-5 a 5
	σ_y	t+1	-5 a 5
	σ_z	t+1	-5 a 5
Magnetômetro	B_x	t+1	-5 a 5
	B_y	t+1	-5 a 5
	B_z	t+1	-5 a 5

A eficácia do método de treinamento através de mínimos quadrados recursivos foi amplamente demonstrada, conforme o exemplo mostrado no Apêndice

D. Porisso, embora tenham sido tentadas outras possibilidades, adotou-se um treinamento composto de 200 pontos (ou 200 segundos), em cada simulação. A simulação da abertura dos painéis foi realizada sempre no meio do intervalo de simulação, ou seja, após 100 segundos de treinamento, assegurando, com isso, que a rede fosse treinada tanto na geometria do satélite com painéis fechados quanto com os painéis abertos. O exemplo citado indicou, também, que cerca de 8 neurônios na camada oculta seriam suficientes para simular o comportamento dinâmico de cada sinal amostrado dos sensores. Como são 6 os sinais de saída da rede, isto poderia levar à conclusão que seriam necessários cerca de 50 neurônios ocultos na simulação da atitude. Uma rede composta por 50 neurônios ocultos e 6 neurônios na saída tornaria o procedimento de treinamento através do método de mínimos quadrados quase impraticável, em virtude do tempo necessário para treinar a rede em um único ponto, além da grande quantidade de memória necessária para armazenar todas as matrizes. Somente uma das matrizes de covariância teria dimensão de 1856 por 1856, o que, em precisão dupla, levaria à necessidade de 27 Mbytes. Felizmente, e graças ao grande poder de generalização apresentado pelas rede neurais, não foi necessário chegar a tais extremos. De fato, os primeiros testes, considerando apenas 8 neurônios na camada oculta, foram suficientes para garantir a precisão requerida a ponto de ser desnecessário aumentar este valor. É claro que tampouco atingiu-se, com esta configuração, precisões de 10^{-17} como aquelas obtidas no Apêndice D. Bastou tão somente atingir, com esta rede, precisões compatíveis com aquelas oriundas do magnetômetro e do sensor solar (que conseguem, no máximo, precisões de 1%). Decidiu-se, em função destes aspectos, levar adiante o treinamento da rede de identificação até que esta atingisse um erro inferior a $5 \cdot 10^{-2}$ (1% do valor máximo), ou um erro quadrático médio menor do que $2,5 \cdot 10^{-3}$. Como será visto, este patamar foi atingido após cerca de 28 conjuntos de simulações, cada uma delas com condições iniciais diferentes entre si. Após o treinamento, foram realizados testes da rede utilizando condições iniciais nas quais ela não havia sido treinada. Verificou-se assim que o erro permaneceu próximo aos valores obtidos durante o treinamento.

Outros parâmetros utilizados no treinamento foram (ver Apêndice D): algoritmo de treinamento dos mínimos quadrados com fator de esquecimento λ de 0,996; magnitude inicial da matriz de covariância de 10, amplitude inicial dos pesos de 0,1 e semente aleatória nula. O processo de abertura dos painéis foi simulado com a condição crítica de desprendimento no final. A Figura 5.6 mostra a evolução do erro quadrático médio em função do número de treinamentos. Cada conjunto de treinamento é composto por uma simulação contendo 200 pontos (um a cada segundo de propagação). Com cerca de 40 simulações, atingiu-se a precisão desejada para a identificação do modelo direto. As figuras que seguem mostram uma comparação entre a saída apresentada pela rede neural e o resultado da propagação da dinâmica do satélite, com condições iniciais diferentes daquelas empregadas na fase de treinamento. Nas Figuras 5.7, 5.9 e 5.11 são mostrados os 3 sinais do sensor solar, superpostos à saída correspondente da rede. As diferenças entre ambos os sinais, bastante pequenas, são mostradas nas Figuras 5.8, 5.10 e 5.12. Percebe-se que embora em alguns pontos o erro supere o limite estabelecido, na média a diferença permanece confinada dentro da precisão desejada, mesmo neste caso onde a rede não foi treinada. As Figuras 5.13, 5.15 e 5.17 representam as saídas da saída relativas ao sinal do magnetômetro nos eixos x^o , y^o e z^o do satélite, respectivamente. As Figuras 5.14, 5.16 e 5.18, mostram os erros cometidos pela rede com relação a estes sinais, nos três eixos. Também aqui o erro ultrapassa o valor de 0.05 apenas esporadicamente. Nas Figuras 5.19, 5.20 e 5.21 são mostrados os sinais de entrada dos propulsores, gerados pelo simulador de torques aleatórios.

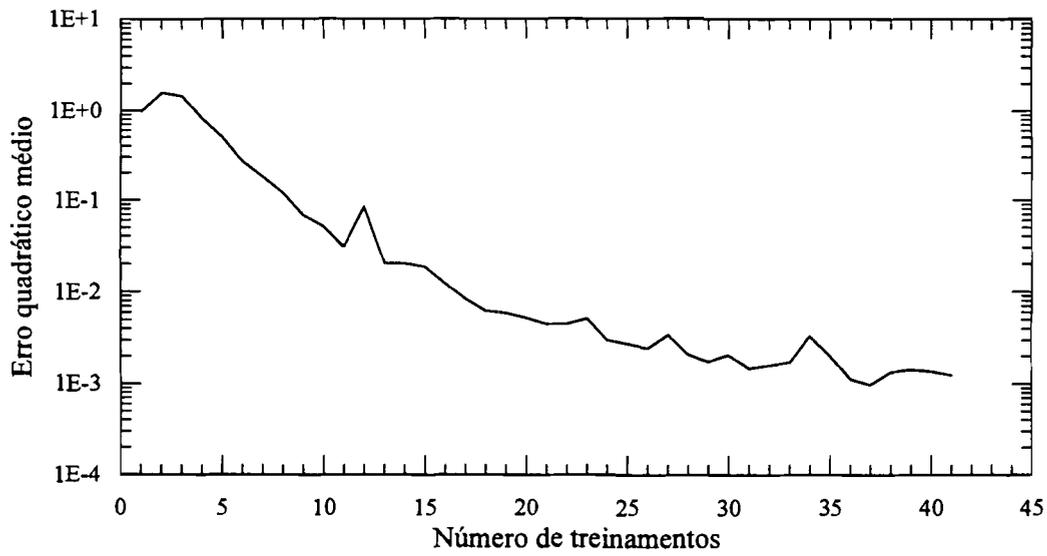


Fig. 5.6 - Treinamento da rede de identificação da atitude

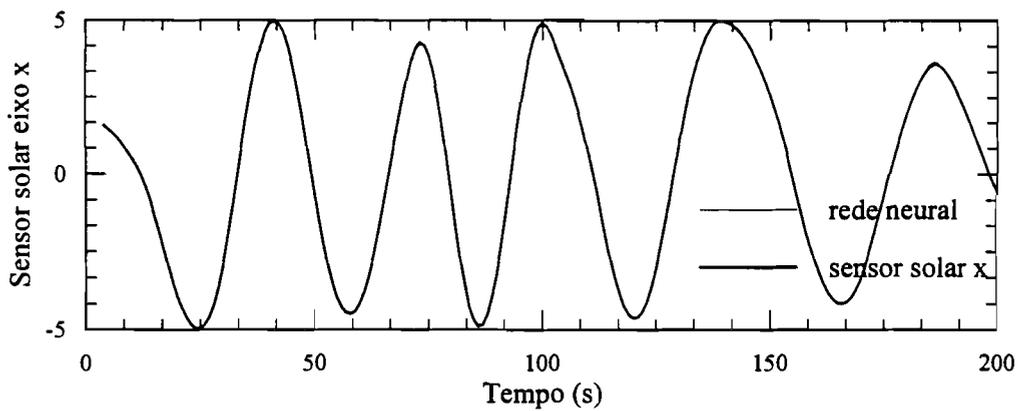


Fig. 5.7 - Teste da RN com o modelo direto - direção do Sol no eixo x^o .

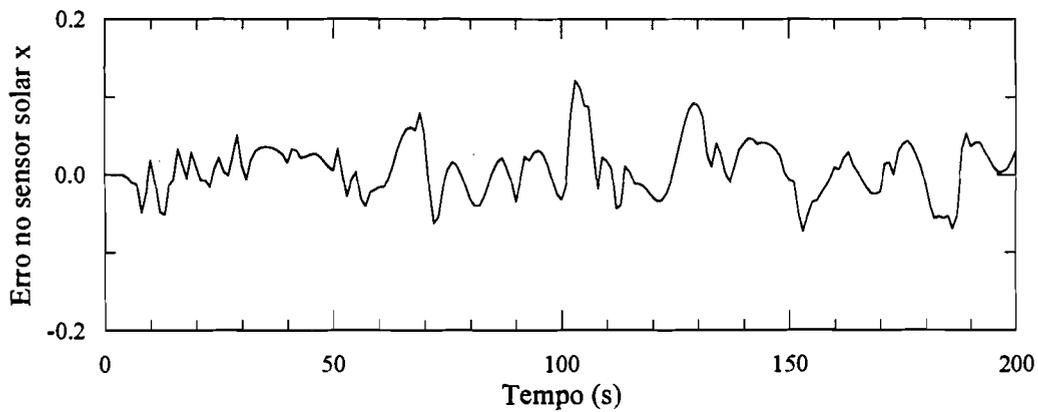


Fig. 5.8 - Erro apresentado pela rede neural na leitura do sensor solar σ_x .

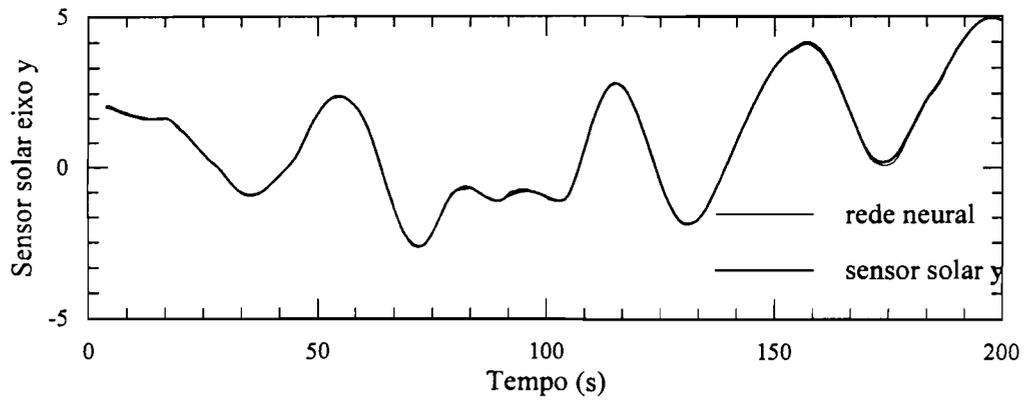


Fig. 5.9 - Teste da RN com o modelo direto - direção do Sol no eixo y^o .

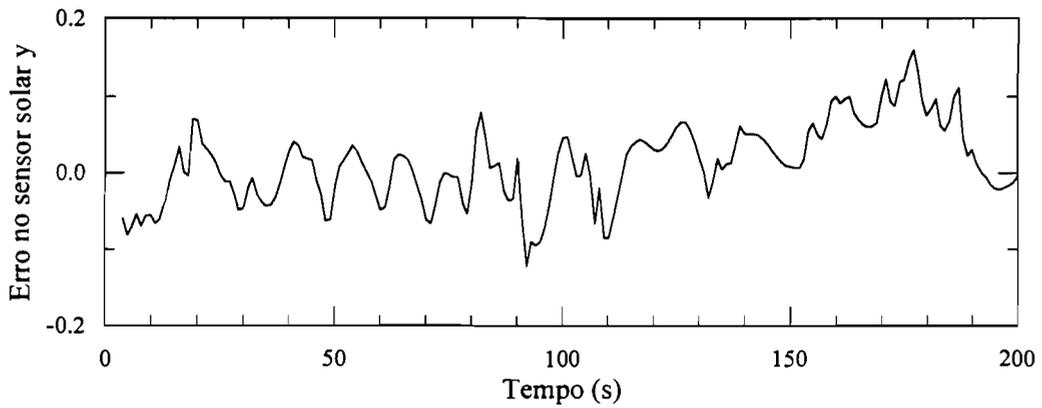


Fig. 5.10 - Erro apresentado pela rede neural na leitura do sensor solar σ_y .

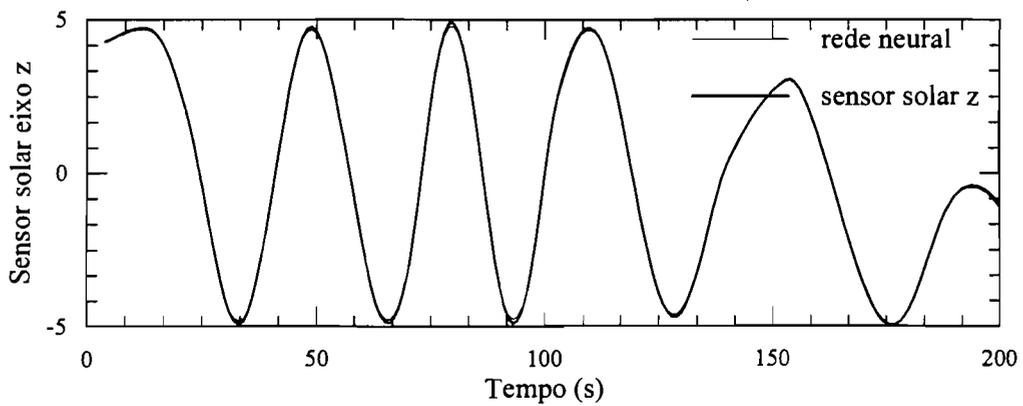


Fig. 5.12 - Teste da RN com o modelo direto - direção do Sol no eixo z^o .

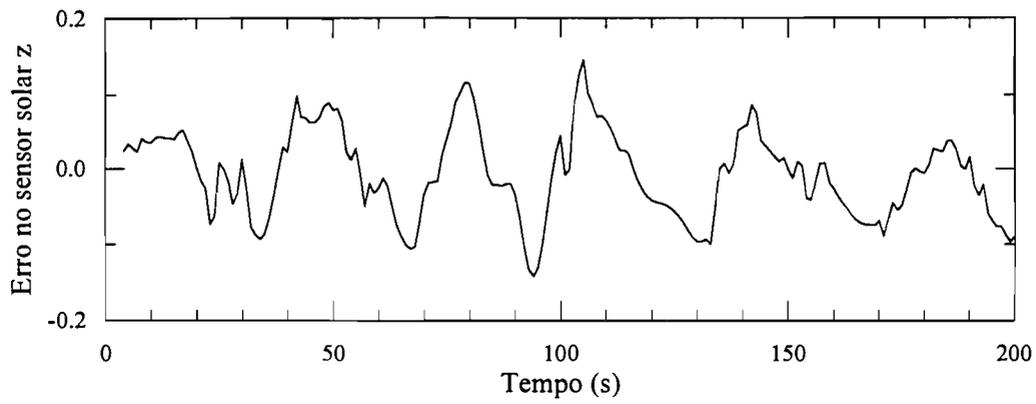


Fig. 5.13 - Erro apresentado pela rede neural na leitura do sensor solar σ_z .

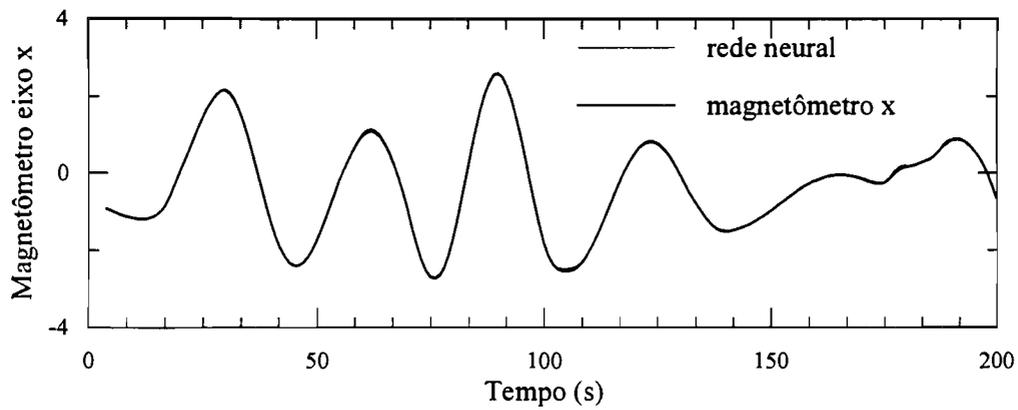


Fig. 5.15 - Teste da RN com o modelo direto - magnetômetro no eixo x° .

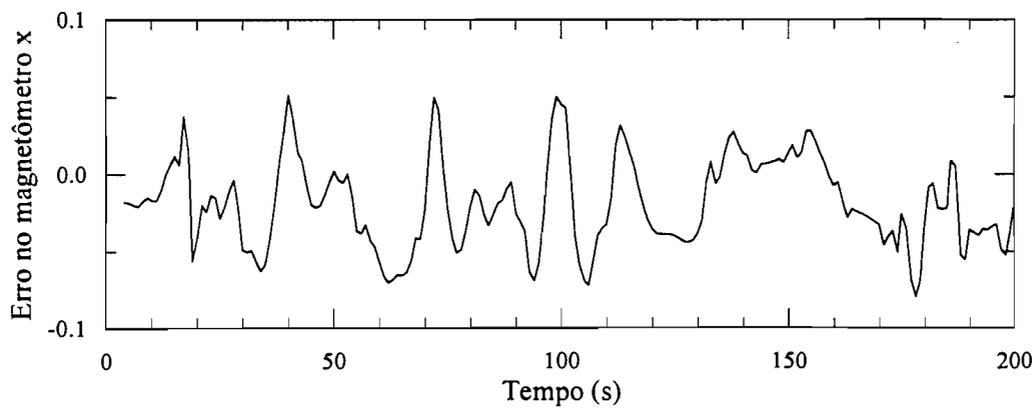


Fig. 5.14 - Erro apresentado pela rede neural na leitura do magnetômetro B_x .

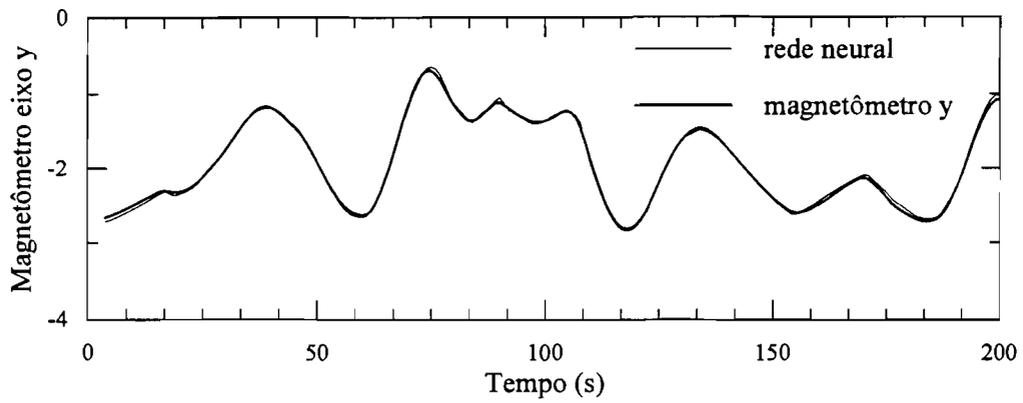


Fig. 5.18 - Teste da RN com o modelo direto - magnetômetro no eixo y^o .

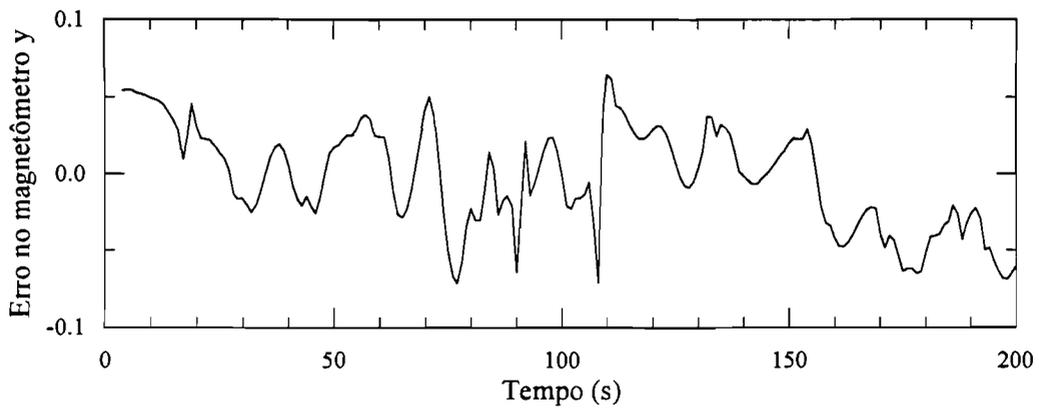


Fig. 5.17 - Erro apresentado pela rede neural na leitura do magnetômetro B_y .

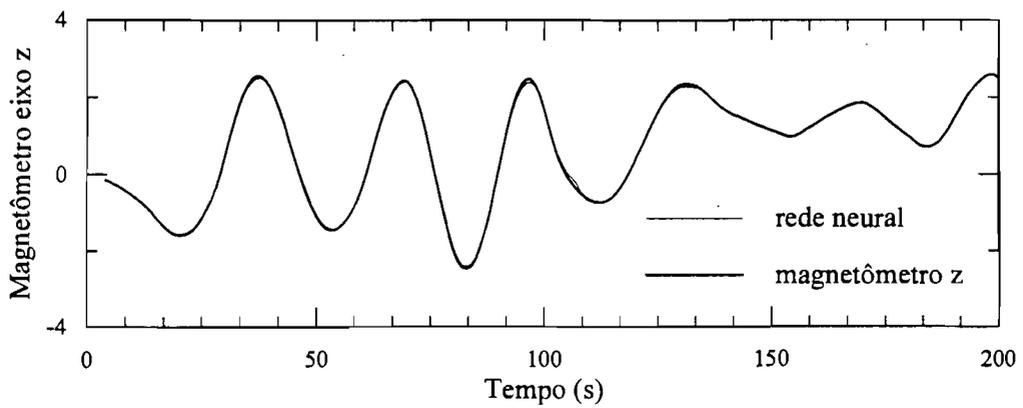


Fig. 5.21 - Teste da RN com o modelo direto - magnetômetro no eixo z^o .

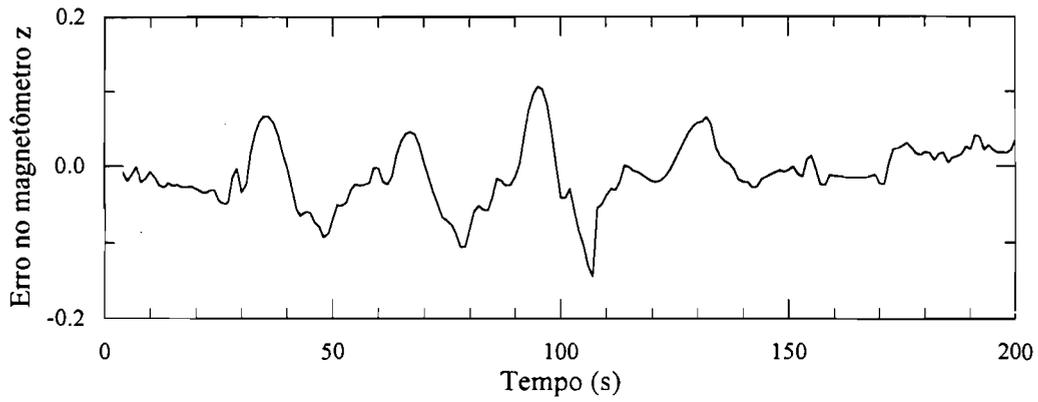


Fig. 5.20 - Erro apresentado pela rede neural na leitura do magnetômetro B_z .

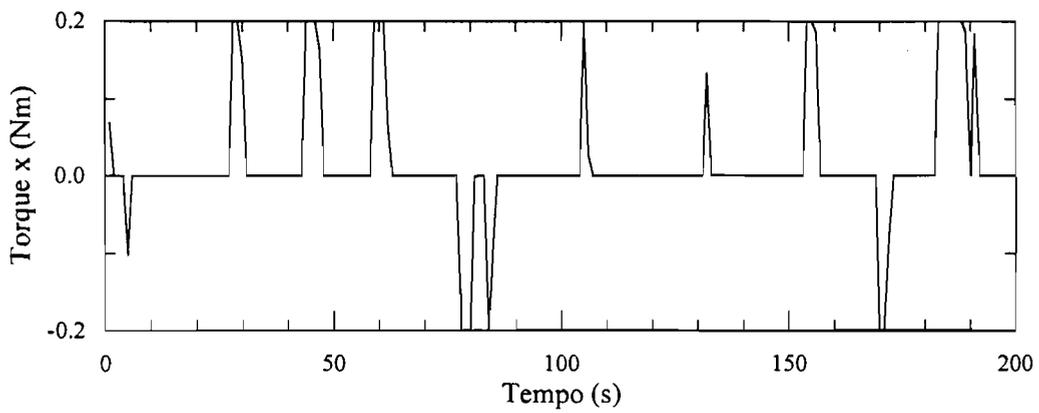


Fig. 5.22 - Perfil de torque aplicado no eixo x^o no treinamento do modelo direto.

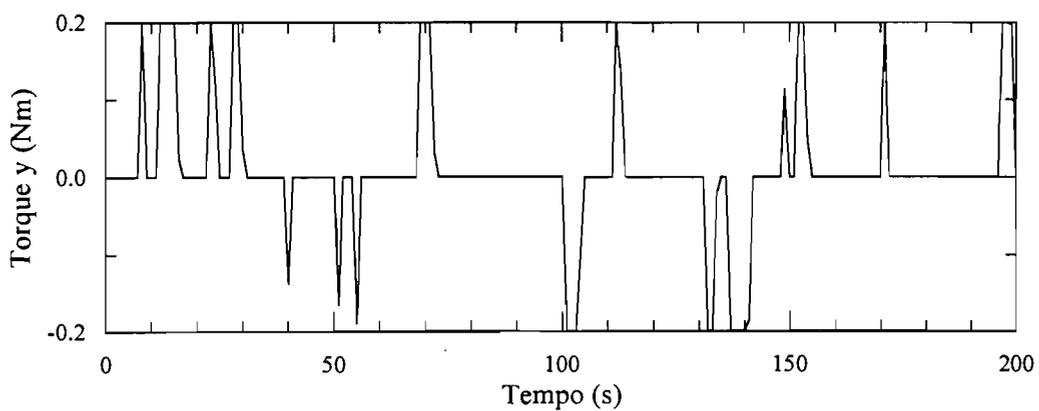


Fig. 5.24 - Perfil de torque aplicado no eixo y^o no treinamento do modelo direto.

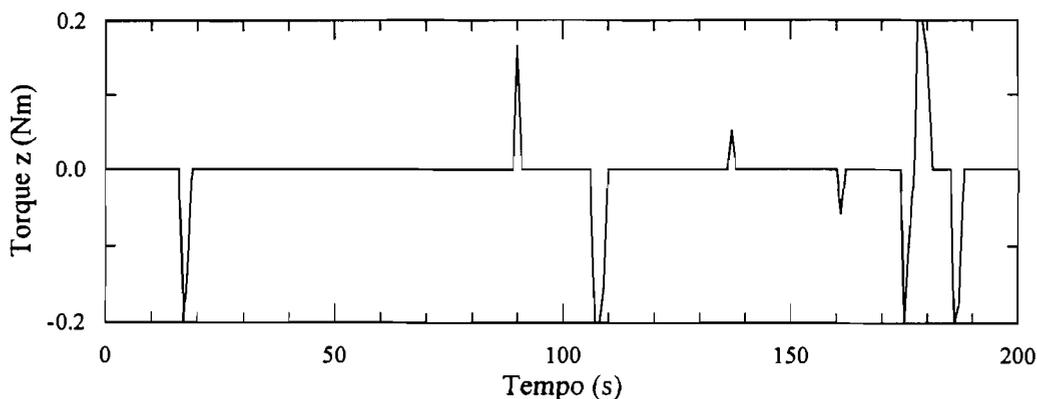


Fig. 5.26 - Perfil de torque aplicado no eixo z° no treinamento do modelo direto.

5.4 O TREINAMENTO DA REDE NEURAL DE CONTROLE

Foram realizadas diversas tentativas para obter o modelo inverso do comportamento dinâmico. Na primeira delas, utilizou-se o treinamento especializado inverso, conforme mostrado no Capítulo 3, Seção 3.3. A principal razão desta escolha, além das naturais vantagens apontadas na seção citada, deveu-se ao fato de não se dispor de uma lei de controle previamente estabelecida. Não que não fosse possível obter esta lei de controle através de métodos convencionais de controle moderno, mas sim que um dos motivos deste trabalho consiste em justamente obter um controlador não linear para um sistema no qual, supõe-se, seja desconhecida tal lei.

As entradas utilizadas na identificação do modelo inverso foram as mesmas daquelas utilizadas no modelo direto, exceto pela retirada, no conjunto de valores, dos sinais dos 3 propulsores (controlador) no instante t (o qual não se conhece, e que corresponde justamente à saída da rede inversa) e da inclusão neste mesmo conjunto, do valor de referência dos sensores no instante $t+1$, conforme demonstra a Equação 3.9 do modelo inverso. Este valor de referência deve ser obtido em função do objetivo a ser atingido pelo sistema de controle. No caso mais geral, os valores de referência são funções do tempo, e definem a trajetória que o satélite é

obrigado a seguir. Casos particulares ocorrem quando esta trajetória leva a um estado estacionário, que tanto pode ser qualquer como um valor previamente especificado. No primeiro caso, o satélite deve manobrar para a posição especificada, e permanecer fixo nesta posição. No segundo caso, não há uma atitude específica para o satélite, bastando tão somente que o controle mantenha um apontamento fixo. Como exemplo, pode-se desejar que o controle anule as velocidades angulares do satélite, e as mantenha assim. No caso do satélite utilizado na simulação, a fase de abertura dos painéis ocorre num instante no qual ainda não se conhece a atitude do satélite, e, portanto, a geração de uma trajetória de referência deverá levar em conta este desconhecimento. Alguns cuidados, entretanto, devem ser tomados quando da geração da trajetória.

O primeiro deles consiste em garantir que os pontos da trajetória e a atitude a ser atingida realmente correspondam a valores reais. A situação contrária poderia ocorrer, por exemplo, se, naquele ponto do espaço e naquele instante, as direções (e magnitudes) dos vetores posição do Sol e do campo magnético não corresponderem àquelas fornecidas como referência. Neste caso, o controlador irá agir de forma a minimizar o erro de atitude, mas haverá um valor mínimo diferente de zero (que depende da referência), que o controle não mais conseguirá reduzir.

O segundo ponto necessita de uma análise mais cuidadosa. Trata-se do problema de até que ponto o valor da referência pode se afastar do valor real do sistema no instante $t+1$. Valores muito diferentes entre eles significa que o controle terá que agir de forma mais rápida, o que certamente irá provocar a saturação do sistema de atuação. Uma outra consequência desta diferença é que a rede inversa poderá não responder adequadamente, visto que a rede direta não foi treinada em tal condição. Em outras palavras, há um compromisso entre a definição da trajetória de referência e o estado atual do sistema. Para que as diferenças entre o estado e a referência sejam pequenas, e com isso evitar a saturação do sistema de controle, deve-se passar gradualmente da posição atual para a trajetória de referência. Seja

então $y^p(t)$ o valor do estado no instante t e $y^m(t)$ o valor que o sistema deveria possuir no mesmo instante. Sugere-se, aqui, uma mudança gradual de um valor para o outro, na forma:

$$y^d(t+1) = g(t) y^m(t) + (1 - g(t)) y^p(t) \quad (5.1)$$

onde $y^d(t+1)$ é o valor de referência a ser fornecido a rede neural. $g(t)$, é uma função com as seguintes características:

$$g(t) = \begin{cases} 0, & \text{para } t = 0 \\ 1, & \text{para } t \rightarrow \infty \end{cases} \quad (5.2)$$

$$\frac{dg}{dt} > 0, \text{ para } t > 0 \quad (5.3)$$

Desta forma, $y^d(t+1)$ passa gradualmente de $y^p(t)$ para $y^m(t)$. Através do ajuste da função $g(t)$, pode-se adequar o treinamento para evitar que o controlador sature. Na grande maioria das aplicações, inclusive aquela aqui apresentada, o valor desejado final da trajetória consiste de um único ponto, em torno do qual o sistema deve ser mantido pelo controlador. No caso do satélite de sensoriamento remoto, pode-se tanto anular a velocidade angular e manter o satélite nesta posição, como também comandá-lo para uma posição previamente definida. No primeiro caso, $g(t) = 1$ e, no segundo caso, $g(t)$ segue as condições expostas acima, e $y^m(t)$ é um valor calculado previamente (poderia ser até mesmo $y^p(0)$). Uma escolha lógica para a função $g(t)$ seria a sigmóide:

$$g(t) = \frac{e^{\alpha t} - 1}{e^{\alpha t} + 1}, \quad t > 0, \quad (5.4)$$

embora pareça provável que trechos de funções lineares também possam ser utilizados, desde que obedeçam as condições estabelecidas para $g(t)$.

O modelo inverso tem, portanto, os 3 sinais do sensor solar, os 3 sinais do magnetômetro, e o ângulo de abertura dos painéis. Para estes sinais, 2 valores atrasados são também incluídos. Além destes, o vetor de entradas conta ainda com os sinais defasados no tempo dos propulsores, e também os sinais da trajetória de referência no instante $t+1$ tanto para o sensor solar quanto para o magnetômetro, totalizando 34 sinais (incluindo a entrada referente ao patamar de ativação). A Tabela 5.3 mostra a composição do vetor de entradas da rede inversa (do controlador).

TABELA 5.3 - SINAIS DE ENTRADA DA REDE DO MODELO INVERSO

Sinal	Variável	Instantes	Valor numérico
Sensor Solar	σ_x	$t+1, t, t-1, t-2$	-5 a 5
	σ_y	$t+1, t, t-1, t-2$	-5 a 5
	σ_z	$t+1, t, t-1, t-2$	-5 a 5
Magnetômetro	B_x	$t+1, t, t-1, t-2$	-5 a 5
	B_y	$t+1, t, t-1, t-2$	-5 a 5
	B_z	$t+1, t, t-1, t-2$	-5 a 5
Propulsores	τ_x	$t-1, t-2$	-1 a 1
	τ_y	$t-1, t-2$	-1 a 1
	τ_z	$t-1, t-2$	-1 a 1
Ângulo de abertura	θ	$t, t-1, t-2$	0° a 90°
Patamar	b	t	1

O vetor de saídas da rede é composto tão somente pelo sinal dos 3 propulsores no instante t , conforme indicado na Tabela 5.4

TABELA 5.4 - SINAIS DE SAÍDA DA REDE DO MODELO INVERSO

Sinal	Variável	Instantes	Valor numérico
Propulsores	τ_x	t	-1 a 1
	τ_y	t	-1 a 1
	τ_z	t	-1 a 1

Infelizmente, o treinamento com o modelo especializado inverso não adquiriu a informação adequada para o controle da rede, fazendo que os motores pulsassem de forma errática e levando o sistema, invariavelmente, a adquirir velocidades angulares nas quais a rede direta não havia sido treinada. O verdadeiro motivo pelo qual isto ocorreu é ainda nebuloso. Acredita-se que o baixo torque gerado pelos propulsores é incapaz de impregnar a rede do modelo direto com informações sobre sua atuação. De fato, sendo a aceleração angular máxima produzida pelos propulsores da ordem de $0,01 \text{ rd/s}^2$, resulta que o deslocamento angular durante o intervalo de 1 segundo utilizado no treinamento não ultrapassa $0,005 \text{ rd}$. Supondo uma relação linear (para pequenos ângulos) entre o deslocamento angular e a medida efetuada pelos sensores, tal distância angular corresponde a uma alteração máxima de $0,025$ na medida do sensor (com amplitude igual a 5). Para que o efeito deste deslocamento seja sentido pela rede, ela deve ser treinada até que o erro seja pelo menos 5 vezes menor que do que a alteração provocada pelo propulsor, ou seja $0,005$, o que corresponde a um erro quadrático total de $7,5 \cdot 10^{-5}$. Utilizando-se de uma rede com 12 neurônios na camada oculta para o treinamento do modelo direto, o menor erro que se obteve foi de $1 \cdot 10^{-3}$, depois de 200 iterações, conforme a Figura 5.22, erro este ainda muito grande comparado ao valor necessário para detectar o efeito dos motores. Neste treinamento, para que o efeito dos propulsores pudesse ser maximizado, estabeleceu-se em 50% a probabilidade dos motores serem acionados num dado instante. Para detectar este efeito, decidiu-se comparar os erros apresentados pela rede de modelo direto com os erros obtidos pela mesma rede onde o sinal de entrada referente aos propulsores foi subtraído (na verdade, foi anulado). Como pode ser visto nas Figuras 5.23 e 5.24, ambos os erros são bastante similares e possuem a mesma ordem de grandeza. Isto confirma que a rede ainda não conseguiu assimilar a influência dos propulsores na dinâmica do satélite. Nos gráficos mostrados foi simulada a abertura com desprendimento do painel superior, e o treinamento utilizou o algoritmo dos mínimos quadrados.

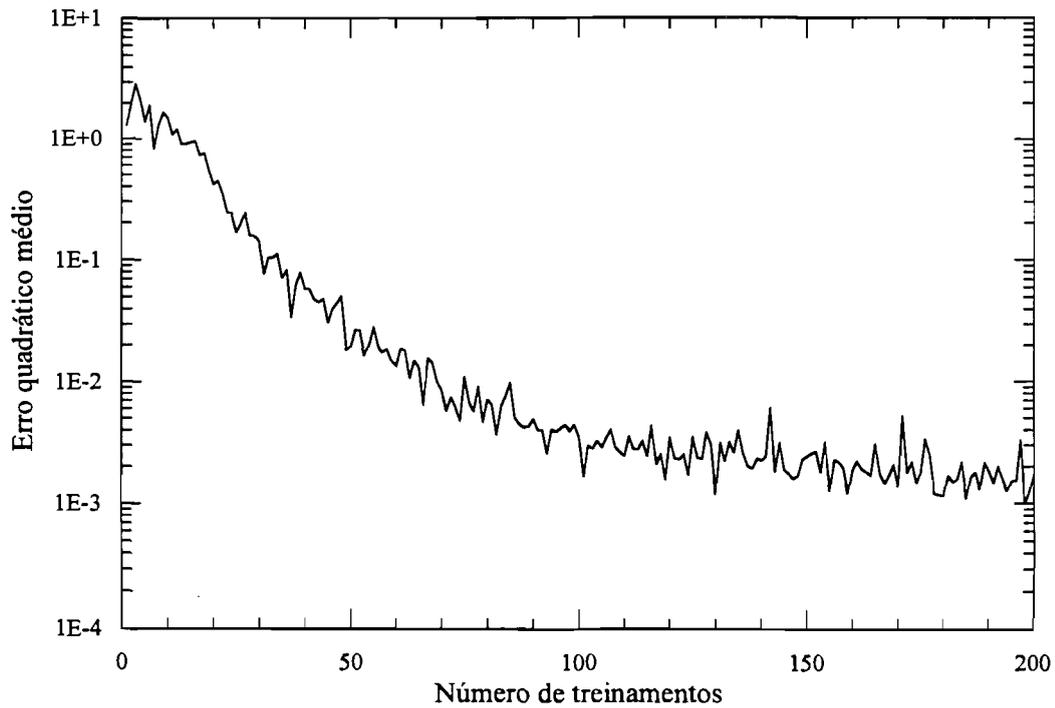


Fig. 5.22 - Treinamento do modelo direto com 12 neurônios na camada oculta.

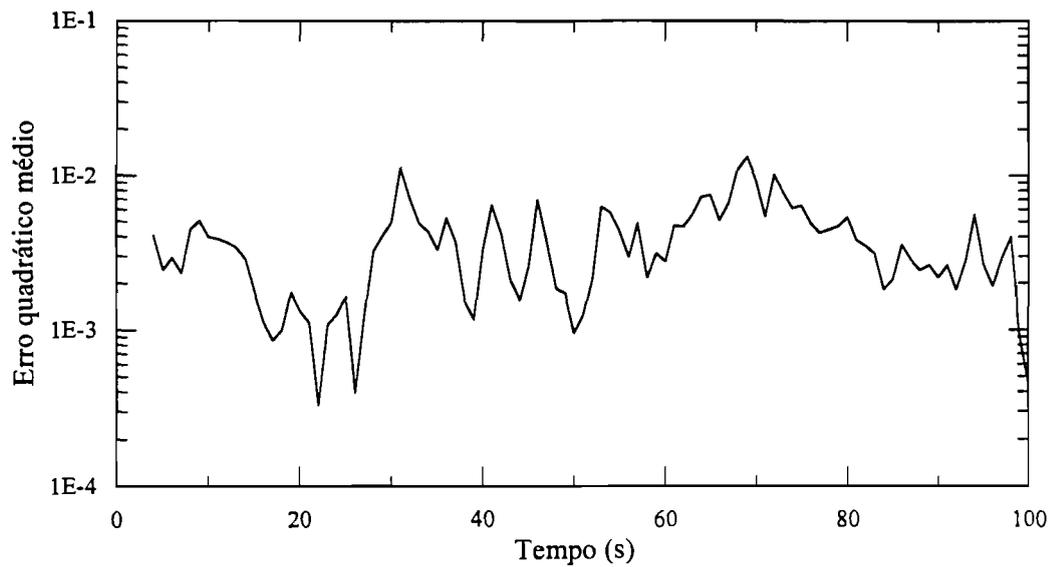


Fig. 5.23 - Erro apresentado pela rede numa simulação após o treinamento.

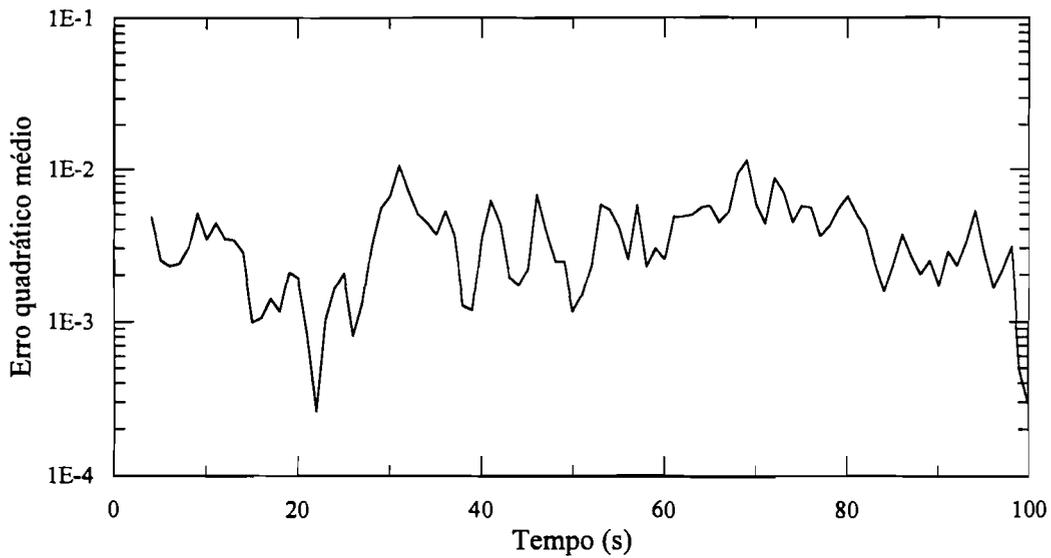


Fig. 5.24 - Erro da rede com sinal nulo nas entradas referentes aos propulsores.

É possível que a utilização de redes do modelo direto com maior número de neurônios na camada oculta ou com mais camadas possam melhor representar o comportamento dinâmico do satélite e, assim, viabilizar o treinamento especializado inverso. No entanto, o método de treinamento através dos mínimos quadrados utilizado nesta simulação mostrou-se extremamente lento quando o número de neurônios foi aumentado, além da necessidade de grande quantidade de memória, como já foi dito, para armazenar as matrizes de covariância. Por sua vez, o método do gradiente (retro-propagação), cujo tempo para processar um único ponto é proporcional ao número de neurônios da rede - enquanto que nos mínimos quadrados o tempo é proporcional ao quadrado - não atinge a precisão desejada, caso seja utilizado um parâmetro de convergência λ alto, ou então estabiliza num patamar de erro relativamente alto (além do treinamento ser muito lento) no caso de λ pequeno, da ordem de 10^{-4} .

Passou-se, então, a uma tentativa de treinar a rede de controle através do método indireto. Também neste caso não foram obtidos os resultados esperados. Como salientado em diversos trabalhos, este tipo de treinamento pode levar a uma situação onde a rede apresenta a mesma saída para duas entradas distintas, o que,

infelizmente, invariavelmente aconteceu, qualquer que fosse o algoritmo de treinamento (mínimos quadrados ou gradiente) utilizado. O aprendizado, realizado sem informações suficientes a respeito do comportamento dinâmico da planta, levou a rede a responder com os motores saturados com as ambas as entradas, y^d e y^p (ver Figura 3.5). Já que ambas possuem a mesma saída, mesmo considerando entradas distintas, o erro é praticamente nulo e a rede não consegue mais aprender.

O terceiro método de treinamento, o inverso generalizado, não foi testado, por requerer, a princípio, o conhecimento da lei de controle. Embora seja possível controlar o satélite através de um controlador PID (proporcional ao erro, derivativo e integral), o treinamento da rede através deste procedimento não teria benefício outro do que controlar a atitude através de uma rede neural que emula um controlador PID. Durante os testes de validação do treinamento, utilizou-se deste procedimento algumas vezes, para comprovar o aprendizado da rede. Em todos eles, a rede adquiriu quase sem dificuldade a informação necessária para se comportar como um PID. Uma possível vantagem deste processo seria a possibilidade de utilizar a rede como um controlador adaptativo, através do ajuste ótimo de dois conjuntos distintos de ganhos para o controlador PID: um antes e outro após a abertura dos painéis. A rede utilizaria, neste caso, a informação do sensor de abertura dos painéis para ajustar os pesos a uma ou outra forma de controle.

Todos os procedimentos de obtenção do modelo inverso apresentaram, na verdade, um ponto fraco no que se refere à estabilidade do treinamento. Foi freqüente, nos casos citados, que o desconhecimento inicial da rede inversa levasse o sistema, através de excitação errática, a regiões afastadas da trajetória de referência. Quando isto ocorre, é muito difícil que a repetição do processo de treino acabe por levar o sistema para a trajetória esperada. Uma forma de contornar este problema é com o uso do aprendizado com realimentação do erro, proposto por Kawato (Kawato et al., 1988; Nascimento Jr., 1994), apresentada na Seção 3.4. A presença do controlador PID na malha de realimentação da planta evita o inconveniente,

mantendo assim o sistema sob controle durante o processo de treino. Mesmo assim, dependendo dos parâmetros utilizados na alimentação da rede de controle, pode ocorrer uma situação de equilíbrio, no qual o sinal desta é igual em módulo, mas em direção contrária à realimentação provida do controlador PID (ver Figura 3.8). Neste caso, o controle introduzido no sistema é nulo e o erro permanece estacionário, realimentando assim o controlador PID. Este aspecto é particularmente prejudicial quando a rede e o controlador PID competem entre si, tentando um anular o outro, e provocam a oscilação do sistema. Embora, nesta situação, o sinal de controle varie com pequena amplitude, a oscilação em grande amplitude de ambos, rede e controlador PID, faz com que a rede não consiga minimizar o sinal do erro na malha de realimentação. Para contornar este problema, Kyung (Kyung et al., 1994) sugere a inclusão de um limitador na aceleração provocada pela rede. A forma com que este limitador é introduzido é apresentado na Figura 5.25. Ao sinal de treinamento da rede provindo do controlador PID é adicionado um valor proporcional à aceleração do sistema, na forma:

$$a^d = v \operatorname{sgn}(\ddot{y}^d - \ddot{y}^p) \left(\frac{1 - e^{-\kappa\gamma}}{1 + e^{-\kappa\gamma}} \right), \quad (5.5)$$

onde v é um ganho da aceleração, e sgn representa a função sinal. A segunda parcela é a função sigmóide, de argumento $\kappa\gamma$, sendo que κ é uma constante e γ é dado pelo módulo da aceleração:

$$\gamma = \|\ddot{y}^d - \ddot{y}^p\|. \quad (5.6)$$

Este limitador da aceleração tem, portanto, o mesmo sentido que o desvio na aceleração, porém compreendido entre os extremos $-v$ e $+v$, mesmo que a diferença entre as acelerações de referência e do satélite ultrapasse este valor. Na prática, os ganhos v e κ devem ser ajustados empiricamente, a partir dos resultados obtidos durante o treinamento. A constante κ , de dimensão igual ao inverso da aceleração, assumiu o valor de 500, equivalente a uma aceleração de 0,0059 quando

o limitador atinge 90% do valor de v . Note que a máxima aceleração angular neste satélite é da ordem de 0.024 rd/s^2 . Portanto, o limitador tem como função não permitir que os motores saturarem por completo.

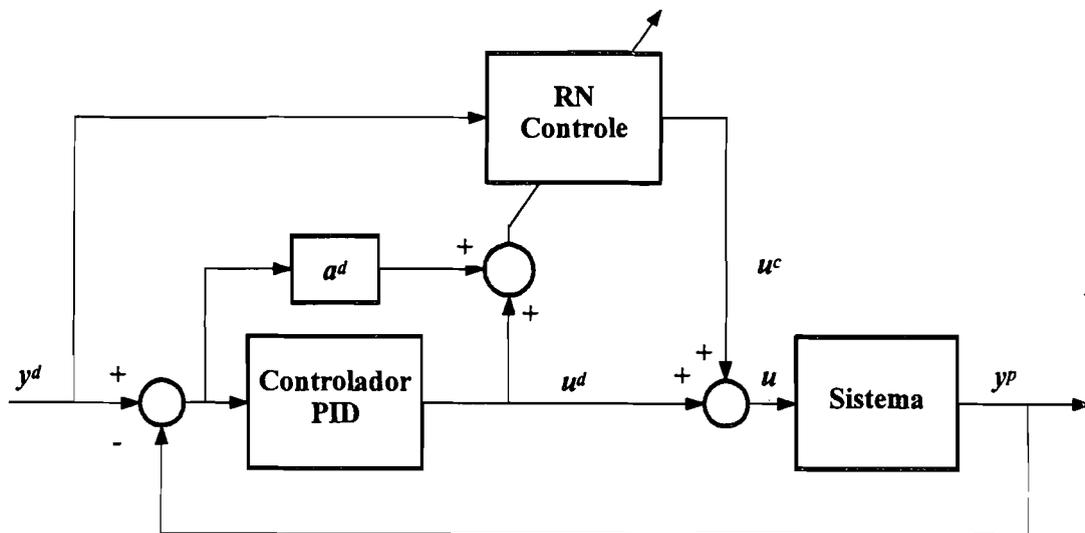


Fig. 5.25 - Treinamento com realimentação do erro e controle na aceleração da rede.

A magnitude dos pesos da rede neural no início do treinamento é fundamental no aprendizado com realimentação do erro. Para evitar que o sinal de controle provindo da rede conduza o sistema a um estado longe da trajetória de referência, deve-se iniciar os pesos com valores pequenos (Kyung et al., 1994).

A aceleração \ddot{y}^d é obtida ao mesmo tempo em que é gerada a trajetória de referência, enquanto que para se ter a aceleração do sistema, \ddot{y}^p , foi necessário fazer uma simplificação ao supor que a matriz de inércia do satélite fosse diagonal, e, desta forma, a aceleração é dada pela razão entre o torque aplicado e o momento de inércia daquele eixo.

Note que a aceleração, definida pela Equação 5.5, é vetorial, de onde pode-se obter o sinal de treinamento da rede neural de controle:

$$\varepsilon = u^d + a^d \quad (5.7)$$

Esta expressão mostra, no entanto, que o erro a ser minimizado é diferente daquele estabelecido na formulação do treinamento pelo método do gradiente e mínimos quadrados (Equação 2.30). Veja que, na atualização dos pesos da rede, conforme a Equação 2.55, a magnitude do vetor de erros desempenha papel fundamental na convergência do aprendizado. Embora o método do gradiente apresente esta dependência, ainda assim a convergência pode ser garantida através da seleção de um valor pequeno para o parâmetro λ (Equação 2.32). De fato, as simulações mostraram que existe um valor crítico de λ acima do qual o aprendizado não mais converge. O método dos mínimos quadrados, entretanto, além de apresentar também a dependência da magnitude do erro, tem como característica uma convergência muito rápida, e, portanto, caso a direção do vetor de erros ε esteja próxima, mas seja diferente daquela esperada, a convergência já não pode ser garantida. De fato, durante as simulações não se conseguiu treinar a rede de controle usando o algoritmo dos mínimos quadrados, e portanto passou a ser empregado o método do gradiente.

A atualização dos pesos foi realizada em cada instante de propagação da atitude. Foram realizados alguns testes com atualização ao final de uma integração completa, porém os resultados pouco diferiram dos primeiros.

No treinamento com realimentação do erro, a trajetória de referência deve ser fixa, isto é, calculada a priori, para evitar que a introdução da dinâmica da planta (através das leituras dos sensores) induza oscilações na saída da rede neural. Foram então adotados na simulação dois tipos de trajetória de referência: exponencial e de tempo mínimo. Ambas são realizadas em torno do eixo de rotação de Euler (Wertz, 1978), descrito da seguinte forma: dados dois sistemas de coordenadas (neste caso a atitude inicial do satélite e a atitude à qual deseja-se levá-lo) existe uma direção em

torno da qual uma rotação de um determinado ângulo conduzirá o primeiro sistema ao segundo. A direção de rotação pode ser obtida diretamente a partir dos quatérnions que relacionam o primeiro com o segundo sistema de coordenadas:

$$\phi = 2 \arccos(q_4) \quad (5.8)$$

$$e_v = \frac{q}{\sin(\phi / 2)}, \quad (5.9)$$

onde e_v é o vetor em torno do qual é realizada uma rotação do ângulo ϕ . q_4 é a quarta componente do vetor de quatérnions, q , e, na definição de e_v , somente são utilizados as três primeiras componentes de q .

A trajetória exponencial foi estabelecida então como:

$$\varphi(t) = \phi (1 - e^{-t/t_m}) \quad (5.10)$$

onde $\varphi(t)$ descreve, no instante t , o ângulo de rotação de Euler. No início da trajetória, $\varphi(t)$ é nulo, fazendo com que os sistemas de coordenadas, inicial e referência coincidam. Para t suficientemente grande, $\varphi(t)$ aproxima-se de ϕ . Na trajetória de tempo mínimo, busca-se acelerar o satélite durante metade do percurso, e então reduzir a velocidade angular durante a segunda metade:

$$\varphi(t) = \begin{cases} 0,5 \alpha t^2, & \text{para } 0 \leq t < \sqrt{\phi / \alpha} \\ 2 \sqrt{\phi \alpha} t - \phi - 0,5 \alpha t^2, & \text{para } \sqrt{\phi / \alpha} \leq t < 2 \sqrt{\phi / \alpha} \\ 0, & \text{para } t \geq 2 \sqrt{\phi / \alpha} \end{cases} \quad (5.11)$$

Os parâmetros t_m e α foram ajustados de forma a garantir que durante o treinamento da rede os motores não saturassem. Chegou-se, assim, a valores de 20 s e 0,003 rd/s² para t_m e α , respectivamente. Quando os motores saturam, a rede neural não consegue, por si só, prover o satélite com o torque necessário, e o controlador

PID passa a intervir. Isto ocorre durante a abertura dos painéis, como será visto mais adiante. A trajetória gerada pela exponencial, além de não ser dinamicamente compatível, pois necessita de uma aceleração infinita no instante $t=0$, também não trouxe contribuições para a simulação, já que os resultados foram bastante semelhantes àqueles oriundos da trajetória de tempo mínimo.

O treinamento foi realizado a partir de uma atitude escolhida aleatoriamente, como ponto de partida, até uma atitude final, dada pelos ângulos $\phi = -120^\circ$, $\eta = 45^\circ$ e $\psi = -90^\circ$ referentes a uma rotação 3-1-3 com relação ao sistema de coordenadas inerciais (eixo x coincidente com o ponto Vernal, eixo z na direção do pólo norte da Terra). Foram realizados treinamentos com diferentes parâmetros de convergência λ , compreendidos entre 0,01 a 0,00001. Verificou-se que para valores altos a convergência do treinamento é comprometida, enquanto que a precisão final atingida não melhora muito quando este parâmetro é reduzido. Em vista disso, adotou-se para λ o valor 0,001. A utilização de valores pequenos para o parâmetro de convergência pode levar a uma solução a um mínimo local, onde o erro é ainda muito grande. Para evitar esta situação, alguns autores (Chen e Billings, 1989), sugerem que a atualização dos pesos incorpore a informação da direção seguida na última iteração, na forma:

$$W^k(t+1) = W^k(t) - \lambda \nabla J^k + \lambda_m \Delta W^k(t) \quad (5.12)$$

onde $\Delta W^k(t)$ é definido como:

$$\Delta W^k(t) = W^k(t) - W^k(t-1) \quad (5.13)$$

e λ_m é o parâmetro de momento, compreendido entre 0 (não incorpora nenhuma informação anterior) e 1. No treinamento realizado, a convergência foi acelerada com valores de λ_m acima de 0,5.

O número de entradas atrasadas no tempo (trajetória de referência) fornecidas à rede também foi analisado, utilizando-se de 3 a 6 valores, incluindo o instante atual. As diferenças no resultado foram pequenas, e por isso adotou-se 4 informações atrasadas mais o instante t atual. O mesmo pode ser dito com relação ao número de entradas referentes à referência futura, onde testou-se de 1 a 3 informações, utilizando-se por fim somente uma informação futura em $t+1$.

O número de neurônios na camada oculta foi também investigado, obtendo-se precisões semelhantes a partir de 10 neurônios na camada oculta. O aumento deste número (foram realizados testes com até 100 neurônios) não trouxe benefícios ao treinamento e, conseqüentemente, fixou-se 30 neurônios na rede de controle.

A rede mostrou também pouca suscetibilidade a variações na magnitude inicial dos pesos (adotou-se 0,1) e ao valor da semente aleatória (0). O limitante da aceleração, no entanto, provou ser muito útil para evitar que a rede neural competisse com o controlador PID, mas, por outro lado, inibiu também a possibilidade de comandos rápidos realizados pela rede. O coeficiente teve de ser ajustado ao valor mínimo, que neste caso resultou em 0,002.

Os ganhos do controlador PID foram ajustados empiricamente, evitando-se assim um controle ótimo do ponto de vista de desempenho. Embora fossem realizadas algumas tentativas de ajuste do ganho do controle integral, este na verdade não trouxe vantagens evidentes e, portanto, foi abandonado. Os ganhos dos controladores proporcional e derivativo foram, respectivamente, de 0,3 e 20. Todos estes ganhos são adimensionais, já que o sinal de controle foi estabelecido como confinado ao intervalo -1 a 1. Na conversão do sinal de controle para torque aplicado ao satélite, foi utilizado um fator de 0,2 Nm.

O treinamento foi conduzido até 5400 iterações. A cada iteração propagou-se a atitude por 100 segundos, com passo de 1 s - a abertura dos painéis ocorreu sempre no instante $t = 50$ s, supondo um desprendimento do painel superior. A partir de 500 iterações aproximadamente, a convergência reduziu seu ritmo, e o erro apresentado pela rede neural decresceu somente de 0,03 para 0,023 (Figura 5.26). Os pesos, entretanto, ainda não atingiram uma situação de equilíbrio, pelo que se pode observar do gráfico mostrado na Figura 5.27.

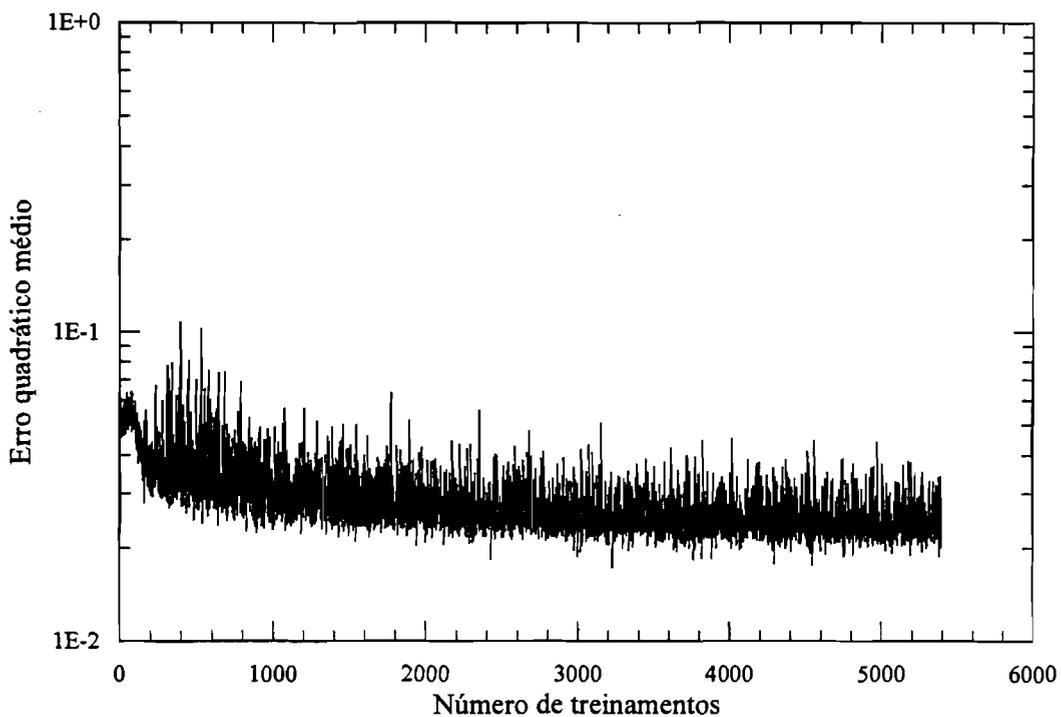


Fig. 5.26 - Erro apresentado pela rede de controle durante o treinamento.

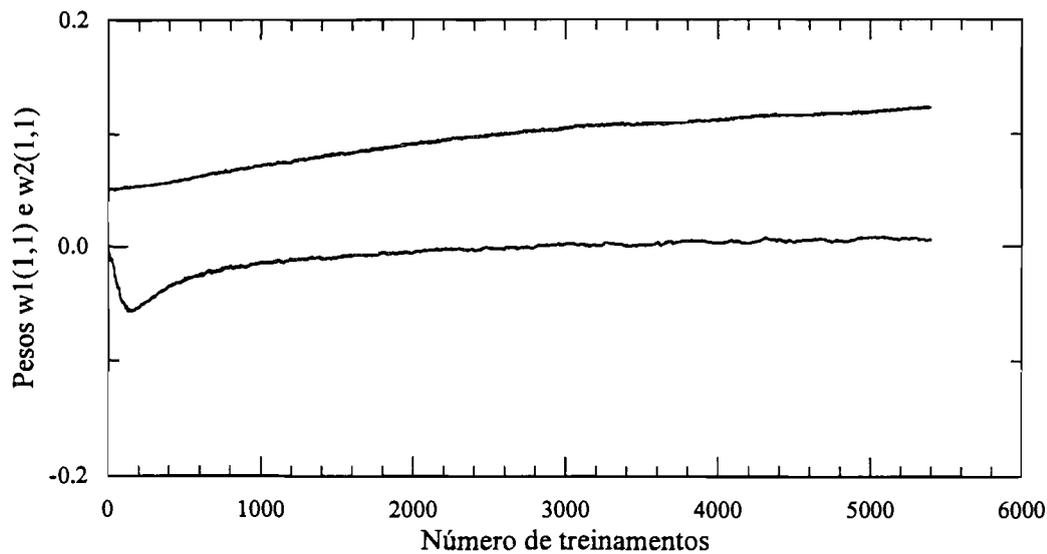


Fig. 5.27 - Variação dos pesos $W_{1,1}^1$ e $W_{1,1}^2$ durante o treinamento.

Após o treinamento, realizou-se uma simulação de uma manobra partindo de uma posição aleatória, e com a trajetória de referência conduzindo o satélite deste ponto até a posição final. Os resultados, mostrados nas Figuras 5.28 e 5.29, indicam que a rede foi capaz de assimilar o comportamento dinâmico do satélite, principalmente durante a abertura dos painéis. Na Figura 5.28 o controle é exercido somente pelo controlador PID, enquanto que na figura seguinte ambos, PID e rede neural, estão atuando simultaneamente. Ambas as figuras mostram a evolução da atitude em termos dos ângulos de Euler φ , θ e ψ referentes a uma rotação 3-1-3, respectivamente. O sobrescrito d refere-se à trajetória de referência, enquanto que o p é a atitude do satélite.

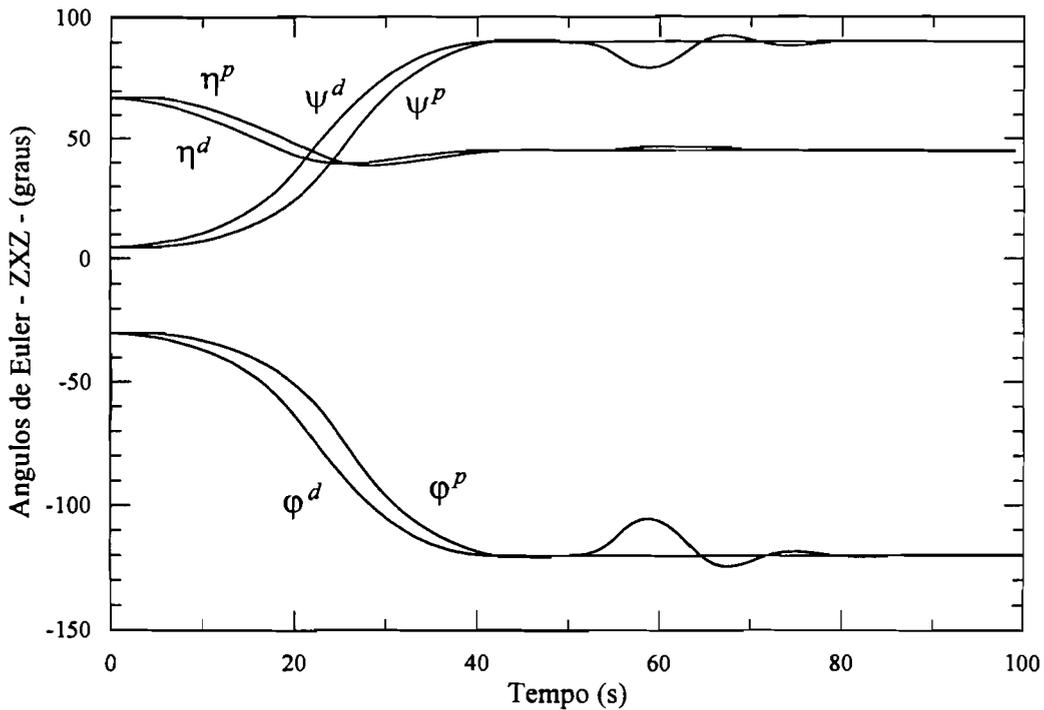


Fig. 5.28 - Evolução da atitude utilizando somente o controlador PID.

Os torques gerados nos três eixos do satélite quando somente o controlador PID está acionado são mostrados nas Figuras 5.30, 5.31 e 5.32. Pode-se notar que a perturbação na atitude gerada pela abertura dos painéis demanda um torque muito maior do que aquele necessário para efetuar a manobra, principalmente no eixo x^o , onde o controle chega a saturar. As Figuras 5.33, 5.34 e 5.35 apresentam ambos os torques, da rede neural e do controlador PID após o treinamento. Note, nas figuras, que durante os 40 segundos da manobra, a rede neural conseguiu assumir o papel de controlador principal do satélite, restando ao PID somente alguns pulsos eventuais. Embora, teoricamente, fosse possível eliminar totalmente a ação do PID neste fase, não se conseguiu ir além deste limite. Este resultado, embora comprove a versatilidade das redes neurais e a possibilidade de utilizá-las como controladores de atitude, adverte, entretanto, da impossibilidade de se dispensar o controlador PID e de se passar a confiar unicamente no controlador *feedforward*, pelo menos nas condições empregadas aqui.

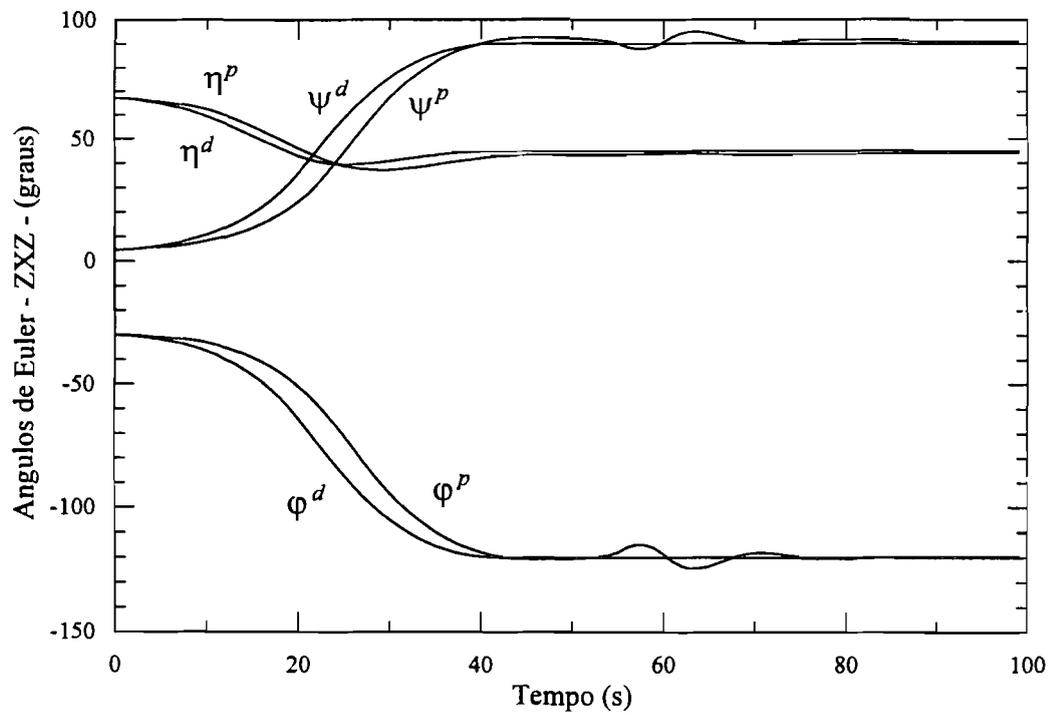


Fig. 5.29 - Evolução da atitude utilizando o controlador PID e a rede

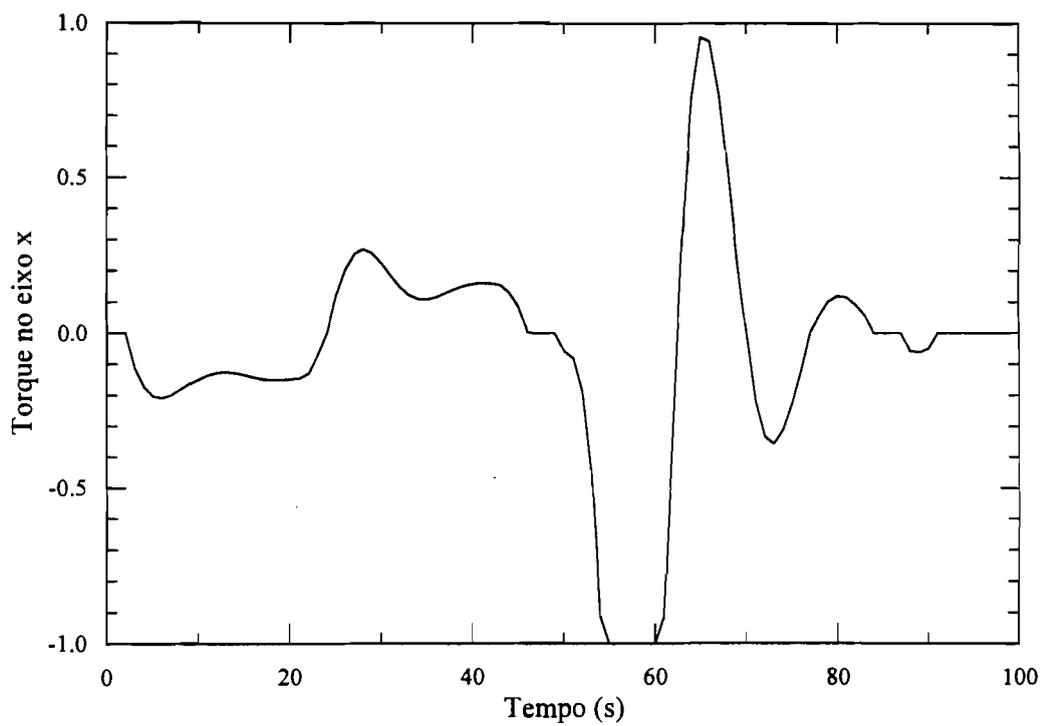


Fig. 5.30 - Torque no eixo x^o do satélite utilizando somente o controlador PID.

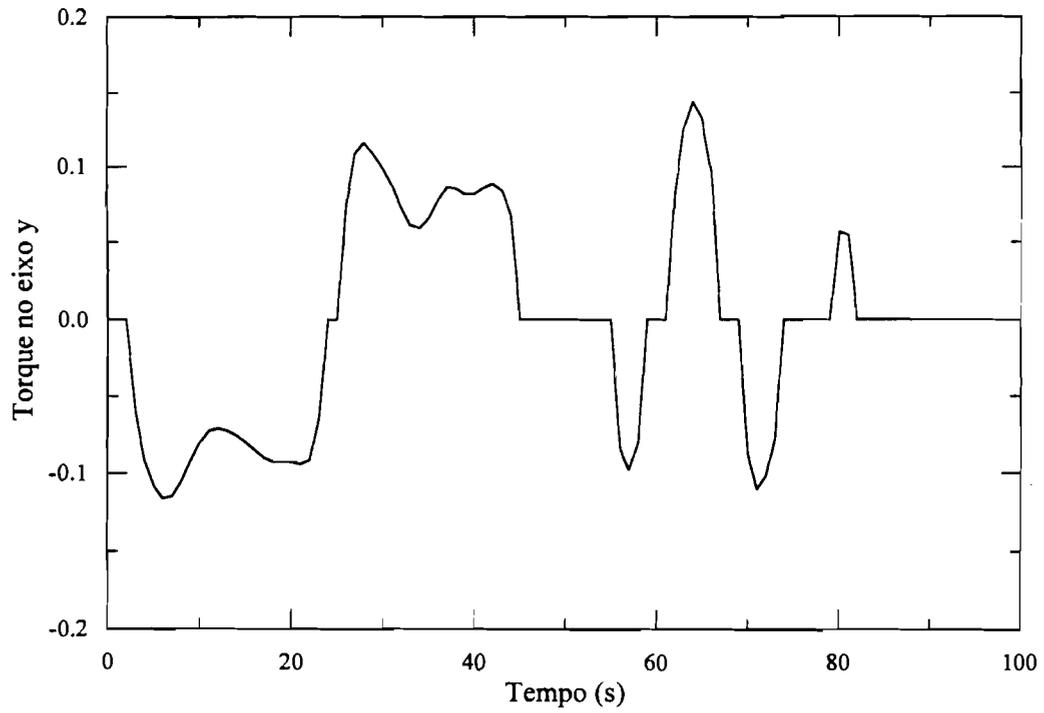


Fig. 5.31 - Torque no eixo y^o do satélite utilizando somente o controlador PID.

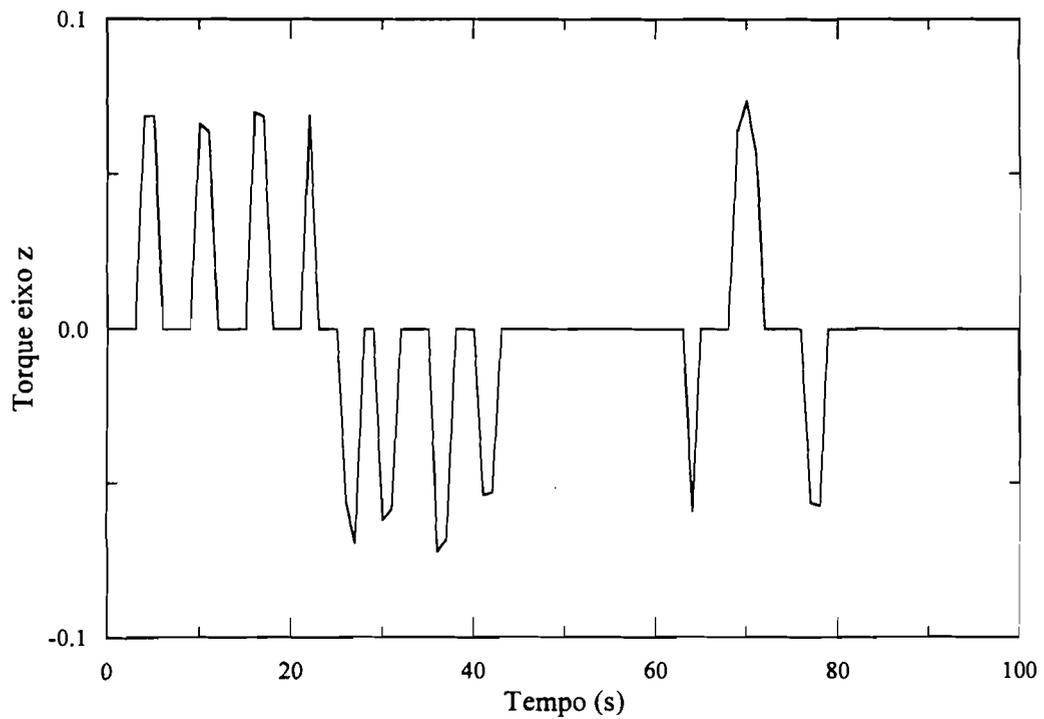


Fig. 5.32 - Torque no eixo z^o do satélite utilizando somente o controlador PID.

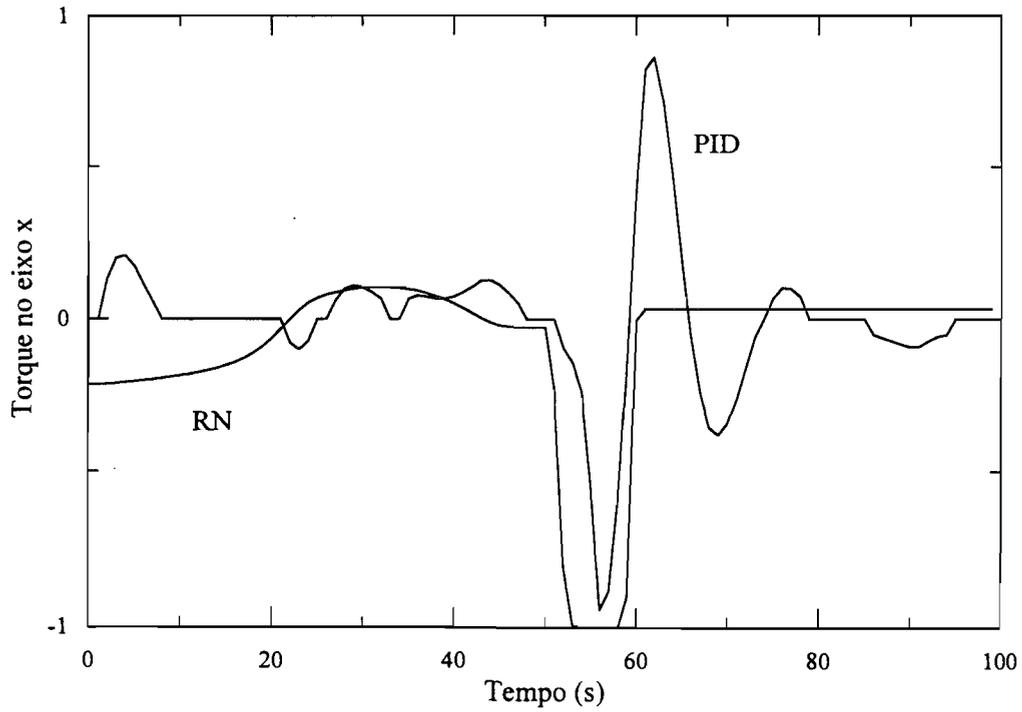


Fig. 5.33 - Torque no eixo x^o do satélite utilizando a rede neural e o controlador PID.

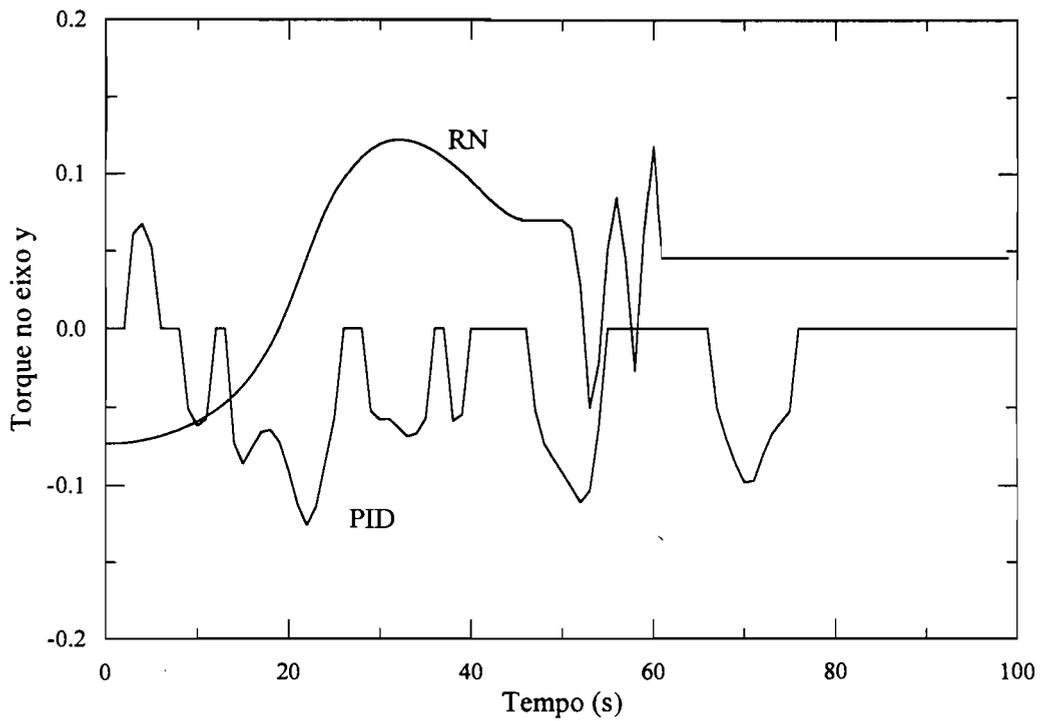


Fig. 5.34 - Torque no eixo y^o do satélite utilizando a rede neural e o controlador PID.

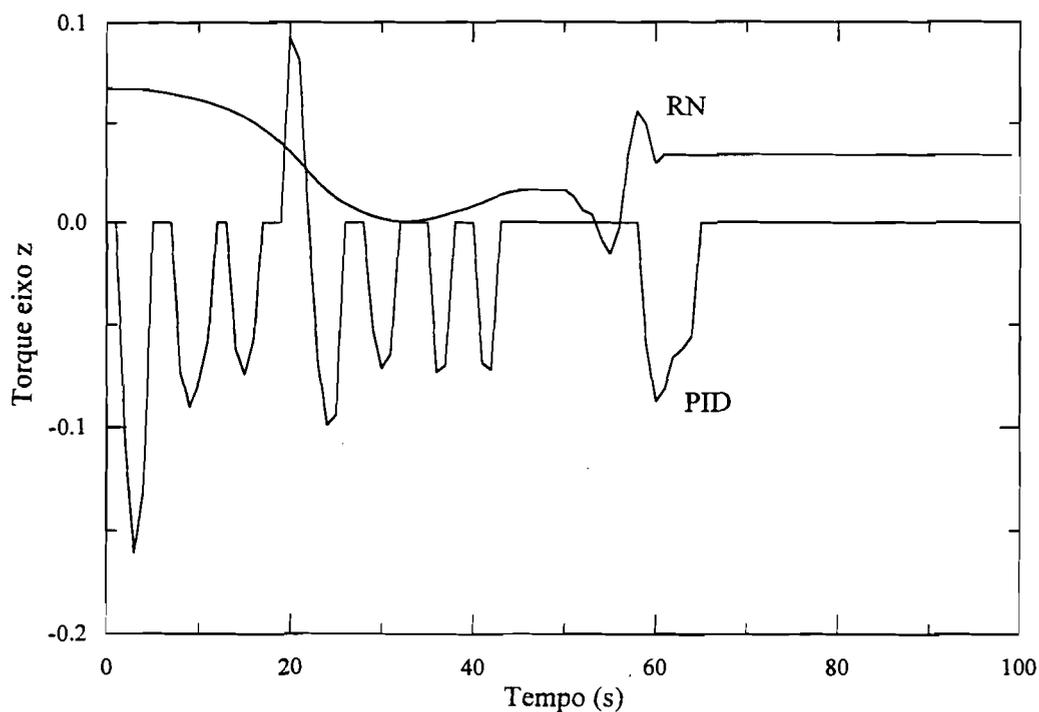


Fig. 5.35 - Torque no eixo z^o do satélite utilizando a rede neural e o controlador PID.

O desempenho do conjunto rede neural mais controlador PID durante a abertura dos painéis é sensivelmente melhor do que o uso somente do PID, conforme mostrado na Figura 5.29. Após a abertura, que se encerra em 60 segundos de propagação, a rede não mais consegue comandar torques porque não há informações que permitam um comportamento dinâmico, pois as entradas estão estabilizadas. Embora a rede tenha tentado minimizar o erro, a saturação dos motores, visto na Figura 5.33, impediu que o objetivo fosse atingido. Ainda assim, os controladores atuando em conjunto mostraram-se particularmente vantajosos durante a abertura dos painéis, como mostrado nas Figuras 5.36 e 5.37, que apresentam o erro cometido pelo controle nas duas configurações: somente PID e em conjunto com a rede neural. Veja que, enquanto o erro durante a manobra permanece quase que inalterado, o mesmo não acontece durante a abertura dos painéis. De fato, toda a informação necessária para controlar a atitude do satélite nesta fase foi obtida com o sinal provindo do sensor de abertura. Caso esta informação seja retirada da rede, o erro sobe para o mesmo patamar daquele onde somente o controlador PID está acionado.

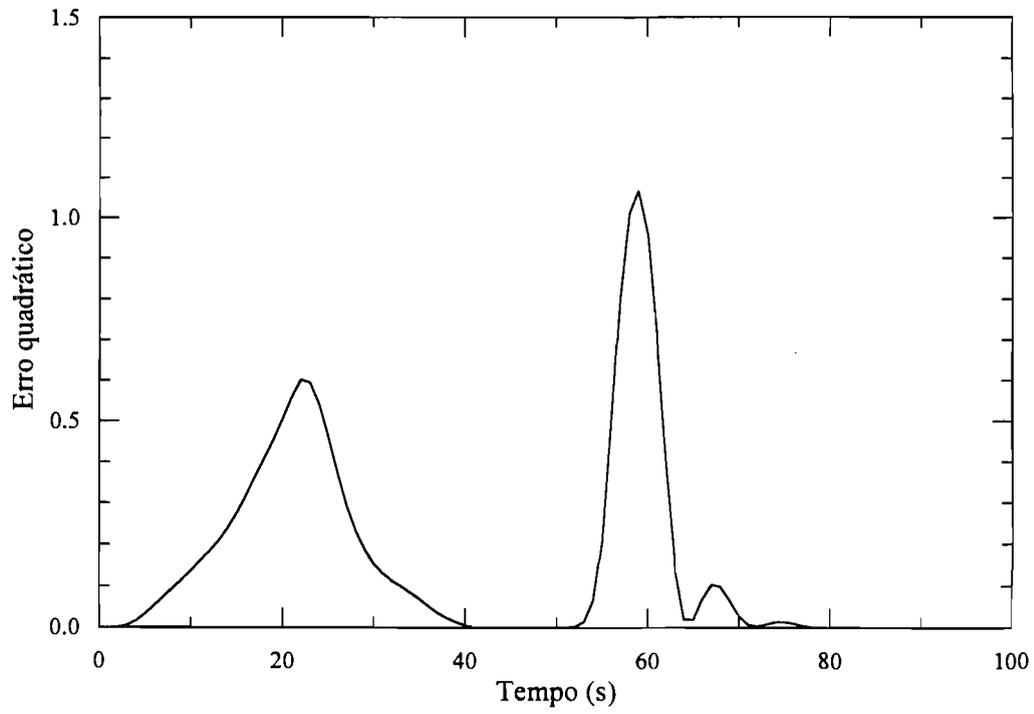


Fig. 5.36 - Erro quadrático do controlador PID.

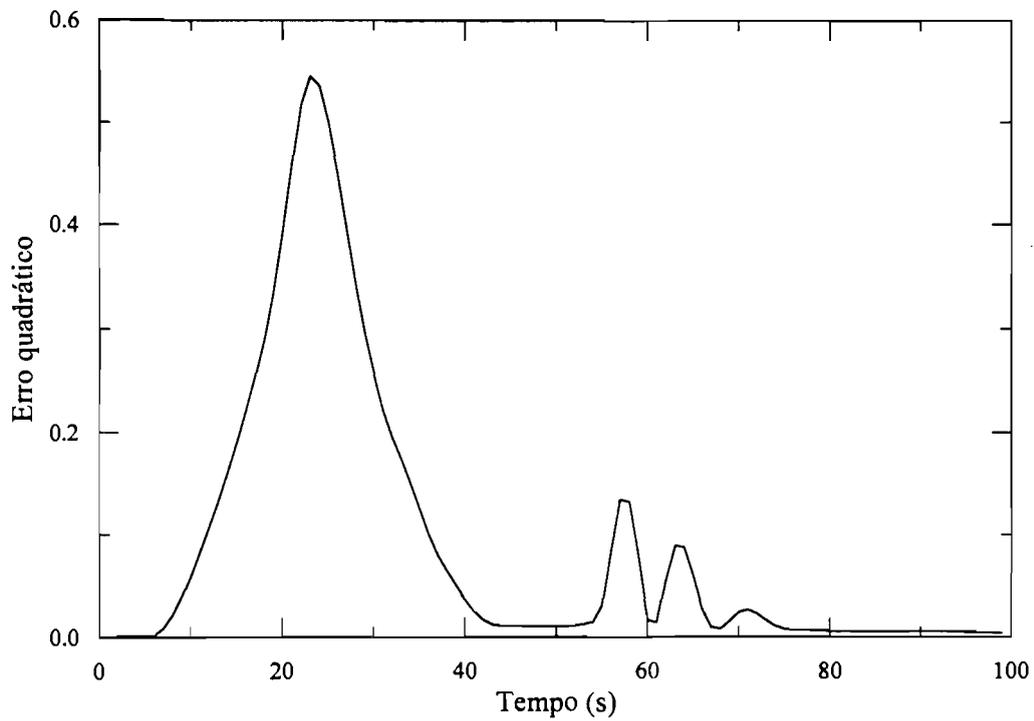


Fig. 5.37 - Erro quadrático do rede neural e controlador PID.

5.5 CONTROLADOR COM INCERTEZA NOS PARÂMETROS

Para comprovar a eficácia do controle através de rede neural, decidiu-se investigar o comportamento do erro em função de variações nos parâmetros do satélite, como por exemplo as massas e momentos de inércia do corpo principal e dos painéis, a posição do centro de massa, ruídos nas medidas dos sensores, etc. Inicialmente, analisou-se o erro considerando-se as seguintes modificações na massa e momentos de inércia, todas elas compatíveis com as tolerâncias permitidas ao projeto (compare com os valores mostrados no Apêndice C):

$$I_o = \begin{bmatrix} 22.14 & 0.1 & 0.1 \\ 0.1 & 21.63 & 0 \\ 0.1 & 0 & 10.50 \end{bmatrix} \text{ (kg. m}^2\text{)}, \quad (5.14)$$

$$m_o = 190.0 \text{ kg}, \quad (5.15)$$

$$I_1 = I_2 = \begin{bmatrix} 0.092 & 0 & 0 \\ 0 & 0.290 & 0 \\ 0 & 0 & 0.400 \end{bmatrix} \text{ (kg. m}^2\text{)}, \quad (5.16)$$

$$I_3 = \begin{bmatrix} 0.105 & 0 & 0 \\ 0 & 0.290 & 0 \\ 0 & 0 & 0.410 \end{bmatrix} \text{ (kg. m}^2\text{)}, \quad (5.17)$$

$$m_1 = m_2 = m_3 = 2.7 \text{ kg}. \quad (5.18)$$

A posição do centro de massa altera-se em função das novas relações de massa do corpo principal e dos painéis, o que é calculado automaticamente pelo programa de simulação. É natural que se tenha, devido a estas alterações, e também devido a desalinhamentos na montagem, um erro no torque gerado pelos propulsores.

Este erro foi exagerado, para certificar a capacidade do controlador, resultando para o torque no sistema de coordenadas do satélite: $N_x = 0,2 \tau_x + 0,01 |\tau_x|$, $N_y = 0,2 \tau_y - 0,02 |\tau_y|$ e $N_z = 0,18 \tau_z + 0,004 |\tau_z|$, onde τ_x , τ_y e τ_z compõem o sinal do controlador, compreendidos entre -1 e +1. Em certas situações, devido ao consumo de combustível do satélite, a migração do centro de massa pode alterar significativamente o torque aplicado pelos propulsores. Esta migração, entretanto, é lenta e pode ser negligenciada durante o intervalo de simulação. Conforme mostra a Figura 5.38, o erro do controlador (rede neural e PID), sofre somente pequenas alterações, passando de 0,15 para 0,25 durante o intervalo de abertura dos painéis. Fora deste intervalo, o erro permanece praticamente o mesmo. Este resultado confirma que a rede adquiriu o conhecimento necessário para controlar o satélite, pois mesmo nesta situação o erro ainda se manteve menor do que aquele do controlador PID atuando sozinho.

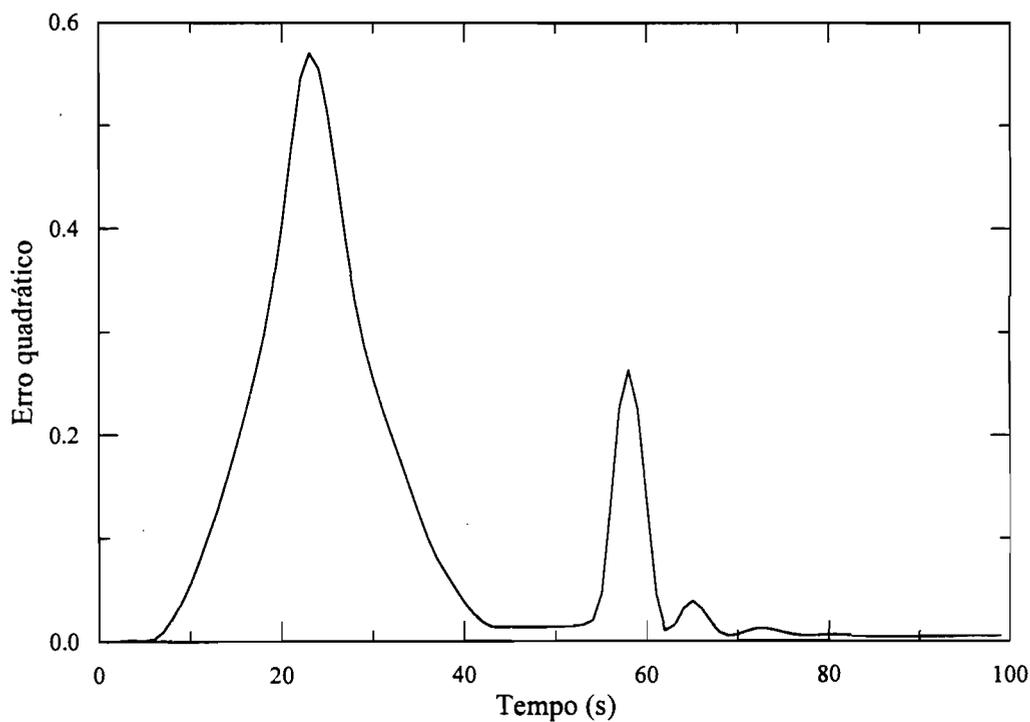


Fig. 5.38 - Erro quadrático com incerteza nos parâmetros do satélite.

Outro fator que influencia o comportamento do controle é a presença de ruídos nos sinais dos sensores. Não houve necessidade da inclusão do ruído durante o treinamento, pois este foi realizado a partir de simulação. Evitou-se, assim, que a rede tivesse que atuar também como filtro dos sensores, e o treinamento pode desta forma ser acelerado. O ruído introduzido tem distribuição normal com média nula e desvio padrão de 2% no sensor solar e 600 nT (equivalente a 2%) no magnetômetro. A Figura 5.39 mostra a evolução da atitude e a trajetória de referência considerando não somente o ruído nos sensores, mas também a incerteza nos parâmetros do satélite. Percebe-se, pela figura, que o erro introduzido pelo ruído nos sensores foi compensado, em parte, pelo controlador PID, já que a rede não possui informação que permita avaliar o desvio provocado pelos erros na medida. A falta de um filtro no sinal provindo dos sensores durante a simulação provocou uma resposta instantânea do controlador PID, acarretando os desvios na trajetória. A ausência de um filtro nas medidas impediu o treinamento da rede com realimentação do erro, justamente devido à presença de ruído no sinal de realimentação.

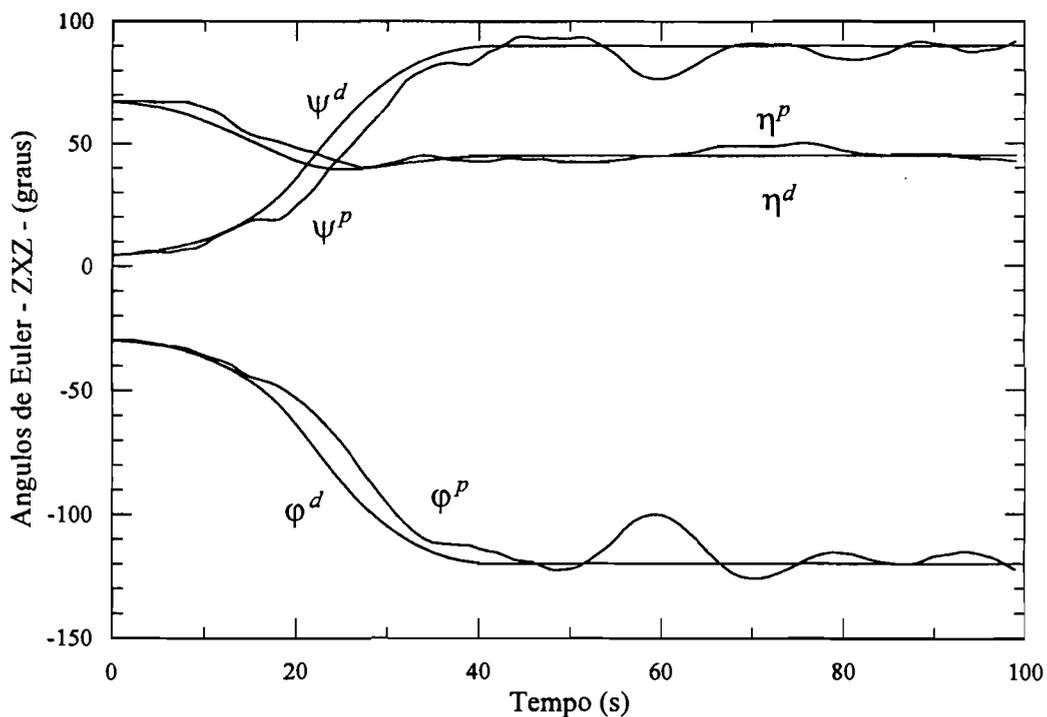


Fig. 5.39 - Evolução da atitude com incerteza nos parâmetros e ruído nos sensores.

O maior problema encontrado no método de realimentação do erro foi, na verdade, a incapacidade da rede de lidar com desvios da trajetória provocados por fatores a respeito dos quais a rede não possui informações (sinais de entrada). Enquanto houver alterações nos pontos fornecidos na trajetória de referência, a rede pode desenvolver uma dinâmica capaz de compensar efeitos diversos (como a abertura dos painéis, por exemplo). Uma vez que o sinal da trajetória estabilize, cessa também a capacidade da rede de produzir um controle dinâmico. Infelizmente, as tentativas de treinamento realizadas com a inclusão dos sinais provindo dos sensores também fracassaram, embora a rede tivesse adquirido, de certa forma, o aprendizado necessário para controlar o satélite. O que geralmente ocorre, neste caso, é uma competição entre a rede e o controlador PID, com um tentando anular o outro, isto é, mesmo módulo e sinais contrários. Quando ocorre a saturação dos propulsores, o torque efetivamente aplicado ao satélite se anula, pois rede e PID atuam de forma contrária. Quando o erro muda de sinal, ambos os controladores também se alternam. Na média, conforme a trajetória de referência evolui, a atitude tende a acompanhar esta trajetória, porém com um erro significativamente maior do que aqueles mostrados aqui, sem a inclusão das medidas dos sensores na alimentação da rede.

5.6 CONTROLADOR COM ABERTURA NORMAL DOS PAINÉIS

Nos exemplos anteriores foi suposto que houvesse uma falha durante a abertura, de tal forma que o painel superior desprendesse-se ao colidir com o batente. Este certamente é um caso crítico, embora com probabilidade de ocorrer reduzida. Para confirmar a capacidade das redes neurais de adquirirem o conhecimento do sistema numa situação normal, com os painéis abrindo segundo a Figura 5.3, realizou-se um novo treinamento, composto por 5000 simulações de 100 s cada, com a abertura sempre ocorrendo em $t = 50$ s. Os parâmetros utilizados foram iguais àqueles da Seção 5.4, ou seja: treinamento realizado com o algoritmo do gradiente, com 0,1 na amplitude de geração dos pesos e semente aleatória nula. A rede é composta das mesmas entradas, envolvendo os 6 sinais de referência para a trajetória

(3 do sensor solar e 3 do magnetômetro), os 3 sinais dos propulsores e o ângulo de abertura, todos fornecidos em 3 instantes passados e no atual. Os 6 sinais da trajetória no instante $t+1$ também fazem parte do conjunto de valores de entrada. Utilizou-se 30 neurônios na camada oculta. A atitude inicial foi escolhida aleatoriamente, o mesmo acontecendo com as velocidades angulares, porém limitadas a 0,5 rd/s em cada eixo. Os ganhos utilizados no controlador PID foram: proporcional de 0,3, integrativo nulo e derivativo 20. Adotou-se um coeficiente limitador da aceleração, ν (conforme a Equação 5.5) de 0,002 e um coeficiente de convergência do aprendizado, λ , de 0,001 (com λ_m nulo).

A Figura 5.40 mostra a evolução da atitude numa simulação efetuada após o treinamento. Veja no entanto que não houve diferença significativa entre a situação com abertura normal e a abertura com desprendimento do painel, conforme a Figura 5.29. As Figuras 5.41, 5.42 e 5.43, a seguir, apresentam os torques aplicados aos eixos x^o , y^o e z^o do satélite, respectivamente, distribuídos pela rede neural e pelo controlador PID. A comparação com um controle PID é realizada nas Figuras 5.44, 5.45, 5.46 e 5.47, que representam, respectivamente, a evolução da atitude e os torques nos 3 eixos do satélite. Pelo menos nesse caso, não houve vantagem aparente no uso da rede neural para controlar o satélite. Como pode ser visto nas Figuras 5.48 e 5.49, que mostram o erro quadrático médio da atitude nas duas formas de controle (com ou sem rede neural), o erro quadrático médio cometido pelo controlador PID, na ausência da rede neural é menor do que ambos controlando simultaneamente. Isto não necessariamente significa que um controlador PID sempre apresente desempenho melhor do que redes neurais, mas sim que o grau de aproximação da rede foi insuficiente para produzir resultados satisfatórios. O principal problema, sem dúvida, é a ausência de um sinal para o treinamento que corresponda aos valores apresentados na saída da rede, ou seja, o próprio controle. O número de parâmetros a serem ajustados, de forma a garantir que o aprendizado convirja, também tem um forte impacto no grau de aproximação da rede.

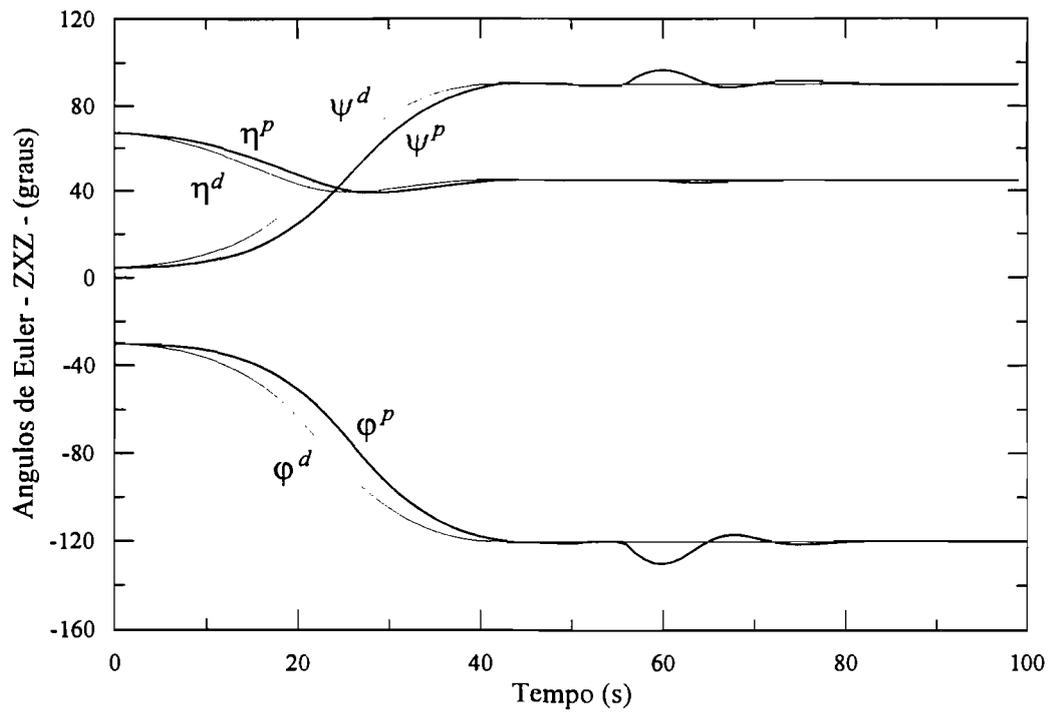


Fig. 5.40 - Atitude com abertura normal dos painéis, controle PID e rede.

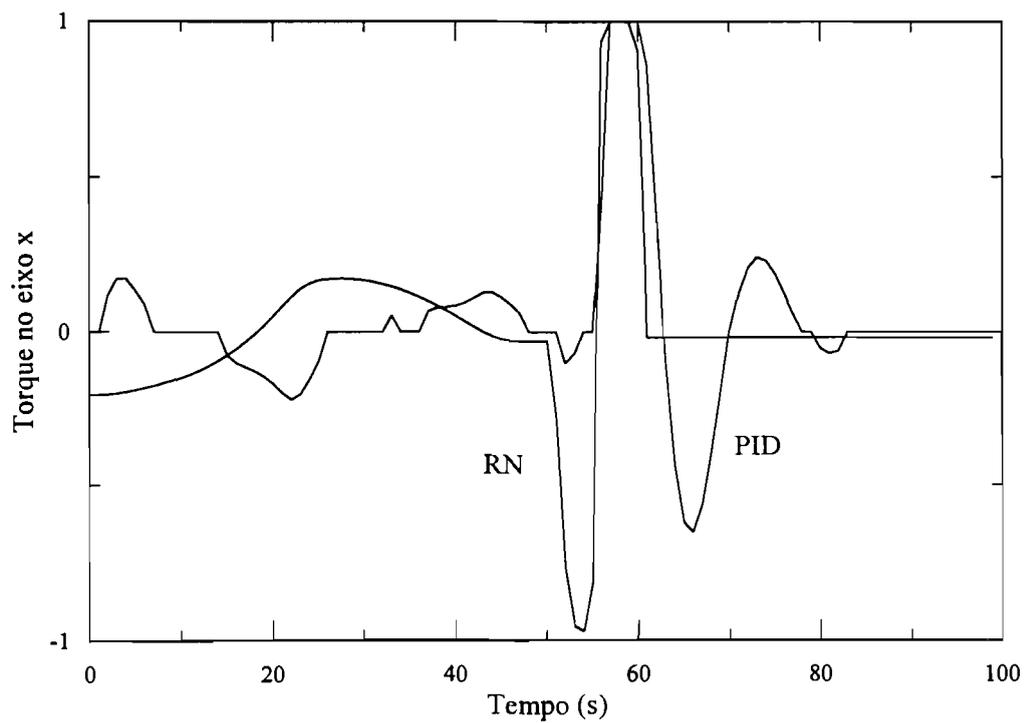


Fig. 5.41 - Torques no eixo x^o , com abertura normal dos painéis.

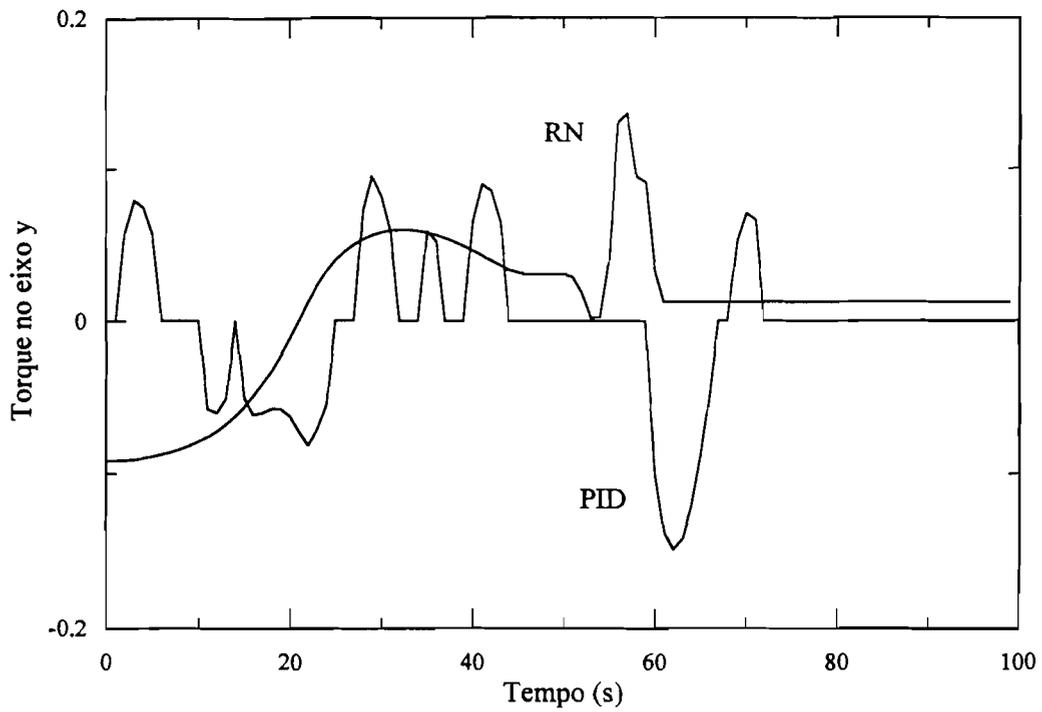


Fig. 5.42 - Torques no eixo y^o , com abertura normal dos painéis.

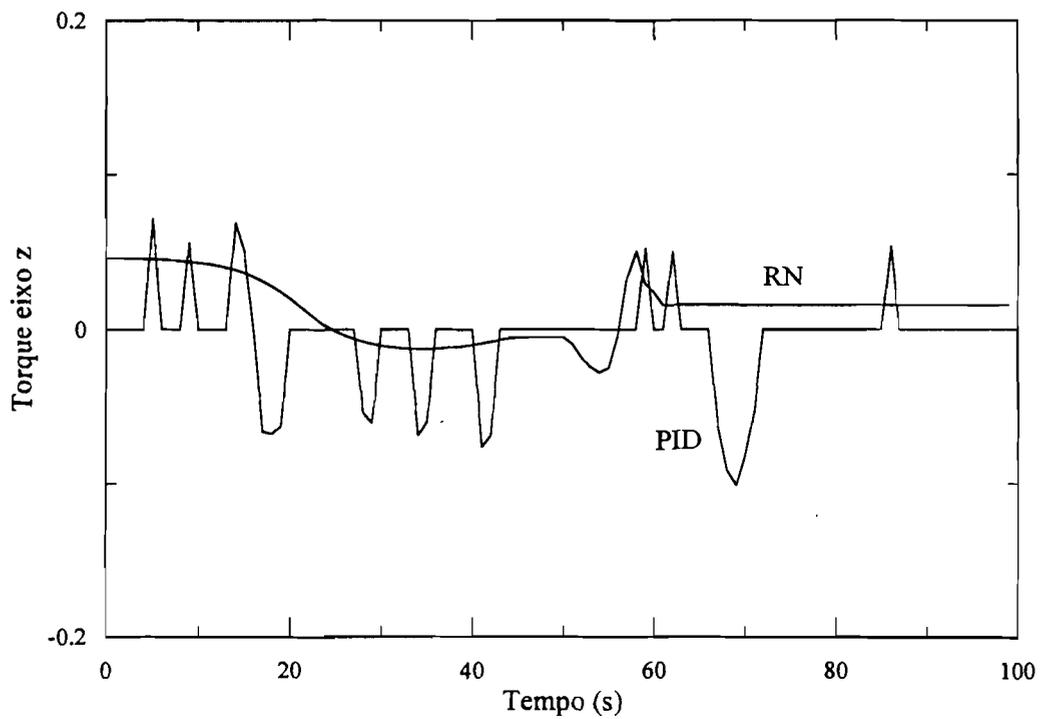


Fig. 5.43 - Torques no eixo z^o , com abertura normal dos painéis.

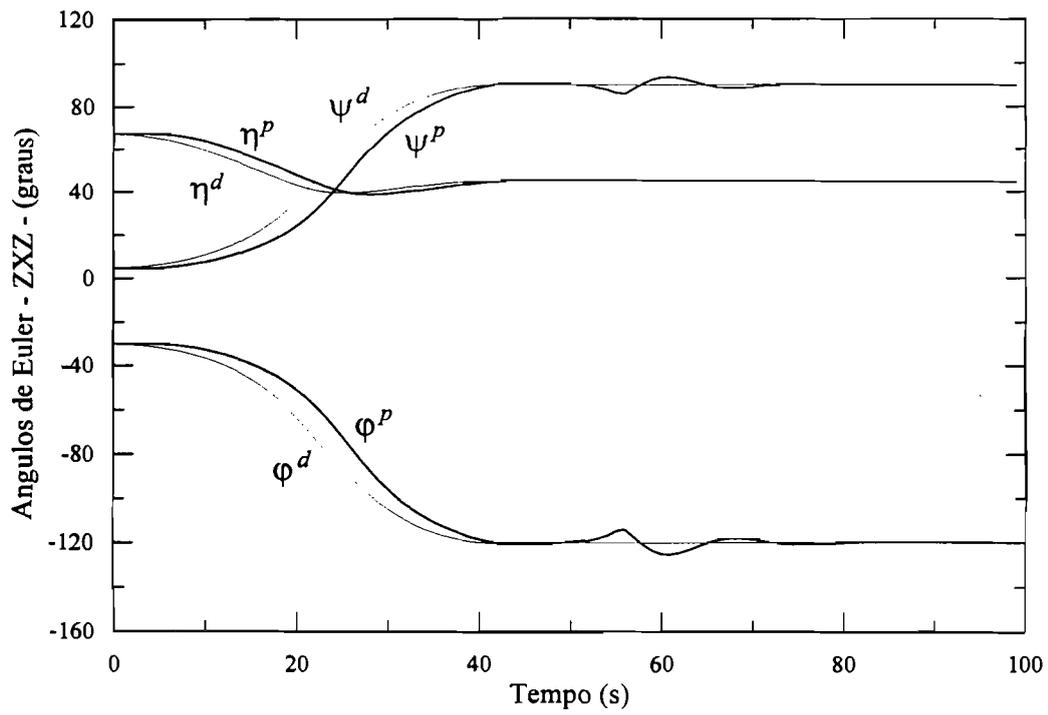


Fig. 5.44 - Atitude com abertura normal dos painéis, somente controle PID.

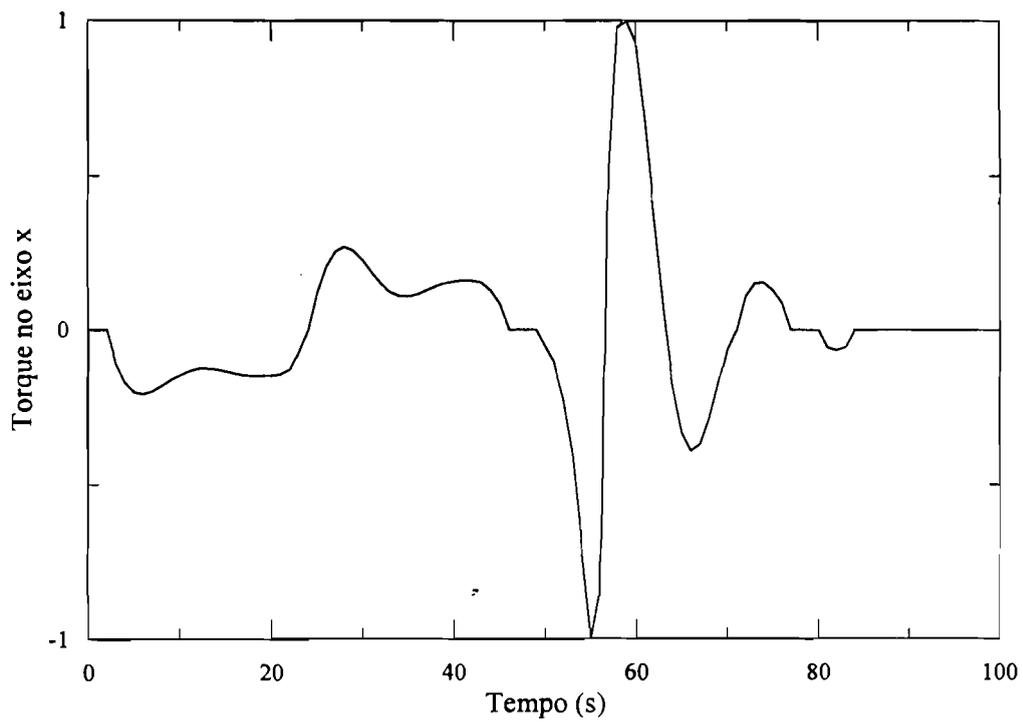


Fig. 5.45 - Torque no eixo x^o , com abertura normal dos painéis, controle PID.

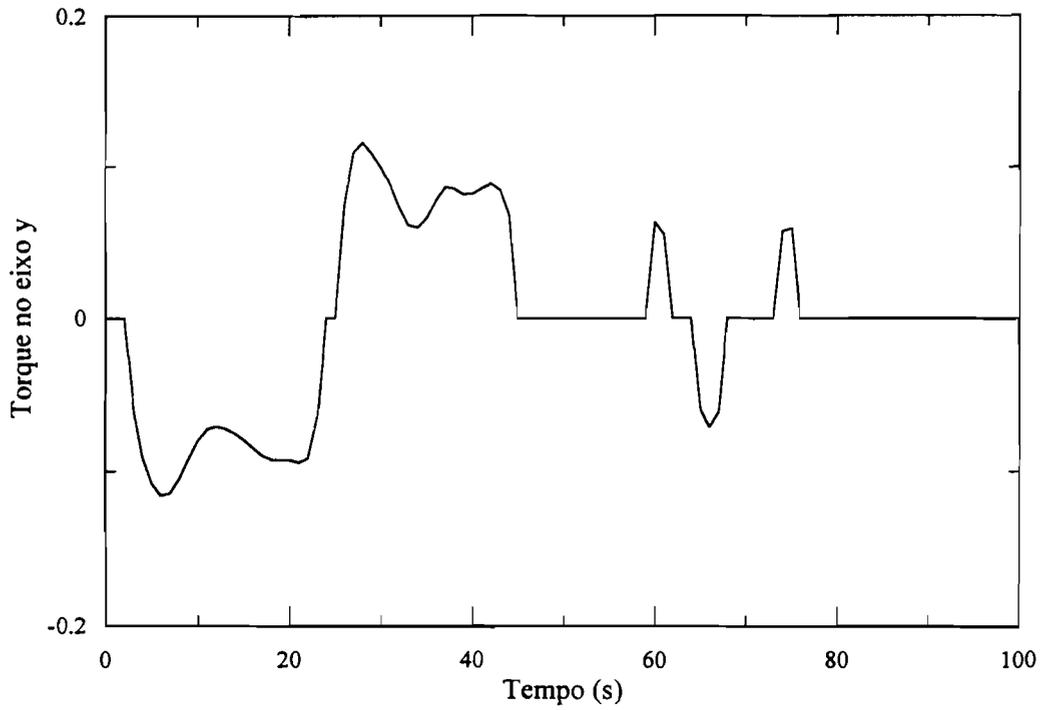


Fig. 5.46 - Torque no eixo y^o , com abertura normal dos painéis, controle PID.

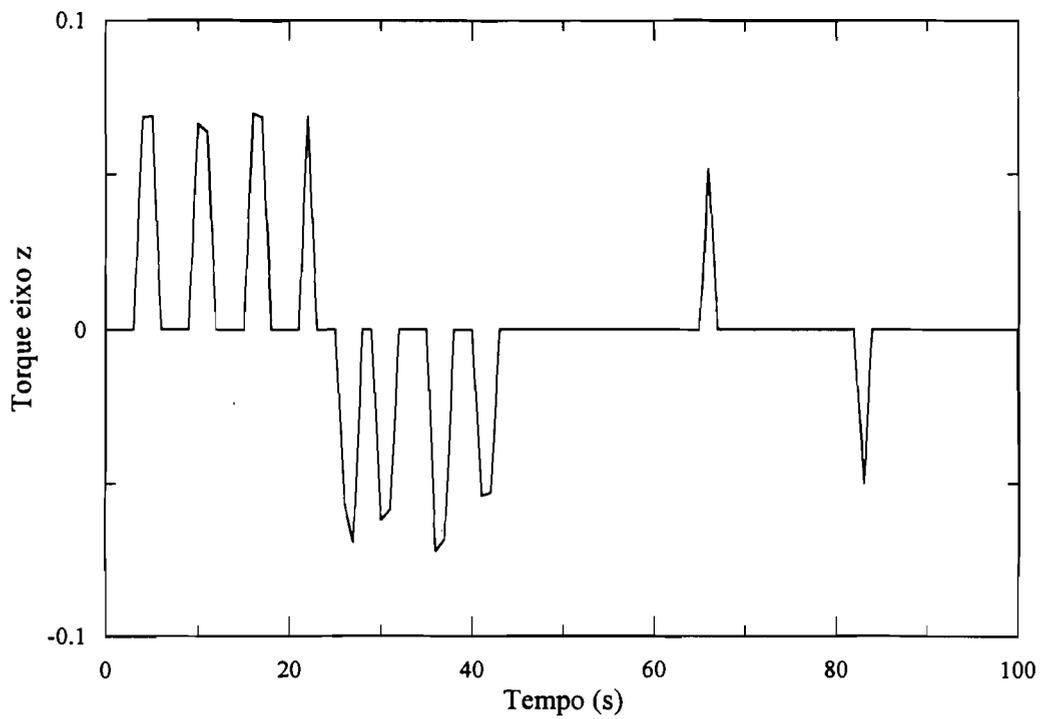


Fig. 5.47 - Torque no eixo z^o , com abertura normal dos painéis, controle PID.

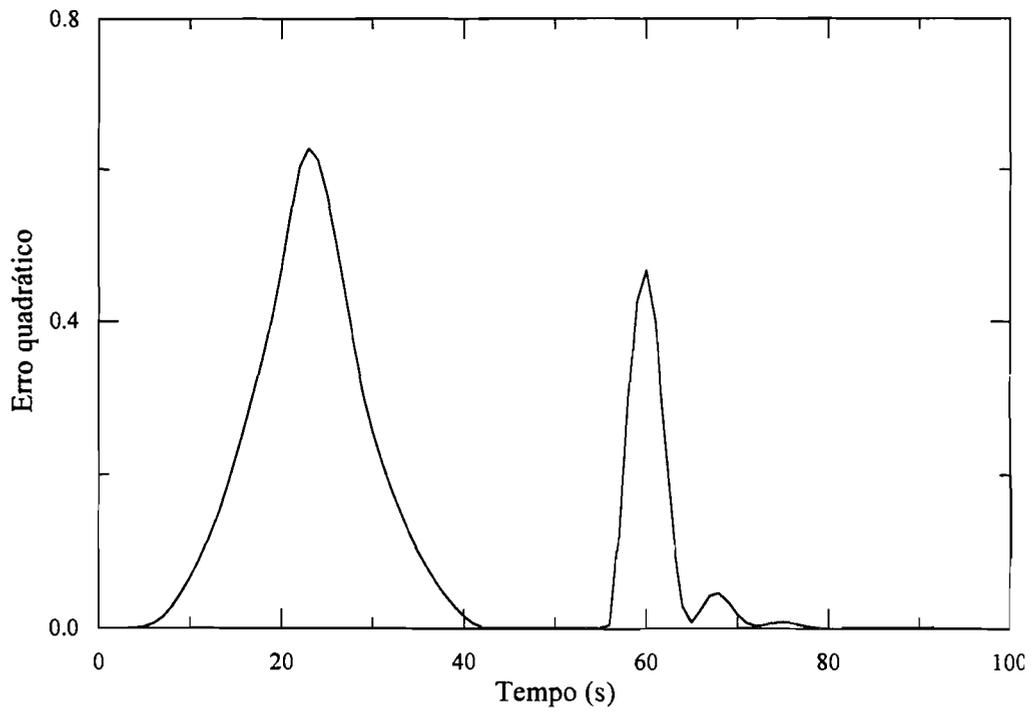


Fig. 5.48 - Erro quadrático com abertura normal, e controle efetuado pela rede e PID.

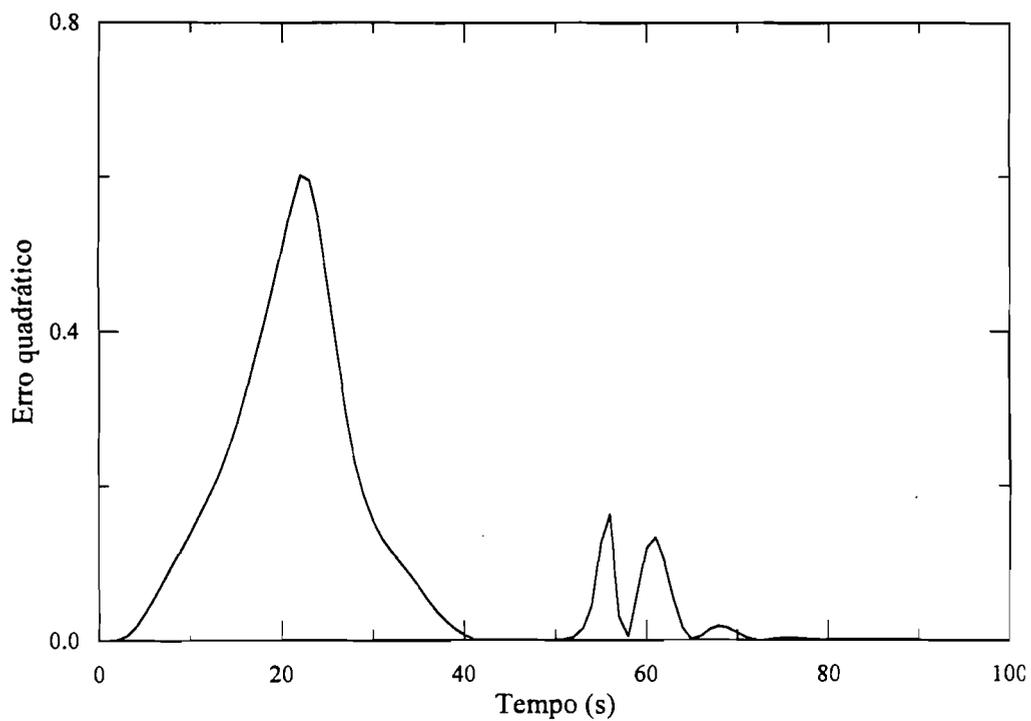


Fig. 5.49 - Erro quadrático com abertura normal e controle exercido apenas pelo PID.

5.7 CONTROLADOR SEM A SUPERVISÃO DO PID

Os problemas encontrados durante o treinamento do controlador com o método de realimentação do erro, como a competição observada entre o PID e a rede neural, além da ausência de um sinal que possibilitasse a rede a atuar mesmo quando a referência já havia estabilizado levou a busca para um esquema que compensasse estes inconvenientes. O método de realimentação do erro provou que pode fornecer um sinal de treinamento, embora não dirigido à minimização da diferença apresentada na saída da rede neural. A realimentação do erro pode ser utilizada desde haja uma correspondência entre o vetor y^p de observações (ou, eventualmente, o estado do sistema) e o sinal de controle, isto é, exista uma função matemática que corresponda cada valor da saída com um específico sinal de treinamento. Na verdade, o que esta função permite é substituir a matriz de derivadas parciais do sistema (Jacobiano), necessária para retropropagar o erro no vetor y^p através da planta e assim obter o sinal de erro diretamente no sinal de controle u^c , conforme mostrado na Figura 5.50. Esta função faz as vezes da matriz de derivadas, mas é possível que não leve a um controle baseado na dinâmica inversa do sistema, embora ainda assim garanta a controlabilidade numa região bastante ampla em torno do ponto de operação, onde as não linearidades certamente se fazem sentir. Na Figura 5.50, a rede é alimentada com informações do erro na saída do sistema, e é justamente este o sinal que se deseja minimizar. Para que ele tenha as características de um sinal de controle, aplicam-se ganhos proporcionais aos erros em velocidade, posição e erro acumulado em posição, e, a seguir, a rede é treinada no sentido de minimizar este valor. O ajuste dos ganhos foi realizado de forma empírica, até que a convergência do treinamento e a controlabilidade do satélite fosse adquirida. A solução apresentou, freqüentemente, oscilações devido a atrasos na realimentação. Para contornar o problema, foi adicionado um outro valor ao sinal do erro, proporcional ao próprio sinal do controle no mesmo instante, com ganho K_d , conforme a Figura 5.50. Este artifício mostrou-se capaz não somente de prover estabilidade no controle, mas também acarretou uma

diminuição no combustível consumido pelo controlador, já que minimizar o controle u^c significa também minimizar a energia consumida.

O aprendizado só convergiu num espaço bastante restrito de valores dos ganhos, o que pode dificultar a aplicação do método de realimentação do erro sem supervisão do controlador PID. Um parâmetro de convergência do aprendizado, λ (ver Seção 2.3.2), próximo de zero, irá compensar, em parte, o desconhecimento do Jacobiano do sistema e o ajuste imperfeito do sinal PID. Era de se supor, portanto, que a convergência do treinamento fosse garantida somente para um pequeno intervalo de valores de λ . De fato, se o valor do parâmetro fosse muito alto (superior a 0,04 no exemplo), o treinamento não mais convergia, provavelmente devido às discrepâncias entre o sinal de treino e a matriz de derivadas do sistema.

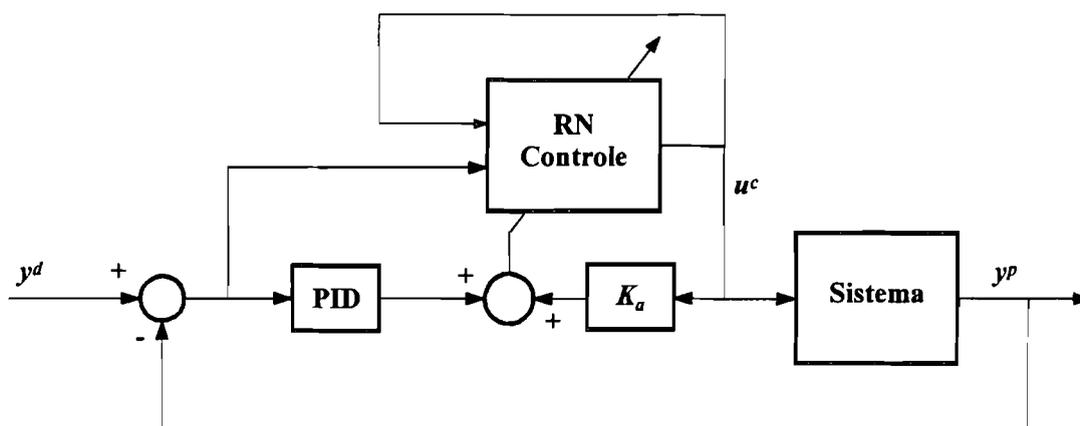


Fig. 5.50 – Controlador com realimentação do erro, sem supervisão do PID.

Para que a rede pudesse ser treinada até a estabilização completa do satélite, as simulações foram efetuadas em intervalos de 1000 segundos de propagação, com passo de 1 segundo. A abertura dos painéis sempre ocorreu no instante $t = 500$ s. As condições iniciais foram aleatórias, com ângulos compreendidos entre $\pm 45^\circ$, e velocidades angulares entre $\pm 0,5$ rd/s, valores estes bastante altos numa situação típica de satélites. As entradas da rede consistiram da atitude do satélite descrita por uma rotação XYZ (ângulos φ , η e ψ , respectivamente),

as velocidades angulares ω_x , ω_y e ω_z nos eixos do satélite, o valor do sinal de controle no instante anterior τ_x , τ_y e τ_z , e o ângulo de abertura θ dos painéis. Além destes, foram fornecidos também valores atrasados em 3 instantes passados, totalizando 40 valores. O patamar de ativação foi desativado, para evitar que a rede enviasse um sinal de controle mesmo quando as entradas fossem nulas. Foram utilizados 40 neurônios na camada oculta, com função de ativação do tipo sigmóide, e função linear limitada na camada de saída (3 neurônios). O aprendizado não foi muito influenciado pelo número de neurônios ocultos. Embora a qualidade do controle diminuísse com este número, ainda assim o processo convergiu com até 5 neurônios na camada oculta. Usou-se o próprio vetor de estados na alimentação da rede ao invés das leituras dos sensores para evitar o cálculo necessário para transformar as leituras no sinal de treinamento. Posteriormente testou-se o aprendizado a partir da leitura dos sensores, mas não obteve-se a convergência do treinamento. Como esta convergência é sensível aos ganhos do PID e também ao parâmetro λ , é possível que o treinamento pudesse convergir com a seleção adequada destes valores, como de fato funcionou no método de realimentação do erro, mostrado na seção anterior.

A Figura 5.51 mostra os resultados obtidos após um treinamento com cerca de 6000 simulações. Partindo de uma atitude qualquer, dentro dos limites impostos durante o aprendizado, a rede consegue levar o satélite para a atitude de referência, que no caso consiste de ângulos e velocidade nulos. A abertura dos painéis em 500 s. provoca uma alteração na atitude, rapidamente compensada pela rede. Comparou-se, a seguir, estes resultados com aqueles gerados por um controle exclusivamente PID, mostrados na Figura 5.52 com ganhos iguais aos utilizados no treinamento da rede: 0,08, 0,05 e 20 para os ganhos proporcional, integrativo e derivativo, respectivamente. O ganho do controle utilizado na composição do sinal de treinamento foi de 0,02. Utilizou-se uma semente aleatória nula durante o treinamento, mas alterou-se para 7 na simulação apresentada nas figuras. Com isso garantiu-se que a simulação fosse efetuada com parâmetros iniciais nos quais ela não foi treinada. Os pesos foram inicializados com de forma aleatória com amplitude

inicial de 0,1. Os parâmetros de convergência do aprendizado foram $\lambda = 0,001$ e $\lambda_m = 0$.

Deve ser salientado, todavia, que os ganhos utilizados no controlador PID não foram otimizados no sentido de obter tempos de resposta mais curtos, ou algum outro critério. Mas assim como é possível que o desempenho do PID possa ser melhorado, também é verdade que a rede tem possibilidades de reduzir ainda mais os erros mostrados na Figura 5.51, desde que o sinal de treinamento seja dirigido ao erro apresentado em sua saída. Na simulação efetuada, o controle sequer chegou a saturar, e, portanto, há ainda um excedente de torque que pode ser utilizado para melhorar o controle efetuado pela rede.

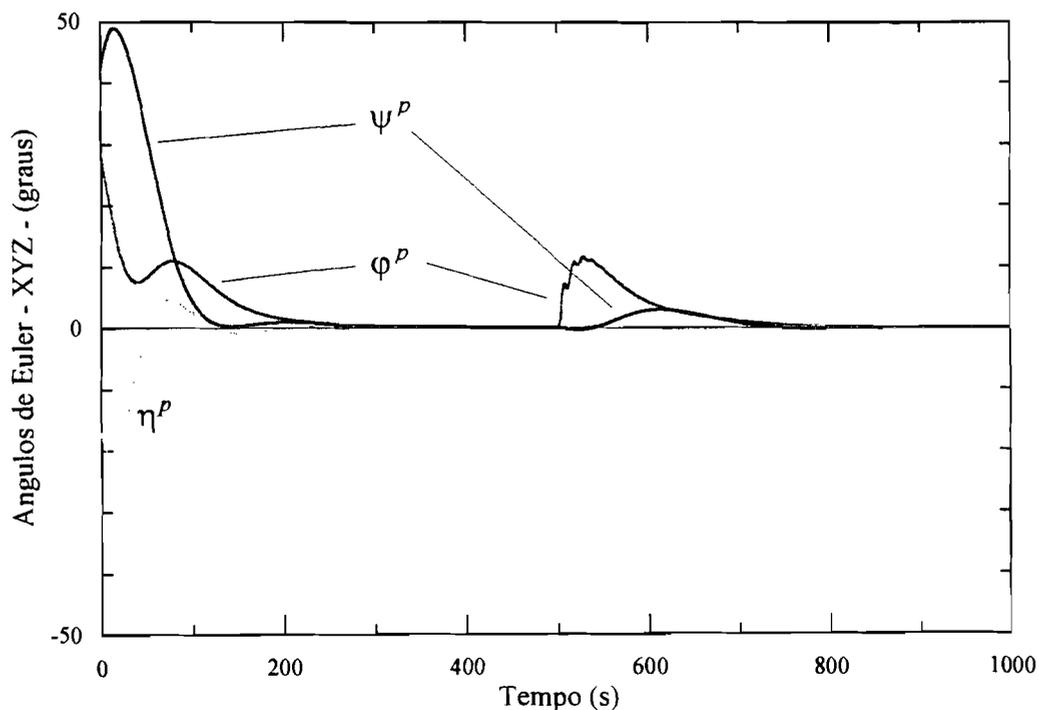


Fig. 5.51 – Controle com realimentação do erro sem supervisão do PID.

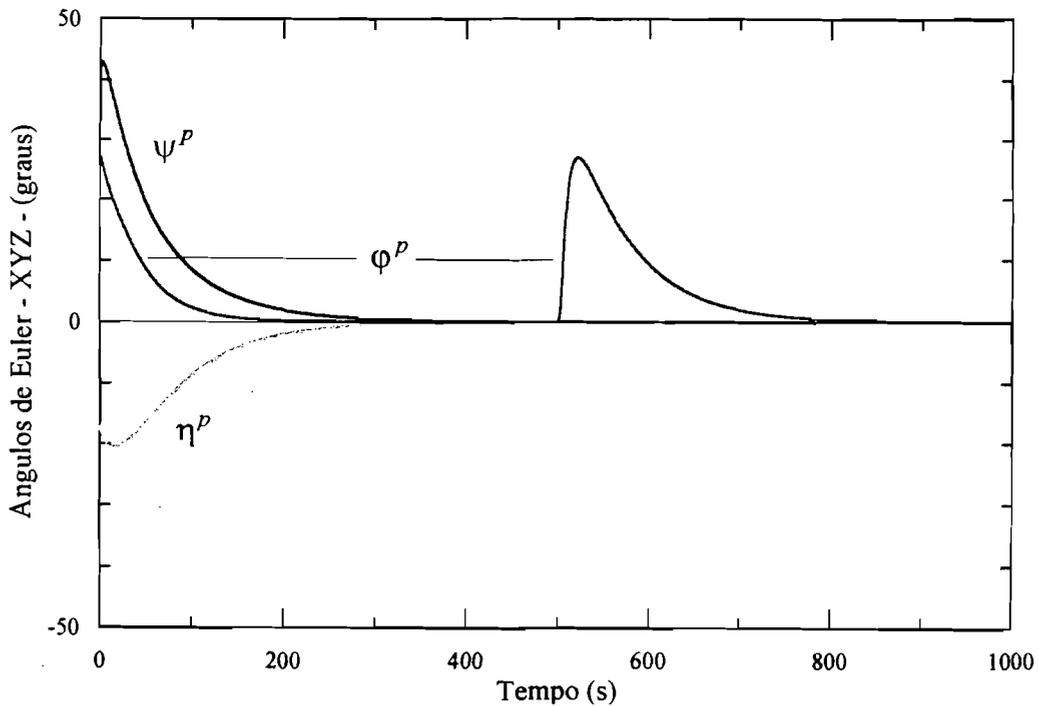


Fig. 5.52 – Controle efetuado com o sinal do PID.

Uma possível solução para corrigir as desvantagens citadas acima seria a utilização do método especializado inverso com sinais de treinamento baseados no erro entre o sistema e a referência. Sugere-se a utilização deste método, cujo esquema é mostrado na Figura 5.53, como desenvolvimentos futuros. Ele é bastante semelhante ao método especializado inverso, mas, ao contrário deste, a região de interesse na qual deve ser efetuado o treinamento fica bastante reduzida (pelo menos no problema da atitude de satélites), já que a solução não mais depende da trajetória de referência adotada. Uma outra vantagem, também devida a este aspecto, é que o erro não precisa estar próximo de zero para que o treinamento aconteça, ao contrário do método especializado inverso, onde é fundamental que a trajetória de referência estivesse próxima do estado do sistema.

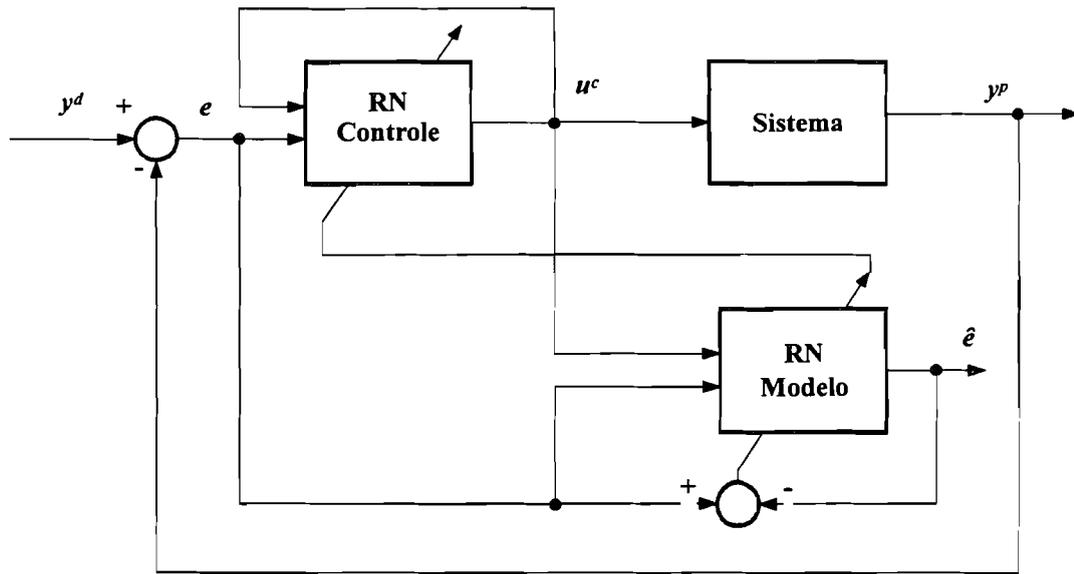


Fig. 5.53 – Controle baseado no método especializado inverso com sinais do erro.

CAPÍTULO 6

COMENTÁRIOS E CONCLUSÕES

Neste trabalho, investigou-se a possibilidade de uso de redes neurais em controle de atitude de satélites. Em sistemas convencionais, este controle exige alto grau de sofisticação face aos requisitos de apontamento e autonomia normalmente impostos aos satélites artificiais. Embora a dinâmica da atitude seja não-linear, os controladores convencionais atualmente empregados não encontram dificuldade pois operam normalmente bastante próximos da referência, onde linearizações são sempre possíveis. Exceções ocorrem quando manobras de atitude envolvendo posições distantes entre si são realizadas, ou quando a própria dinâmica da atitude se altera em virtude de alguma mudança nas características do satélite. Esta última foi o principal motivo de investigação deste trabalho. Considerou-se o caso de um satélite (satélite de sensoriamento remoto da MECB) cujo comportamento dinâmico é alterado em virtude da movimentação de apêndices articulados, ou seja, painéis solares, que modificam os momentos de inércia e posição do centro de massa do satélite durante o movimento.

Em virtude da impossibilidade de treinar a rede neural de controle utilizando o próprio satélite (treinamento *on-line*), foram obtidas relações cinemáticas e dinâmicas para simulação em computador da atitude do satélite antes, durante e após a abertura dos painéis. Mesmo sem ser possível o treinamento local, ainda assim o processo de treinamento pode ser validado, e no futuro todo o treinamento poderá ser efetuado com o satélite já em órbita, ou de forma adaptativa, isto é, com parte do treinamento efetuado em Terra, por simulação, e aperfeiçoado após o lançamento. As relações de movimento foram derivadas da equação de Euler e adaptadas para um corpo que possui apêndices articulados.

As equações da dinâmica aqui obtidas possuem carácter geral, isto é, aplicam-se a qualquer corpo no espaço que possua geometria variável. Embora

encontrem-se, na literatura (Asada e Slotine, 1985) relações gerais para mecanismos articulados, estas formulações partem da premissa que a massa do corpo principal é infinita, e, portanto, não se move. Este, obviamente, não é o caso de satélites artificiais, e tornou-se necessário, portanto, obter uma nova formulação para o movimento do corpo principal do satélite junto com os demais apêndices. Um grande número de satélites artificiais já lançados possuem tais características: satélites de comunicação em órbita geoestacionária com painéis solares e/ou antenas móveis (Brasilsat, Intelsat, etc.); satélites de observação com painéis que acompanham o Sol (Advanced Tiros N, Radarsat, Hubble Space Telescope, etc.).

Tanto a simulação da atitude como também os treinamentos nas redes foram desenvolvidos em MicroSoft FORTRAN Power Station V. 1.0, utilizando precisão dupla. Utilizou-se alocação dinâmica de memória para garantir a flexibilidade dos programas no que tange ao número de neurônios, número de entradas atrasadas e número de parâmetros de entrada e saída. O treinamento foi realizado num computador 486 66 MHz, na identificação do modelo direto e num Pentium 166 MHz na identificação da rede de controle. A rapidez do aprendizado com o método do gradiente possibilita sua operação *on-line* em tempo real. Embora a convergência do método de mínimos quadrados seja bastante superior à do gradiente, o aprendizado em tempo real fica comprometido pela sua baixa velocidade na iteração dos pesos. A otimização do algoritmo, em termos das operações realizadas e da memória consumida pode reverter esta desvantagem, no futuro, principalmente considerando o rápido desenvolvimento de sistemas computacionais mais poderosos e velozes.

Efetou-se o treinamento da rede de identificação do modelo direto com 8 neurônios na camada oculta, e três valores defasados no tempo. As entradas da rede consistiram das leituras dos sensores (magnetômetro e sensor solar), do sinal dos propulsores e do sensor de abertura dos painéis. A rede foi treinada para apresentar na saída o valor dos sensores no instante $t+1$. Verificou-se que, muito embora após o treino o erro fosse relativamente baixo, ainda assim não foi possível obter o modelo inverso a

partir da retro-propagação do erro através do modelo direto. Em virtude deste aspecto, adotou-se uma forma de treinamento que não dependesse da identificação do modelo direto, como por exemplo o método indireto e o método de realimentação do erro. Enquanto que o último mostrou grande tolerância ao desconhecimento do comportamento dinâmico do satélite no início do treinamento, devido à presença do controlador PID, o mesmo não aconteceu com o método indireto. É possível que a grande quantidade de variáveis envolvida torne o treinamento um processo que dependa, para ter sucesso, do ajuste de parâmetros tais como a seleção de pontos especiais para o treino - descartando os demais -, de um controle artificial que impeça o satélite de afastar-se da trajetória de referência durante o início do aprendizado e de um controle suficientemente forte que não sature em torques relativamente baixos, pontos estes não testados neste trabalho.

De certa forma, o método de realimentação do erro compensou em parte estas características negativas do método indireto, e com isso permitiu o aprendizado do modelo inverso, ainda que não dispensasse a presença do controlador PID. Deve ser investigado, no futuro, a possibilidade de utilização do algoritmo dos mínimos quadrados, já que, embora sendo mais rápido do que o gradiente, não convergiu durante o treinamento.

Este trabalho demonstrou que o controle da atitude através de redes neurais tanto em satélites rígidos quanto aqueles providos de apêndices articulados é viável. A rede neural é bastante robusta no que se refere a perturbações no sistema, desconhecimento de parâmetros e ausência de alguns sinais. Foi proposto e mostrado através de testes, também, que o método de treinamento com realimentação do erro (Kawato et al., 1988) pode, em certos casos, dispensar a supervisão do PID no controle do sistema, embora necessite de um ajuste empírico dos seus parâmetros de forma a garantir a convergência do aprendizado. Cabe ainda investigar outras configurações possíveis para o controlador, principalmente se for considerado que a maioria dos satélites necessita da realimentação do erro para efetuar o controle. Uma possível

arquitetura de controlador baseado na realimentação do erro utilizando o método especializado inverso foi sugerida como solução do problema. Uma aplicação típica consiste no controle de atitude de satélites em torno de uma referência fixa, ou então com variação lenta, com período da ordem do período orbital (6000 s). Estas condições são obedecidas por satélites com estabilização geocêntrica, onde um dos eixos aponta para o centro da Terra. Satélites de órbita baixa e geoestacionária, tanto de observação da Terra, quanto meteorológicos, de comunicações e de navegação empregam o apontamento geocêntrico via de regra. A necessidade existe, e, como foi demonstrado, as redes neurais podem ser utilizadas para efetivamente controlar a atitude, bastando simplesmente garantir um pré treinamento num ambiente simulado em computador, seguido de um refinamento (ou treinamento *on-line*) com o satélite já em órbita.

Deve ser realçado que, dentro da pesquisa bibliográfica realizada, não se encontrou sequer um trabalho que efetivamente utilizasse redes neurais para controle de atitude de satélites, embora já se pense em utilizá-las na área espacial (Rios Neto e Rao, 1996; Vadali et al., 1993; Yen, 1995). É claro que até o momento nenhum satélite já lançado possui controle de atitude baseado em redes neurais, mas é perfeitamente possível que no futuro isto venha a acontecer, principalmente se for considerado que somente uma rede neural pode substituir com vantagens os diversos modos de operação normalmente empregados nos satélites. Outro aspecto digno de nota é ausência quase completa, na literatura, de trabalhos sobre controle de sistemas complexos através de redes neurais, com múltiplas entradas e múltiplas saídas, como este abordado aqui. Em razão destes fatos, o presente trabalho teve um carácter exploratório, visando principalmente viabilizar e identificar as principais dificuldades na sua implementação.

REFERÊNCIAS BIBLIOGRÁFICAS

- Adade Filho, A. *Fundamentos de robótica - cinemática, dinâmica e controle de manipuladores robóticos*: Notas de aula dadas no ITA. São José dos Campos, 1992.
- Ahmed, M.S; Tasadduq, I. A. Neural-net controller for nonlinear plants: design approach through linearisation. *IEE Proceedings - Control Theory and Applications*, v. 141, n. 5, p. 315-322, Sep. 1994.
- Anderson, C. W. Learning to control an inverted pendulum using neural networks. *IEEE Control Systems Magazine*, v. 9, n. 3, p.31-36, Apr. 1989.
- Antsaklis, P. J. Neural networks in control systems. *IEEE Control Systems Magazine*, v. 10, n. 3, p. 3-5, Apr. 1990.
- Antsaklis, P. J. Neural networks in control systems. *IEEE Control Systems Magazine*, v. 12, n. 2, p. 8-10, Apr. 1992.
- Asada, H.; Slotine J. J. E. *Robot analysis and control*. New York: John Wiley & Sons, 1985.
- Aström, K. J. Where is the intelligence in intelligent control? *IEEE Control Systems Magazine*, v. 11, n. 1, p. 37-39, Jan. 1991.
- Baffes, P. T.; Shelton, R. O.; Phillips, T. A. *NETS, a neural network development tool*. Huston, Lyndon B. Johnson Space Center, 1991. (JSC-23366)
- Bavarian, B. Introduction to neural networks for intelligent control. *IEEE Control Systems Magazine*, v. 8, n. 2, p.3-7, Apr. 1988.
- Billings, S. A.; Jamaluddin, H. B.; Chen, S. Properties of neural networks with applications to modelling non-linear dynamical systems. *International Journal of Control*. v. 55, n. 1, p. 193-224, 1992.
- Brower, D.; Clemence, G. M. *Methods of celestial mechanics*. New York: Academic, 1961.
- Carrara, V. *Modelagem das forças e torques atuantes em satélites*. São José dos Campos, 132p. (INPE-2454-TDL/094). Dissertação (Mestrado em Ciencia Espacial) - Instituto de Pesquisas Espaciais, 1982.
- Chen, S.; Billings, S. A. Neural networks for nonlinear dynamic system modelling and identification. *International Journal of Control*, v. 56, n. 2, p. 319-346, 1992.

- Chen, S.; Billings, S. A.; Grant, P. M. Non-linear system identification using neural networks. *International Journal of Control*, v. 51, n. 6, p. 1191-1214, June 1990.
- Chen, S.; Billings, S. A.; Grant, P. M. Recursive hybrid algorithm for non-linear system identification using radial basis function networks. *International Journal of Control*, v. 55, n. 5, p. 1051-1070, May 1992.
- Chen, S.; Billings, S. A. Recursive prediction error parameter estimator for non-linear models. *International Journal of Control*, v. 49, n. 2, p. 569-594, Feb. 1989.
- Chen, S.; Cowan, C. F. N.; Billings, S. A.; Grant, P. M. Parallel recursive prediction error algorithm for training layered neural networks. *International Journal of Control*, v. 51, n. 6, p. 1215-1228, June 1990a.
- Chen, S.; Billings, S. A.; Cowan, C. F. N.; Grant, P. M. Practical identification of NARMAX models using radial basis functions. *International Journal of Control*, v. 52, n. 6 1327-1350, Dec. 1990b.
- Chu, S. R.; Shoureshi, R.; Tenorio, M. Neural networks for system identification. *IEEE Control Systems Magazine*, v. 10, n. 3, p. 31-35, Apr. 1990.
- Crandall, S. H.; Kornopp, D. C.; Kurtz Jr, E. F.; Pridmore-Brown, D. C. *Dynamics of mechanical and electromechanical systems*. New York: Mc Graw-Hill, 1968.
- Cybenko, G. Approximation by superposition of a sigmoidal function. *Mathematics of Controls, Signals ans Systems*. v. 2, n. 4, p. 303-314, 1989.
- Demuth, H; Beale, M. *Neural network toolbox user's guide*. Natick, MA: Math Works, 1992.
- Deutsch, R. *Orbital dynamics of space vehicles*. Englewood Cliffs, NJ: Prentice-Hall, 1963.
- Etzebarria, V. Adaptive control of discrete systems using neural networks. *IEE Proceedings - Control Theory and Applications*, v. 141, n. 4, p.209-215, July 1994.
- Fehlberg, E. *Classical fifth -, sixth -, seventh – and eighth – order Runge Kutta formulas with step size control*. Washington, DC: NASA, Oct., 1968. (NASA TR R-287)
- Flandern, T. C.; Pulkkinen, K. F. Low-precision fromulae for planetary positions. *The Astrophysical Journal Supplement Series*. v. 41, n. 3, p. 319-411, Nov. 1979.
- Forsythe, G. E.; Malcolm, M. A.; Moler, C. B. *Computer methods for mathematical computations*. Englewwod Cliffs, NJ: Prentice-Hall, 1977.

- Galvão, F. L. *MECB SSR1 satellite configuration status*. São José dos Campos: Instituto Nacional de Pesquisas Espaciais, 1990. (A-CFG-1000).
- Garcia, C. E.; Morari, M. Internal model control - 1. A unifying review and some new results. *Industrial and Engineering Chemistry. Process Design and Development*. v. 21, n. 2, p. 308-323, 1982.
- Gelb, A. *Applied optimal estimation*. Cambridge, MA: The M.I.T. Press, 1974.
- Hopfield, J. J. Neural networks and physical systems with convergent collective computational abilities. *Proceedings of the National Academy of Sciences*, v. 79, p. 2554-2558, 1982.
- Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, v. 2, n. 5, p. 359-366, 1989.
- Hunt, K. J.; Sbarbaro, D.; Zbikowski, R.; Gawthrop, P. J. Neural networks for control systems - a survey. *Automatica*, v. 28, n. 6, p. 1083-1112, 1992.
- International Geomagnetic Reference Field 1980 (IGRF). *LAGA News*. v. 20, p. 100-102, Dec. 1981.
- Jin, L.; Nikiforuk, P. N.; Gupta, M. M. Adaptive control of discrete-time nonlinear systems using recurrent neural networks. *IEE Proceedings Control Theory Applications*, v. 141, n. 3, p.169-176, May 1994.
- Kawato, M. Computational schemes and neural network models for formation and control of multijoint arm trajectory. In: Miller, W. T. et al. (ed.) *Neural networks for control*. Cambridge: MIT Press, 1990. p. 197-226.
- Kawato, M; Uno, Y.; Isobe, M; Susuki, R. Hierarchical neural network model for voluntary movement with application to robotics. *IEEE Control Systems Magazine*, v. 8, n. 2, p. 8-15, Apr. 1988.
- Kuga, H. K.; Carrara, V.; Medeiros, V. M. *Rotinas auxiliares de mecânica celeste e geração de órbita*. São José dos Campos: INPE, 1981. (INPE-2180/RPE-392).
- Kyung, K. H.; Lee, B. H.; Ko, M. S. Acceleration based learning control of robotic manipulators using a multi-layered neural network. *IEEE Transactions on Systems, Man and Cybernetics*. v. 24, n. 8, p. 1265-1272, Aug. 1994.
- Lightbody, G., Wu, Q. H., Irwin, G.W. Control applications for feed forward networks. In: *Neural networks for control and systems*. London: Peter Peregrinus, 1992. p. 51-71.

- Lopes, R. V. F.; Carrara, V.; Kuga, H. K.; Medeiros, V. M. *Cálculo recursivo do vetor campo geomagnético*. São José dos Campos: INPE, 1983. (INPE-2865-PRE/400).
- Miller III, W. T.; Hewes, R. P.; Glanz, F. H.; Kraft III, L. G. Real-time dynamic control of an industrial manipulator using a neural-network-based learning controller. *IEEE Transactions on Robotics and Automation*, v. 6, n. 1, p.1-9, Feb. 1990.
- Nagata, S.; Sekiguchi, M.; Asakawa, K. Mobile robot control by a structured hierarchical neural network. *IEEE Control Systems Magazine*, v. 10, n. 3, p.69-76, Apr. 1990.
- Naidu, S. R.; Zafiriou, E.; Mcavoy, T. J. Use of neural networks for sensor failure detection in a control system. *IEEE Control Systems Magazine*, v. 10, n. 3, p. 49-55, Apr. 1990.
- Narendra, K. S. Intelligent control. *IEEE Control Systems Magazine*, v. 11, n. 1, p. 39-40, Jan. 1991.
- Narendra, K. S.; Parthasarathy, K. Identification and control for dynamic systems using neural networks. *IEEE Transactions on Neural Networks*, v. 1, n.1, p. 4-27, Mar. 1990.
- Nascimento Jr. C. L. *Artificial neural networks for control and optimization*. Manchester, 209p. Thesis (Doctor of Philosophy in the Faculty of Technology) - University of Manchester, Feb. 1994.
- Nguyen, D. H.; Widrow, B. Neural networks for self-learning control systems. *IEEE Control Systems Magazine*, v. 10, n. 3, p.18-23, Apr. 1990.
- Norgaard, M. *Neural network based system identification toolbox* [online]. Institute of Automation, Denmark, Technical University of Denmark, 1995. [cited 1996]. Technical Report 95-E-773. Available in Post Script from World Wide Web: <<http://www.iau.dtu.dk/Projects/proj/nnsysid.html>>.
- Norgaard, M. *Neural network based control system design toolbox* [online]. Institute of Automation, Denmark, Technical University of Denmark, 1996. [cited 1996]. Technical Report 96-E-830. Available in Post Script from World Wide Web: <<http://www.iau.dtu.dk/Projects/proj/nnctrl.html>>.
- Oliveira, R. C. L.; Yoneyama, T. Controle automático de sistemas lineares empregando modelos de redes neuronais. In: *Brazilian Congress in Neural Nets*, 1., Itajubá, 1994. *Proceedings*. Itajubá: EFEI, 1994. p. 247-255.

- Psaltis, D.; Sideris, A.; Yamamura, A. A. A multilayered network controller. *IEEE Control Systems Magazine*, v. 8, n. 2, p. 17-21, Apr. 1988.
- Rao, K. R.; Kuga, H. K. *Manual de uso de um conjunto de integradores numéricos para problemas de condições iniciais*. São José dos Campos: Instituto de Pesquisas Espaciais, 1986. (INPE-3830-RPI/154).
- Rios Neto, A. Stochastic parameter estimation neural nets supervised learning approach. In: Brazilian Congress in Neural Nets, 1., Itajubá, 1994. *Proceedings*, Itajubá: EFEI, 1994. p. 62-65.
- Rios Neto, A.; Rao, K. R. A study on the on board artificial satellite orbit propagations using artificial neural networks. In: International Astrodynamics Symposium, 11, Gifu, Japan, 1996. *Proceedings*, p. 192-195.
- Rosenblatt, F. *Principles of neurodynamics*. Washington, DC: Spartan Press, 1961.
- Rumelhart, D. E.; Hilton, G. E.; Williams, R. J. Learning internal representations by error propagation. In: Rumelhart, D. E.; McClelland, J. L. (ed). *Parallel data processing*. Cambridge, MA: M.I.T. Press, 1986. v. 1, cap. 8.
- Sartori, M. A.; Antsaklis, P. J. Implementations of learning control systems using neural networks. *IEEE Control Systems Magazine*, v. 12, n. 2, p. 49-57, Apr. 1992.
- Shoureshi, R. Learning and decision making for intelligent control systems. *IEEE Control Systems Magazine*, v. 11, n. 1, p. 34-37, Jan. 1991.
- Vadali, S. R.; Krishnan, S.; Singh, T. Attitude control of spacecraft using neural networks. In: *Advances in the astronautical sciences*. San Diego, CA: American Astronautical Society. v. 82, part 1, p. 271-285, 1993.
- Wang, H.; Brown, M.; Harris, C. I. Neural networks modelling of unknown nonlinear systems subject to immeasurable disturbances. *IEEE Proceedings on Control Theory Applications*, v. 141, n. 4, p. 216-222, July 1994.
- Warwick, K.; Irwin, G. W.; Hunt, K.J. *Neural networks for control and systems*. London: Peter Peregrinus, 1992.
- Werbos, P. J. An overview of neural networks for control. *IEEE Control Systems Magazine*, v. 11, n. 1, p. 40-41, Jan. 1991.
- Wertz, J. R. *Spacecraft attitude determination and control*, London: D. Reidel, 1978 (Astrophysics and Space Science Library).
- Widrow, B.; Sterns, S. D. *Adaptive signal processing*. New York: Prentice-Hall, 1985.

Yen, G. G. Autonomous neural control in flexible space structures. *Control Eng. Practice*. v. 3, n. 4, p.471-483, 1995

Żbikowski, R.; Gawthrop, P. J. A survey of neural networks for control. In: *Neural networks for control and systems*. London: Peter Peregrinus, 1992. p. 31-50.

Zhang, Y.; Prathyush, S.; Hearn, G. E. An on-line trained adaptive neural controller. *IEEE Control Systems Magazine*, v. 15, n. 5, p.67-75, Oct. 1995.

APÊNDICE AREPRESENTAÇÃO DE VETORES EM SISTEMAS NÃO INERCIAISA.1 TRANSFORMAÇÕES DE SISTEMAS DE COORDENADAS

A transformação de coordenadas entre dois sistemas quaisquer, envolve rotações em torno de até três direções distintas, de forma a fazer coincidir os dois sistemas de eixos, e uma translação de forma a coincidir ambas as origens. O produto das rotações é representado pela *matriz de rotação*, uma matriz de dimensão 3 por 3. Considere então um vetor qualquer, r^b , cujas componentes estão representadas no sistema de coordenadas b . Considere também um outro sistema de coordenadas a , como representado na Figura A.1. As componentes de r^b , quando expresso no sistema a , serão dadas por:

$$r^a = A_{b,a} r^b + a_{ab}^a \quad (\text{A.1})$$

onde $A_{b,a}$ é a matriz de rotação que leva o sistema b ao sistema a , e a_{ab}^a é a posição da origem do sistema b com relação ao sistema a , expressa neste mesmo sistema. As componentes da matriz $A_{b,a}$, assim definida, são dadas pelas direções dos versores dos eixos x_b, y_b, z_b em relação ao sistema a , na forma:

$$A_{b,a} = [x_b \quad y_b \quad z_b] . \quad (\text{A.2})$$

Em outras palavras, uma mudança de coordenadas consiste numa rotação efetuada pela matriz $A_{b,a}$ e numa translação realizada pelo vetor a_{ab}^a .

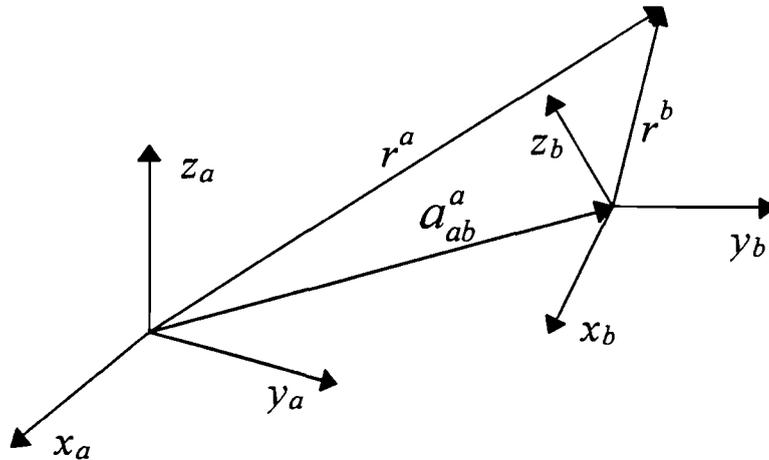


Fig. A.1 - Transformação de coordenadas.

A.2 SISTEMAS GIRANTES

Caso o sistema b seja dotado de velocidade angular relativa ao sistema de coordenadas a , então um vetor fixo ao primeiro irá descrever um movimento giratório quando observado a partir do segundo. Para obter-se a correlação entre as componentes deste vetor, considere-se um sistema de coordenadas x_b, y_b, z_b , fixo a um corpo com velocidade angular $\omega_{b/a}$ com relação a um outro sistema de referência x_a, y_a, z_a , conforme indicado na Figura A.2. Num instante t qualquer, um vetor inicialmente representado no sistema b pode ser expresso no sistema a através da matriz de rotação que correlaciona os dois sistemas:

$$r^a = A_{b,a}(t) r^b \quad (\text{A.3})$$

onde o índice superior representa o sistema no qual o vetor é expresso. Será suposto, a partir daqui, que os sistemas a e b possuem origens coincidentes (embora na Figura A.2 estejam representados com origens distintas, para facilitar a visualização). Na verdade, é sempre possível separar a transformação de coordenadas numa rotação e numa translação puras. O efeito da translação, portanto, não será demonstrado aqui.

A velocidade angular relativa entre os sistemas, $\omega_{b/a}$, pode também ser expressa no sistema a :

$$\omega_{b/a}^a = A_{b,a} \omega_{b/a}^b. \quad (\text{A.4})$$

A dependência da matriz de rotação com o tempo foi suprimida da notação, embora esteja subentendido que tanto a matriz quanto os vetores são funções do tempo.

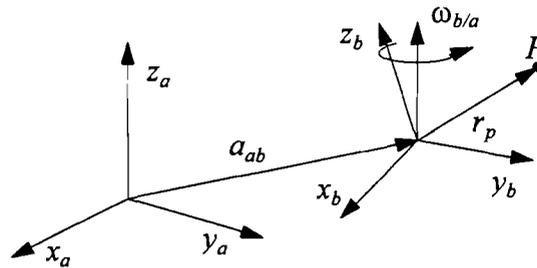


Fig. A.2 - Sistema de coordenadas girante.

Seja agora um vetor r_p que posiciona um ponto P referido ao sistema de coordenadas girante b . A velocidade deste ponto com relação ao sistema de referência a é obtida pela derivada do vetor r_p expresso neste sistema (considerando coincidente a origem de ambos os sistemas):

$$\frac{dr_p^a}{dt} = \frac{dA_{b,a}}{dt} r_p^b + A_{b,a} \frac{dr_p^b}{dt}. \quad (\text{A.5})$$

A derivada da matriz de rotação é bem conhecida e leva ao resultado (Wertz, 1978; Crandall et al., 1968):

$$\dot{A}_{b,a} = \omega_{b/a}^a \times A_{b/a} \quad (\text{A.6})$$

onde a operação indica o produto vetorial da velocidade angular pelo vetor que estiver sendo transformado pela matriz. Será conveniente fazer uso da matriz do produto vetorial, conforme definida em Wertz, 1978:

$$\dot{A}_{b,a} = \Omega(\omega_{b/a}^a) A_{b/a} \quad (\text{A.7})$$

sendo que $\Omega(\cdot)$ é a matriz de produto vetorial, dada por:

$$\Omega(\omega) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (\text{A.8})$$

onde ω_x , ω_y , ω_z são as componentes do vetor ω . A matriz de produto vetorial é uma função ímpar, já que $\Omega(-\omega) = -\Omega(\omega)$. As vantagens de utilizá-la decorrem principalmente da facilidade de operações computacionais com matrizes, facilidade de notação e possibilidade de utilização das propriedades de matrizes. Tanto quanto o produto vetorial, a mudança da ordem dos operandos da matriz Ω inverte o sinal do resultado:

$$\Omega(\omega) r = -\Omega(r) \omega \quad (\text{A.9})$$

Substituindo a derivada da matriz de rotação na Relação A.5, resulta:

$$\dot{r}_{p/a}^a = \Omega(\omega_{b/a}^a) r_p^a + A_{b,a} \dot{r}_{p/b}^b \quad (\text{A.10})$$

Este resultado relaciona a velocidade de um ponto P com relação a dois sistemas de referência. A velocidade de P com relação ao sistema a é igual à velocidade de arrastamento do ponto por se encontrar num sistema girante com velocidade angular $\omega_{b/a}$, adicionada da velocidade de P em relação ao sistema b . Veja que a segunda parcela

do segundo termo é igual à velocidade do ponto com relação ao sistema b , porém expressa no sistema a :

$$\dot{r}_{p/a}^a = \Omega(\omega_{b/a}^a) r_p^a + \dot{r}_{p/b}^a \quad (\text{A.11})$$

Substituindo a Relação A.3 nesta última, chega-se à:

$$\dot{r}_{p/b}^b = -A_{b,a}^T \Omega(A_{b,a} \omega_{b/a}^b) A_{b,a} r_p^b + \dot{r}_{p/a}^b \quad (\text{A.12})$$

Para obter os resultados acima, não foi imposta nenhuma condição para os sistemas de coordenadas (como a necessidade do sistema a ser inercial, por exemplo), de sorte que ambos os sistemas são arbitrários. As equações continuam válidas, portanto, ainda que os sistemas sejam trocados entre si, ou seja:

$$\dot{r}_{p/b}^b = \Omega(\omega_{a/b}^b) r_p^b + \dot{r}_{p/a}^b \quad (\text{A.13})$$

Da simples comparação entre os dois últimos resultados, e lembrando que $\omega_{a/b} = -\omega_{b/a}$, conclui-se que:

$$\Omega(A_{b,a} \omega_{b/a}^b) = A_{b,a} \Omega(\omega_{b/a}^b) A_{b,a}^T, \quad (\text{A.14})$$

ou seja, uma transformação de coordenadas no argumento de uma matriz de produto vetorial é igual ao produto da matriz de produto vetorial com o argumento não transformado pelo produto da inversa da matriz de rotação, multiplicado à esquerda pela matriz de rotação.

A aceleração do ponto P relativa ao sistema de coordenadas a é obtida derivando-se novamente a Relação A.10, considerando também A.7 e A.14 e resulta:

$$\ddot{r}_{p/a}^a = \Omega(\omega_{b/a}^a) \Omega(\omega_{b/a}^a) r_p^a + 2 \Omega(\omega_{b/a}^a) \dot{r}_{p/b}^a - \Omega(r_p^a) A_{b,a} \dot{\omega}_{b/a/b}^b + \ddot{r}_{p/b}^a. \quad (\text{A.15})$$

Os termos desta equação refletem, respectivamente, a aceleração centrípeta, a aceleração de Coriolis, a aceleração devida às alterações na velocidade angular do sistema girante, e a aceleração do ponto P relativa à este último sistema. Note que a derivada da velocidade angular do sistema b com relação ao sistema a , $\dot{\omega}_{b/a/b}^b$, é tomada no referencial girante b , porém transformada para o sistema a , após a pré multiplicação pela matriz de rotação $A_{b,a}$. Em outras palavras, esta aceleração angular reflete as alterações na velocidade angular quando observada a partir do sistema girante. Na verdade, a derivada da velocidade angular é obtida derivando-se a Equação A.4, que resulta:

$$\dot{\omega}_{b/a/a}^a = \Omega(\omega_{b/a}^a) \omega_{b/a}^a + \dot{\omega}_{b/a/b}^a. \quad (\text{A.16})$$

Mas como o produto vetorial é nulo (os vetores são paralelos), a derivada é a mesma, quer tomada com relação ao sistema a , ou com relação ao sistema girante b .

De maneira similar à realizada na obtenção da velocidade do ponto P , também a expressão da aceleração não se altera quando representada no sistema de coordenadas girante, exceto pela derivada da velocidade angular, que agora é expressa neste mesmo sistema:

$$\ddot{r}_{p/a}^b = \Omega(\omega_{b/a}^b) \Omega(\omega_{b/a}^b) r_p^b + 2 \Omega(\omega_{b/a}^b) \dot{r}_{p/b}^b - \Omega(r_p^b) \dot{\omega}_{b/a}^b + \ddot{r}_{p/b}^b. \quad (\text{A.17})$$

Se o ponto P estiver fixo com relação ao sistema b , então tanto a velocidade quanto a aceleração do ponto serão nulas quando tomadas em relação ao sistema girante. Neste caso, as Expressões A.13 e A.17, da velocidade e aceleração no sistema b serão dados respectivamente por:

$$\dot{r}_{p/a}^b = \Omega(\omega_{b/a}^b) r_p^b \quad (\text{A.18})$$

$$\ddot{r}_{p/a}^b = \Omega(\omega_{b/a}^b) \Omega(\omega_{b/a}^b) r_p^b - \Omega(r_p^b) \dot{\omega}_{b/a/b}^b. \quad (\text{A.19})$$

Estas equações, juntamente com as que serão obtidas a seguir, irão fornecer a base para a formulação das equações dinâmicas do movimento, para um satélite composto por apêndices articulados. Como a atitude do satélite é medida no referencial do próprio corpo do satélite, é natural que se deseje obter as velocidades e acelerações dos vetores neste mesmo referencial, que por definição é o sistema girante b . A Seção seguinte irá obter portanto estas mesmas equações no caso do ponto P situar-se em um destes apêndices.

A.3 SISTEMAS COMPOSTOS POR DOIS SISTEMAS GIRANTES.

Considere novamente os sistemas a e b , com este último girando com velocidade angular $\omega_{b/a}$ com relação ao primeiro. Considere também um outro sistema c , conforme a Figura A.3, movido de velocidade $\omega_{c/b}$ com relação ao sistema b , e no qual é conhecida a posição r_p , velocidade $\dot{r}_{p/c}$, e aceleração $\ddot{r}_{p/c}$ de um ponto P .

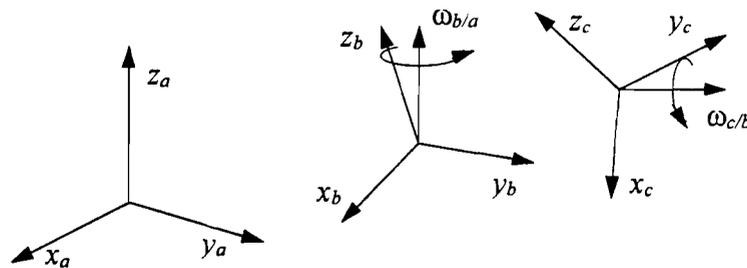


Fig. A.3 - Sistema de coordenadas c fixado nos apêndices do satélite.

Se $A_{c,b}$ for a matriz de rotação que leva um vetor representado no sistema de coordenadas c ao sistema b , então a transformação de um vetor no sistema c para o sistema a será dada por:

$$r^a = A_{b,a} A_{c,b} r^c. \quad (\text{A.20})$$

A velocidade do ponto P com relação ao sistema a é obtida derivando-se a expressão acima:

$$\dot{r}_{p/a}^a = \Omega(\omega_{b/a}^a) A_{b,a} A_{c,b} r_p^c + A_{b,a} \Omega(\omega_{c/b}^b) A_{c,b} r_p^c + A_{b,a} A_{c,b} \dot{r}_{p/c}^c \quad (\text{A.21})$$

que resulta, após efetuar uma transformação para representar a velocidade angular $\omega_{c/b}$ no sistema a (e utilizando a propriedade relacionada na Equação A.14):

$$\dot{r}_{p/a}^a = \Omega(\omega_{b/a}^a + \omega_{c/b}^a) r_p^a + A_{b,a} A_{c,b} \dot{r}_{p/c}^c. \quad (\text{A.22})$$

Se esta velocidade for representada no sistema b (um dos sistemas girantes), então a equação acima fica reduzida à:

$$\dot{r}_{p/a}^b = \Omega(\omega_{b/a}^b + \omega_{c/b}^b) r_p^b + A_{c,b} \dot{r}_{p/c}^c. \quad (\text{A.23})$$

A aceleração do ponto P relativo ao sistema a é obtida com a derivada da Relação A.22, que resulta, já efetuada a transformação para o sistema b :

$$\begin{aligned} \ddot{r}_{p/a}^b = & \Omega(\omega_{b/a}^b + \omega_{c/b}^b) \Omega(\omega_{b/a}^b + \omega_{c/b}^b) r_p^b + 2 \Omega(\omega_{b/a}^b + \omega_{c/b}^b) \dot{r}_{p/c}^b - \\ & - \Omega(r_p^b) \Omega(\omega_{b/a}^b) \omega_{c/b}^b - \Omega(r_p^b) (\dot{\omega}_{b/a/b}^b + \dot{\omega}_{c/b/c}^b) + A_{c,b} \ddot{r}_{p/c}^c. \end{aligned} \quad (\text{A.24})$$

A diferença entre esta equação e a Relação A.15, é a introdução do terceiro termo do segundo membro, que representa a aceleração centrípeta do ponto P com relação ao sistema intermediário b . Caso este ponto esteja fixo no sistema c , então sua

velocidade e aceleração relativas a este sistema serão nulas, e as duas últimas relações resultam, respectivamente:

$$\dot{r}_{p/a}^b = \Omega(\omega_{b/a}^b + \omega_{c/b}^b) r_p^b, \quad (\text{A.25})$$

$$\begin{aligned} \ddot{r}_{p/a}^b = & \Omega(\omega_{b/a}^b + \omega_{c/b}^b) \Omega(\omega_{b/a}^b + \omega_{c/b}^b) r_p^b - \Omega(r_p^b) \Omega(\omega_{b/a}^b) \omega_{c/b}^b - \\ & - \Omega(r_p^b) (\dot{\omega}_{b/a/b}^b + \dot{\omega}_{c/b/c}^b). \end{aligned} \quad (\text{A.26})$$

Estas equações irão permitir e facilitar a formulação das equações dinâmicas de um corpo com apêndices articulados. Nesta formulação, o sistema de coordenadas a passará a representar o sistema inercial i ; o sistema fixo ao corpo do satélite (denominado de sistema o , em torno do qual a atitude é conhecida) será representado pelo sistema intermediário b ; e, finalmente, o sistema de coordenadas fixo ao apêndice (de número k) será equivalente ao sistema c .

APÊNDICE BMATRIZ DE INÉRCIA DE CORPOS RÍGIDOSB.1 MATRIZ DE INÉRCIA

A inércia ou matriz de inércia de um corpo de dimensões finitas com relação à um referencial dado a é dada por (Crandall et al., 1968):

$$I^a = \int_V \left[r^{aT} r^a \underline{1} - r^a r^{aT} \right] dm \quad (\text{B.1})$$

onde r^a é o vetor posição do elemento de massa dm no sistema a e $\underline{1}$ é a matriz identidade. A matriz de inércia deve ser integrada em todo o volume V do corpo. O sobrescrito T representa o vetor ou matriz transposta. Pode ser mostrado, sem grande dificuldade, que a equação acima pode ser também dada por:

$$I^a = - \int_V \Omega(r^a) \Omega(r^a) dm \quad (\text{B.2})$$

sendo $\Omega(\cdot)$ o operador que efetua o produto vetorial, conforme descrito no Apêndice A.

Se o corpo possuir movimento com relação ao sistema de coordenadas a , então a posição r^a e, conseqüentemente, a matriz de inércia I^a irão variar com o tempo, tornando impraticável a integração. Se, entretanto, for efetuada uma mudança no sistema de coordenadas, de tal forma que o novo sistema esteja fixo ao corpo e com origem no seu centro de massa, então a descrição da matriz de inércia se simplifica, e, se o corpo for rígido, esta se torna constante.

Considere novamente os sistemas de coordenadas a e b conforme a Figura A.1, do Apêndice A. Considere também que o sistema b está fixo a um corpo de massa

m_b e matriz de inércia I^b com relação ao sistema b . Finalmente, admita que a origem deste sistema coincide com o centro de massa do corpo, e portanto

$$r_{cm}^b = \int r^b dm = 0 \quad (\text{B.3})$$

onde r_{cm} é a posição do centro de massa do corpo e r o vetor posição do elemento de massa dm referidos ao sistema b . Usando as propriedades das matrizes de rotação, a Relação A.1 que leva o sistema b para o sistema a e considerando a Expressão A.14 que fornece a transformação de coordenadas na matriz de produto vetorial (ambas relacionadas no Apêndice A), obtém-se que o momento de inércia com relação ao sistema a , com origem fora do centro de massa, vale:

$$I^a = A_{b,a} I^b A_{b,a}^T + m_b (a_{ab}^T a_{ab} \mathbf{1} - a_{ab} a_{ab}^T) \quad (\text{B.4})$$

com I^a sendo o momento de inércia do corpo com relação ao sistema a , e a_{ab} é o vetor posição da origem do sistema b em relação ao sistema a . Esta expressão permite que, conhecendo-se a matriz de inércia de um corpo com relação a um sistema com origem no centro de massa, possa ser obtida a matriz de inércia deste corpo com relação a qualquer outro sistema de coordenadas.

APÊNDICE CMODELO DO SATÉLITE SSRC.1 MATRIZES DE INÉRCIA DO SATÉLITE SSR

O satélite SSR é composto por um corpo principal e três painéis solares, como indicado na Figura C.1. Durante o lançamento, os painéis permanecem junto ao corpo principal, e são abertos após a injeção em órbita. O documento de configuração (*lay-out*) deste satélite, A-CFG-1000 (Galvão, 1990) fornece as dimensões, massas e inércias do satélite com painéis fechados e abertos. As dimensões dos três painéis são iguais, como também são os mecanismos de abertura. Ainda assim, existe uma pequena diferença nos momentos de inércia do painel superior (número 3). A geometria do satélite é descrita no sistema de referência geométrico (x_b, y_b e z_b), com origem no cone de adaptação com o lançador, por motivos construtivos. Os momentos de inércia do satélite referem-se, entretanto, ao sistema de coordenadas do centro de massa do conjunto (x_i, y_i e z_i), ajustado para coincidir com o eixo de simetria do corpo principal, conforme indica a Figura C.2, centro este situado a 579 mm da origem do sistema b , ao longo do eixo z_b . Ambos os sistemas possuem a mesma orientação, sendo que x, y e z correspondem, respectivamente, aos eixos de rolamento, guinada e arfagem (o eixo y aponta para o centro da Terra, na configuração de vôo). As matrizes de inércia I_1, I_2 e I_3 dos painéis, e I_c do conjunto todo com os painéis na posição fechada (de lançamento) com relação ao sistema de eixos i é fornecida no documento acima citado:

$$I_c = \begin{bmatrix} 23.5 & 0.1 & 0.1 \\ 0.1 & 23.2 & 0 \\ 0.1 & 0 & 10.6 \end{bmatrix} \text{ (kg. m}^2\text{)} \quad \text{(C.1)}$$

$$I_1 = \begin{bmatrix} 0.385 & 0 & 0 \\ 0 & 0.585 & 0 \\ 0 & 0 & 0.385 \end{bmatrix} \text{ (kg. m}^2\text{)} \quad (\text{C.2})$$

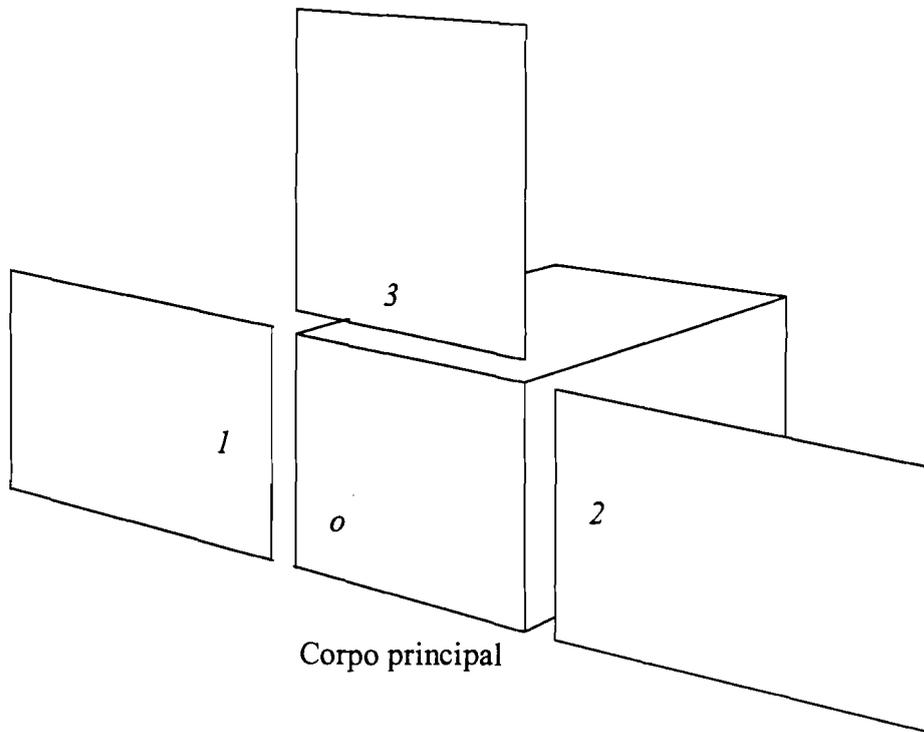


Fig. C.1 - Configuração do satélite SSR com painéis abertos.

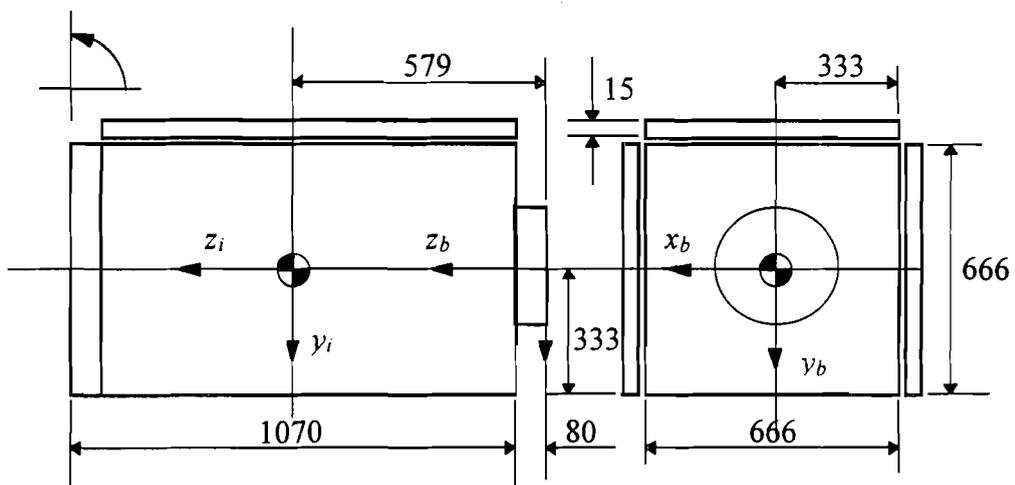


Fig. C.2 - Dimensões do satélite SSR na configuração de lançamento

$$I_2 = \begin{bmatrix} 0.385 & 0 & 0 \\ 0 & 0.585 & 0 \\ 0 & 0 & 0.385 \end{bmatrix} \text{ (kg. m}^2\text{)} \quad (\text{C.3})$$

$$I_3 = \begin{bmatrix} 0.585 & 0 & 0 \\ 0 & 0.398 & 0 \\ 0 & 0 & 0.3976 \end{bmatrix} \text{ (kg. m}^2\text{)}. \quad (\text{C.4})$$

A posição do centro de massa de cada um dos painéis, também na posição de lançamento, e com relação à origem do sistema b é dada por:

$$r_{cm1}^b = \begin{bmatrix} 0.342 \\ 0 \\ 0.546 \end{bmatrix} \text{ (m)} \quad (\text{C.5})$$

$$r_{cm2}^b = \begin{bmatrix} -0.342 \\ 0 \\ 0.546 \end{bmatrix} \text{ (m)} \quad (\text{C.6})$$

$$r_{cm3}^b = \begin{bmatrix} 0 \\ -0.342 \\ 0.546 \end{bmatrix} \text{ (m)}. \quad (\text{C.7})$$

A massa total do satélite considerada é $M = 200$ kg, sendo que cada um dos painéis possui massa $m_k = 2.5$ kg ($k = 1, 2, 3$). A massa do corpo principal, m_o , é portanto de 192.5 kg. Com estas informações pode-se agora obter a posição do centro de massa do corpo principal, r_{cmo} , em relação ao sistema b , o que resulta:

$$r_{cmo}^b = \begin{bmatrix} 0 \\ 0.0044 \\ 0.5803 \end{bmatrix} \text{ (m)}. \quad (\text{C.8})$$

Como as matrizes de inércia das partes constituintes são todas conhecidas em relação ao centro de massa do conjunto, para obtê-las com relação ao seu próprio centro de massa deve-se primeiro obter as posições destes centros em relação ao centro de massa do satélite, o que resulta:

$$r_{cm0}^i = \begin{bmatrix} 0. \\ 0.0044 \\ 0.0013 \end{bmatrix} \text{ (m)}, \quad (\text{C.9})$$

$$r_{cm1}^i = \begin{bmatrix} 0.342 \\ 0 \\ -0.033 \end{bmatrix} \text{ (m)}, \quad (\text{C.10})$$

$$r_{cm2}^i = \begin{bmatrix} -0.342 \\ 0 \\ -0.033 \end{bmatrix} \text{ (m)}, \quad (\text{C.11})$$

$$r_{cm3}^i = \begin{bmatrix} 0 \\ -0.342 \\ -0.033 \end{bmatrix} \text{ (m)}. \quad (\text{C.12})$$

Com isso, pode-se agora utilizar a Relação B.4 (Apêndice B), e obter as matrizes de inércia de cada corpo com relação ao seu próprio centro de massa, o que será realizado individualmente nas próximas seções.

C.2 POSIÇÃO DAS JUNTAS

O próximo passo consiste na obtenção dos vetores a_{ok} e a_{ko} que posicionam as juntas articuladas com relação ao sistema de coordenadas fixo ao centro de massa do

corpo principal e do apêndice k , respectivamente, bem como as matrizes de rotação A_{ok} e as matrizes de inércia.

C.2.1 Corpo Principal

Considere os sistemas de coordenadas intermediários conforme a Figura C.3. Os painéis descrevem um ângulo de abertura de 90 graus, desde a posição fechada até a posição aberta. Um mecanismo de trava irá prender os painéis quando estes estiverem na posição aberta. Os eixos de rotação das juntas estão afastados de 50 mm da borda da face superior do satélite, como mostrado na figura, e em cada um deles será fixado um sistema de coordenadas intermediário, de forma a possibilitar a obtenção das matrizes de rotação.

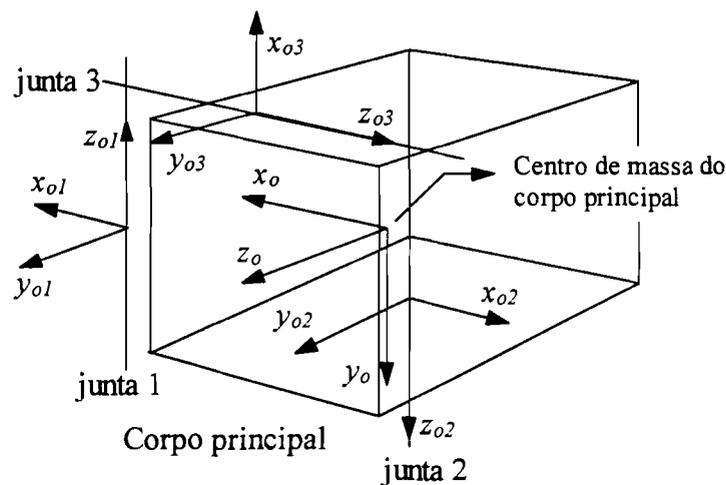


Fig. C.3 - Sistema de coordenadas do corpo principal do satélite.

Para o corpo principal do satélite, os vetores que posicionam as origens dos sistemas de referência das juntas com relação ao sistema de coordenadas o valem:

$$a_{o1}^o = \begin{bmatrix} 0.383 \\ -0.0044 \\ 0.5697 \end{bmatrix} \text{ (m)}, \quad (\text{C.13})$$

$$a_{o2}^o = \begin{bmatrix} -0.383 \\ -0.0044 \\ 0.5697 \end{bmatrix} \text{ (m)}, \quad (\text{C.14})$$

$$a_{o3}^o = \begin{bmatrix} 0 \\ -0.3874 \\ 0.5697 \end{bmatrix} \text{ (m)}. \quad (\text{C.15})$$

A matriz de inércia do corpo principal neste sistema é dada por:

$$I_o = \begin{bmatrix} 22.14 & 0.1 & 0.1 \\ 0.1 & 21.63 & 0 \\ 0.1 & 0 & 9.43 \end{bmatrix} \text{ (kg. m}^2\text{)}, \quad (\text{C.16})$$

e a massa do corpo principal vale:

$$m_o = 192.5 \text{ kg}. \quad (\text{C.17})$$

Os vetores de velocidade angular nas juntas, representados no sistema fixo ao centro de massa do corpo principal são fornecidos por:

$$\omega_1^o = \begin{bmatrix} 0 \\ -\dot{\theta}_1 \\ 0 \end{bmatrix} \text{ (rd/s)}, \quad (\text{C.18})$$

$$\omega_2^o = \begin{bmatrix} 0 \\ \dot{\theta}_2 \\ 0 \end{bmatrix} \text{ (rd/s)}, \quad (\text{C.19})$$

$$\omega_3^o = \begin{bmatrix} -\dot{\theta}_3 \\ 0 \\ 0 \end{bmatrix} \text{ (rd/s).} \quad (\text{C.20})$$

onde θ_1 , θ_2 e θ_3 são os ângulos nas articulações, e serão obtidos a partir da análise da dinâmica de abertura dos painéis.

C.2.2 Painel Solar 1 (Frontal)

Será também fixado um sistema de coordenadas intermediário na junta 1, de tal forma que torne possível a correspondência entre o sistema de eixos do corpo principal e o sistema de eixos deste painel, conforme indicado na Figura C.4. Com relação ao sistema x_1, y_1 e z_1 , a matriz de inércia e a posição da junta do painel são dados por:

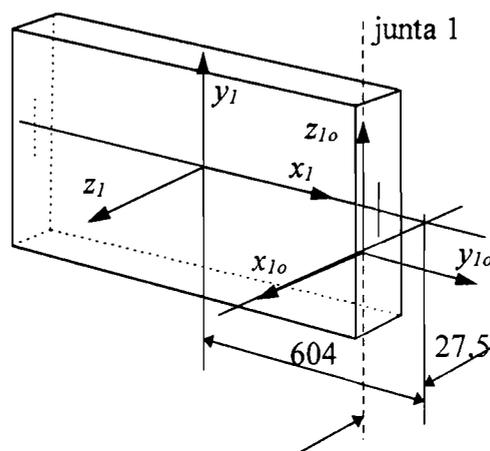


Fig. C.4 - Sistemas de coordenadas do painel frontal.

$$I_1 = \begin{bmatrix} 0.092 & 0 & 0 \\ 0 & 0.290 & 0 \\ 0 & 0 & 0.382 \end{bmatrix} \text{ (kg. m}^2\text{),} \quad (\text{C.21})$$

$$a_{1o}^1 = \begin{bmatrix} 0.604 \\ 0 \\ 0.0275 \end{bmatrix} \text{ (m)}, \quad (\text{C.22})$$

A massa m_1 do painel, e a matriz de rotação $A_{1,o}$ que relaciona o sistema do elo 1 ao sistema do corpo principal resultam:

$$m_1 = 2.5 \text{ kg}, \quad (\text{C.23})$$

$$A_{1,o} = \begin{bmatrix} -\sin\theta_1 & 0 & \cos\theta_1 \\ 0 & -1 & 0 \\ \cos\theta_1 & 0 & \sin\theta_1 \end{bmatrix}, \quad (\text{C.24})$$

onde θ_1 é nulo quando o painel se encontra na posição de lançamento, subindo até 90° quando estiver completamente aberto.

C.2.3 Painel Solar 2 (Traseiro)

Para o painel 2, cujo sistema de coordenadas localizado em seu centro de massa é mostrado na Figura C.5, a matriz de inércia I_2 , o vetor de posição da junta a_{2o} , a massa m_2 e matriz de rotação $A_{2,o}$ até o sistema do corpo principal assumem os valores:

$$I_2 = \begin{bmatrix} 0.092 & 0 & 0 \\ 0 & 0.290 & 0 \\ 0 & 0 & 0.382 \end{bmatrix} \text{ (kg. m}^2\text{)}, \quad (\text{C.25})$$

$$a_{2o}^2 = \begin{bmatrix} 0.604 \\ 0 \\ 0.0275 \end{bmatrix} \text{ (m)}, \quad (\text{C.26})$$

$$m_2 = 2.5 \text{ kg}, \quad (\text{C.27})$$

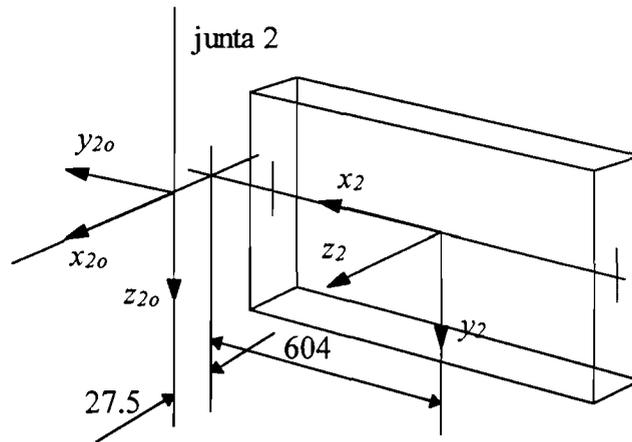


Fig. C.5 - Sistemas de coordenadas do painel traseiro.

$$A_{2,o} = \begin{bmatrix} \sin\theta_2 & 0 & -\cos\theta_2 \\ 0 & 1 & 0 \\ \cos\theta_2 & 0 & \sin\theta_2 \end{bmatrix}, \quad (\text{C.28})$$

onde θ_2 é o ângulo de posição da junta 2, variando de 0 a 90°.

C.2.4 Painel Solar 3 (Superior)

Também no painel 3 foi fixado um sistema de coordenadas intermediário, x_{3o} , y_{3o} e z_{3o} que permite a obtenção da matriz de rotação $A_{3,o}$. Na Figura C.6 estão indicados os sistemas de referência utilizados, e a matriz de inércia I_3 , o vetor de posição da junta a_{3o} , a massa m_3 e matriz de rotação $A_{3,o}$ até o sistema do corpo principal resultam:

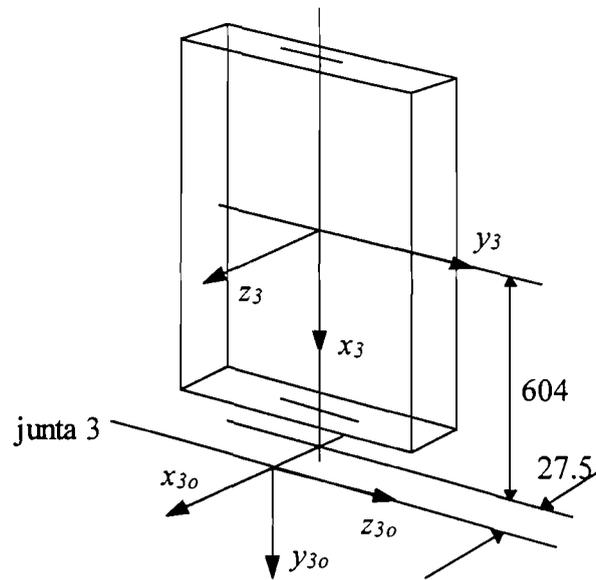


Fig. C.6 - Sistemas de coordenadas do painel superior.

$$I_3 = \begin{bmatrix} 0.105 & 0 & 0 \\ 0 & 0.290 & 0 \\ 0 & 0 & 0.395 \end{bmatrix} \text{ (kg. m}^2\text{)}, \quad (\text{C.29})$$

$$a_{3o}^3 = \begin{bmatrix} 0.604 \\ 0 \\ 0.0275 \end{bmatrix} \text{ (m)}, \quad (\text{C.30})$$

$$m_3 = 2.5 \text{ kg}, \quad (\text{C.31})$$

$$A_{3,o} = \begin{bmatrix} 0 & -1 & 0 \\ \sin\theta_3 & 0 & -\cos\theta_3 \\ \cos\theta_3 & 0 & \sin\theta_3 \end{bmatrix}, \quad (\text{C.32})$$

sendo que θ_3 é o ângulo de posição da junta 3, e também varia de 0 a 90°.

APÊNDICE DANÁLISE PARAMÉTRICA DO TREINAMENTO DE UMA REDE NEURALD.1 O SISTEMA DINÂMICO UTILIZADO COMO EXEMPLO

Durante o treinamento de uma rede neural, diversos são os fatores que possibilitam uma maior ou menor velocidade de aprendizado. Entre eles, o número de pontos amostrados, a precisão final desejada, o pesos utilizados na inicialização, etc. Uma análise paramétrica destas variáveis, então, viria a trazer valiosas contribuições no entendimento de como o aprendizado e também a rede se comportam quando estes são modificados. Visto que diversos sistemas dinâmicos possuem resposta cíclica (pêndulo, sistema massa-mola), decidiu-se utilizar uma função senoidal como provedora dos pontos a serem aproximados pela rede. Os pontos são então amostrados a intervalos regulares na função dada por:

$$y^d(t) = 3 \sin(2\pi t / T) + 5 \quad (\text{D.1})$$

A rede é a seguir treinada utilizando-se como entradas os valores de $y^d(t)$, $y^d(t-1)$, $y^d(t-2)$..., ou seja: o valor atual seguido de valores atrasados no tempo. Na saída, a rede deve aproximar o valor de $y^d(t+1)$. Na forma proposta, o tempo t não aparece explicitamente como argumento da rede, o que a tornaria um sistema algébrico. A saída da rede, y^m , é então uma função que aproxima y^d , e é dada por:

$$y^m(t+1) = f_m(y^d(t), y^d(t-1), y^d(t-2), \dots) \quad (\text{D.2})$$

Utilizou-se uma rede neural composta por duas camadas: uma camada oculta, com função de ativação sigmóide, e uma camada de saída, com função de ativação linear. Empregou-se o método de mínimos quadrados recursivos para efetuar o treinamento, onde este demonstrou uma elevada capacidade de minimização do erro

apresentado pela rede, aliado a uma rápida convergência, quando comparado ao método do gradiente (retro-propagação).

A função senoidal foi dividida em 20 pontos ($T = 20$), e estes pontos, juntamente com seus valores atrasados foram fornecidos à rede para treinamento. A cada iteração, os pontos são processados seqüencialmente segundo as equações recursivas apresentadas no Capítulo 2. Ao final do processamento, o erro médio quadrático é calculado segundo a expressão:

$$\varepsilon = \frac{\sum_{t=1}^T (y^d - y^m)^T (y^d - y^m)}{2T} \quad (\text{D.3})$$

Os pesos são inicializados através de uma função aleatória, com uma semente ajustável previamente. A faixa de variação dos pesos iniciais também pode ser ajustada, de forma a evitar que, eventualmente, o processo de minimização do erro fique confinado a um mínimo local.

Durante o treinamento, deve-se observar o valor apresentado pelos neurônios da camada oculta. Se um ou mais destes neurônios estiver “saturado”, isto é apresentar na saída insistentemente 1 ou -1, significa que o aprendizado dos pesos associados a este neurônio estará comprometido, visto que a derivada de sua função de ativação será nula. A forma utilizada para a verificação da saturação dos pesos, nos exemplos, foi realizada através da observação do valor absoluto do argumento da função de ativação sigmóide, na forma:

$$N_{sat} = \sum_{k=1}^{l-1} \sum_{i=1}^{n_k} s_i^k, \text{ onde } \begin{cases} s_i^k = 1, & \text{se } |\bar{x}_i^k| \geq x_{sat} \\ s_i^k = 0, & \text{se } |\bar{x}_i^k| < x_{sat} \end{cases} \quad (\text{D.4})$$

onde x_{sat} foi fixado em 8. Na relação acima foi suposto que todas as camadas ocultas possuem função de ativação sigmóide, exceto, é claro, a camada de saída.

O processo de treinamento foi iniciado com uma rede composta de 4 entradas (valor atual mais duas entradas atrasadas mais o patamar unitário), 5 neurônios na camada oculta e 1 neurônio na saída. A matriz de covariância inicial foi adotada como uma matriz diagonal cujos elementos são iguais a 100. Por sua vez, os pesos são inicializados com dispersão entre -0.05 e 0.05, e semente nula. Finalmente, adotou-se um fator de esquecimento λ de 0.995. Os primeiros resultados levaram a um erro mínimo de cerca de 10^{-3} , suficientemente bom para o problema proposto, mas impróprio para o treinamento da dinâmica de um satélite. Decidiu-se então modificar o programa computacional para trabalhar em dupla precisão, o que resultou em um erro de 10^{-5} , após 180 iterações, conforme a Figura D.1. Iterações subsequentes levam a rede a divergir, provavelmente causadas por saturação de neurônios, conforme pode ser observado na Figura D.2, que mostra o comportamento de dois pesos da rede: w_{11}^1 e w_{11}^2 . Veja que o valor do peso associado ao primeiro neurônio da primeira camada diverge após 150 iterações, provocando a saturação deste neurônio.

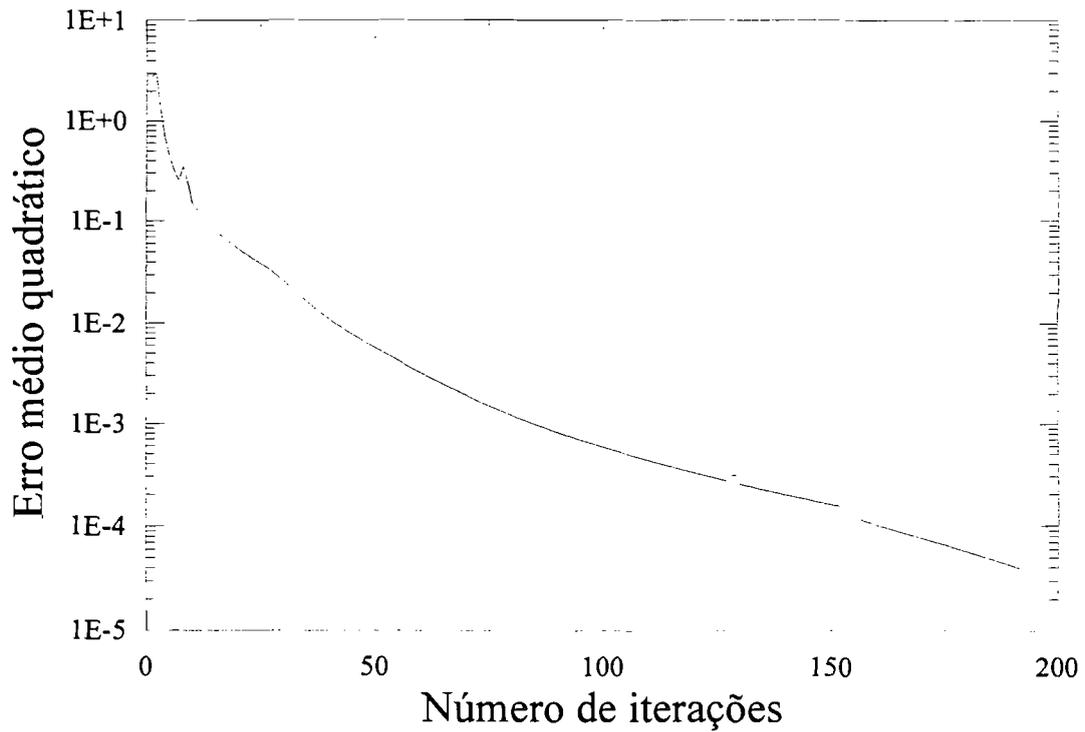


Fig. D.1 - Evolução do erro obtido durante o treinamento da rede neural.

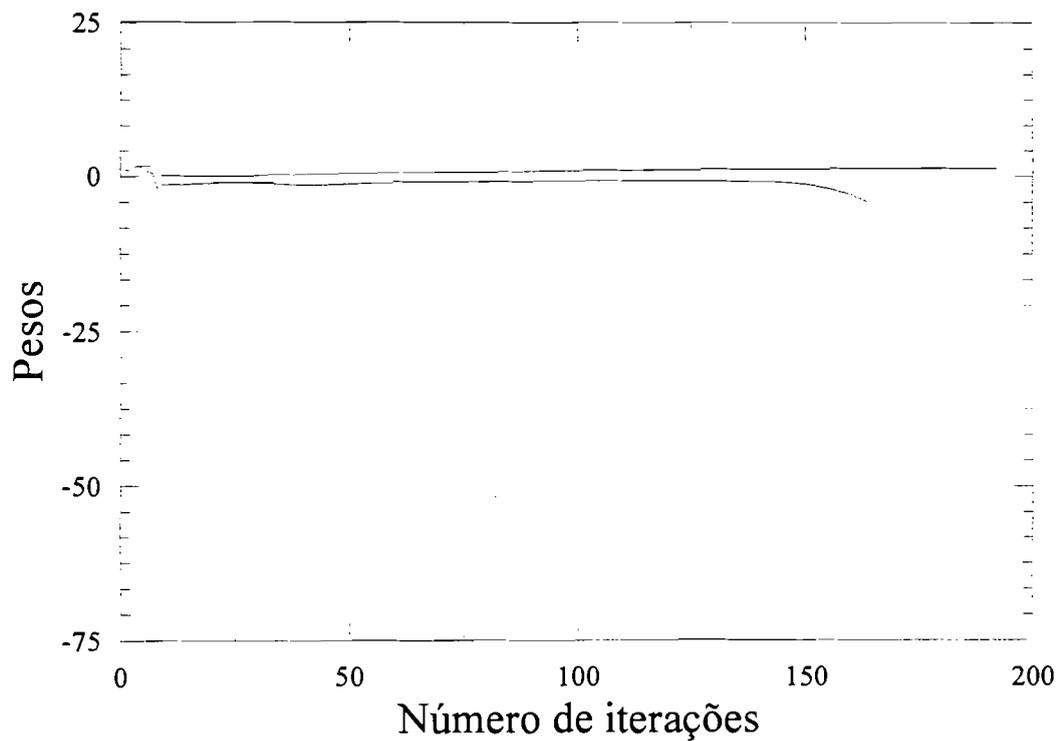


Fig. D.2 - Valores de dois pesos da rede neural durante o treinamento.

D.2 A ANÁLISE PARAMÉTRICA DO TREINAMENTO

Após algumas modificações nos parâmetros da rede, conseguiu-se um erro de saída ainda menor, da ordem de 10^{-12} , depois de 200 iterações, mostrado na Figura D.3. Neste caso o treinamento foi abortado após 200 iterações, com um erro pequeno o suficiente para garantir a precisão requerida ao sistema dinâmico do satélite, embora o treinamento pudesse ainda ser melhorado. Os valores utilizados neste exemplo foram: 10 neurônios na camada oculta, 4 entradas (valor atual, dois valores atrasados e patamar de ativação), dispersão inicial dos pesos de 0.5, semente de 1 e fator de esquecimento de 0.995.

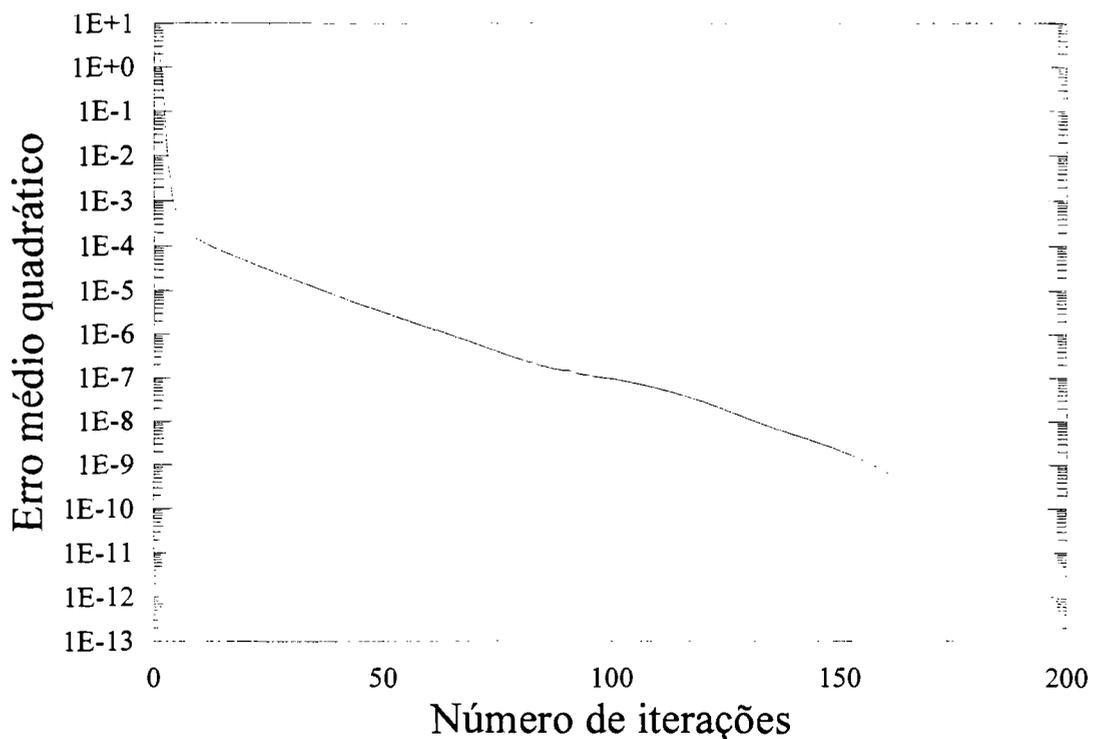


Fig. D.3 - Erro obtido pela rede após uma análise paramétrica.

Durante a análise paramétrica, procurou-se inicialmente verificar a suscetibilidade do erro frente ao número de neurônios da camada oculta. Não foi possível obter convergência com menos que 5 neurônios, e mesmo com este valor o erro mínimo foi de $5 \cdot 10^{-7}$ (após 280 iterações). O melhor resultado foi conseguido quando

empregou-se 10 neurônios na camada oculta, conforme indicado na Figura D.4, porém não se pode garantir ser este o valor ideal, em virtude dos demais parâmetros que permaneceram fixos durante este exemplo. Os valores utilizados foram iguais aos do exemplo anterior, exceto para a curva correspondente a 5 neurônios, cuja semente de inicialização dos pesos foi nula, já que com a semente unitária o erro mínimo era de somente $1 \cdot 10^{-3}$.

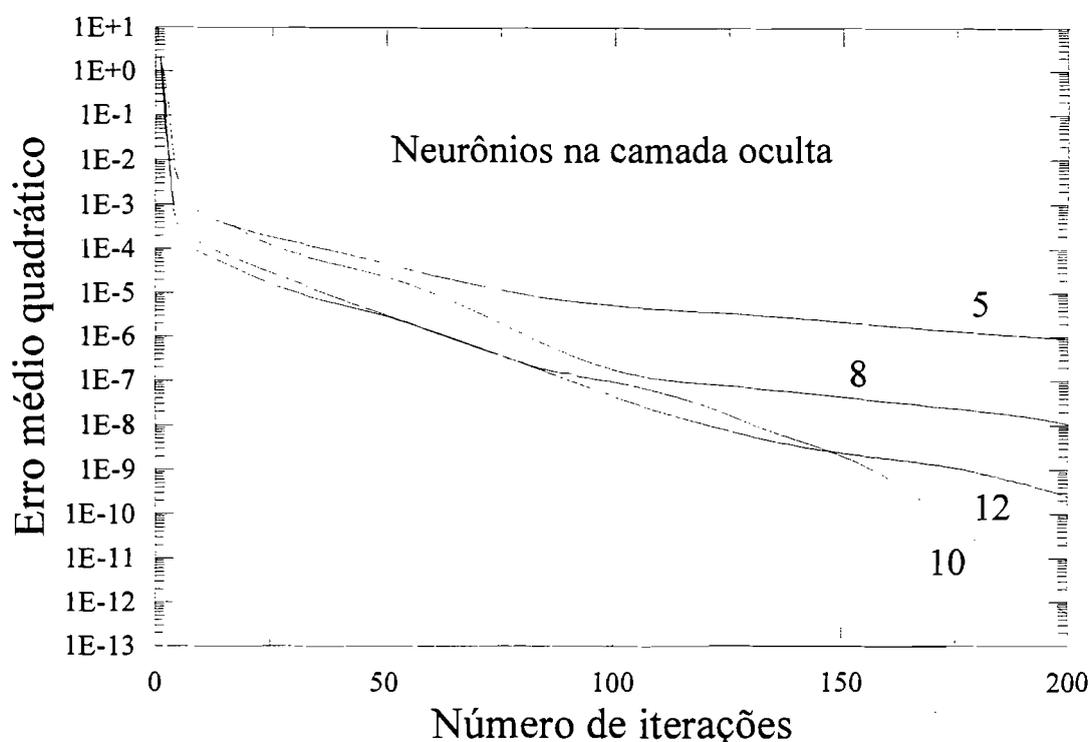


Fig. D.4 - Erro obtido pela rede em função do número de neurônios da camada oculta.

Ao se alterar o número de entradas atrasadas, tanto quanto o número de neurônios na camada oculta, podem ocorrer, durante o treinamento, que alguns neurônios fiquem saturados, comprometendo todo o processo de aprendizado. É necessário, nestes casos, refazer o treinamento utilizando diferentes condições iniciais para os pesos, tanto no que se refere ao valor inicial da semente aleatória quanto também o valor da amplitude máxima. No gráfico seguinte, Figura D.5, é mostrada a convergência da rede em função do número de valores defasados no tempo (inclusive o valor do instante atual). Vê-se que os melhores resultados são obtidos com 3 entradas

(valor atual mais dois valores atrasados), o que está de acordo com Billings et al., 1992, que estabelecem que o número de entradas atrasadas deve satisfazer a ordem do sistema dinâmico, que no caso da função senoidal, é igual a 2. Os demais parâmetros permaneceram iguais ao da Figura D.3.

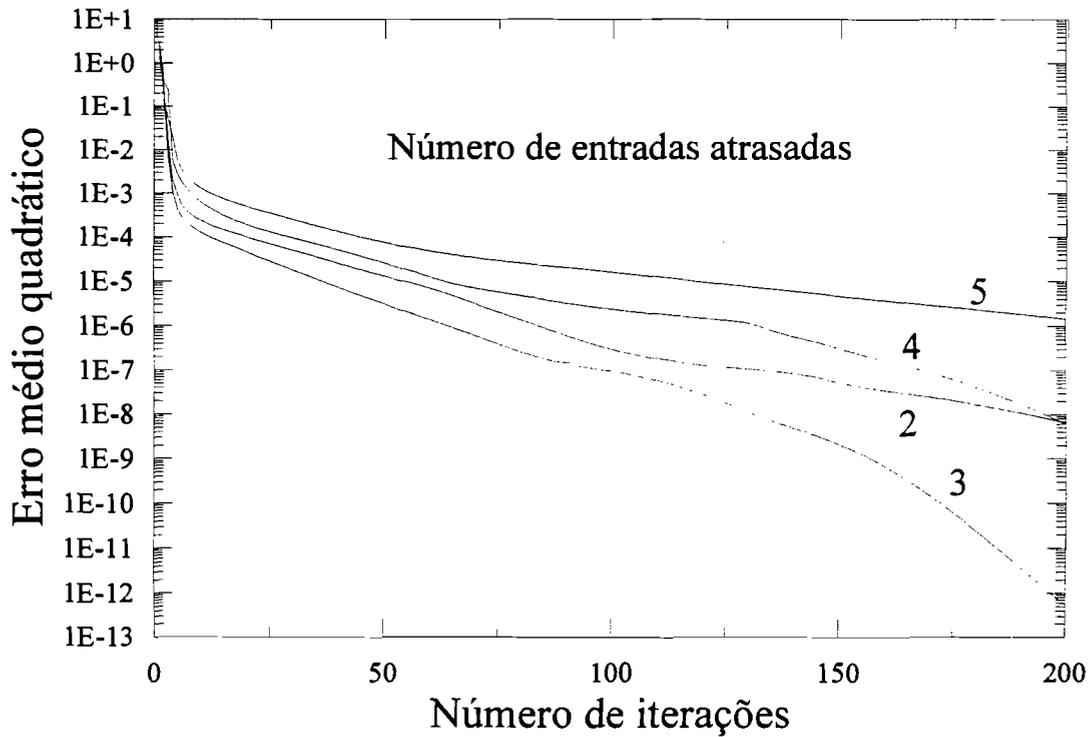


Fig. D.5 - Erro apresentado pela rede em função do número de entradas atrasadas.

Ao se analisar a suscetibilidade da rede aos valores iniciais dos pesos, verificou-se que a convergência depende não apenas destes valores, mas também da amplitude de dispersão inicial. Os melhores resultados, pelo menos para este problema, foram obtidos com dispersão unitária (isto é, com pesos compreendidos entre -0.5 e 0.5). Os valores da semente inicial (*seed*) para geração aleatória dos pesos, bem como o valor de sua dispersão devem ser utilizados como alternativa para evitar possíveis saturações nos neurônios. No exemplo mostrado na Figura D.6, exceto pelo teste número 2 (nenhum neurônio saturado), todos os outros apresentaram um maior ou menor grau de neurônios saturados (entre 3 e 5). Isto reforça a idéia de que uma forma

eficiente de verificar a convergência e o grau de adequação dos pesos iniciais é através do acompanhamento da saturação nos pesos.

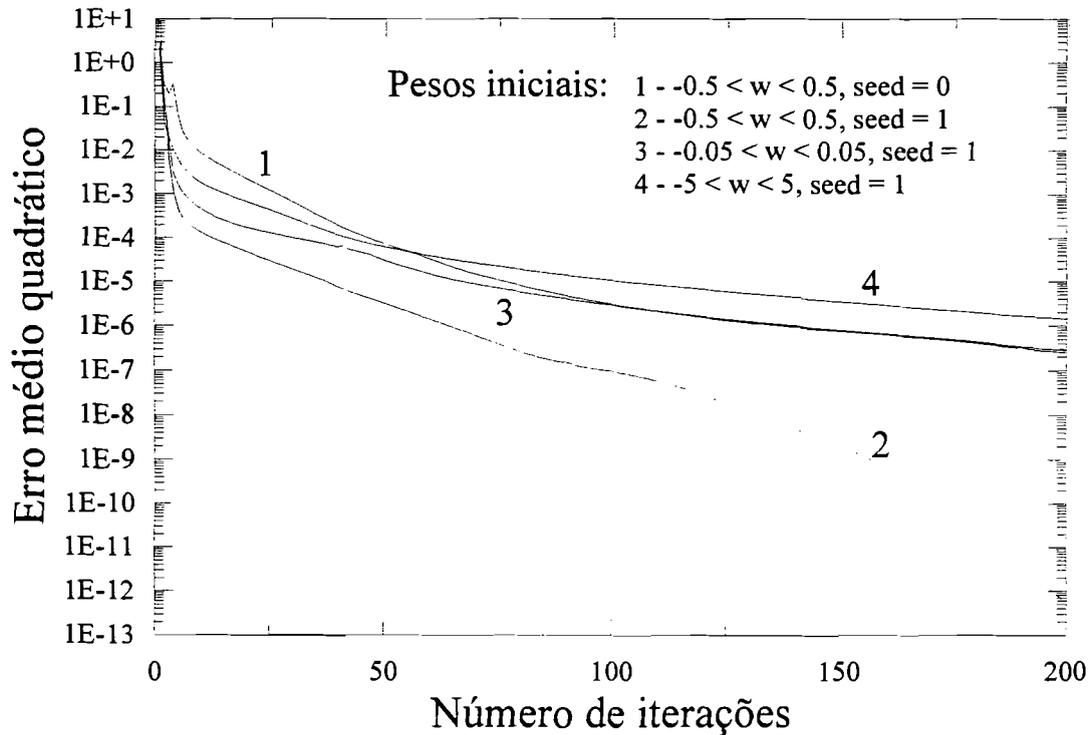


Fig. D.6 - Erro apresentado pela rede em diferentes condições iniciais para os pesos.

Na figura D.7, analisou-se o comportamento da rede sob diferentes valores do fator de esquecimento. Como o treinamento pondera com um peso maior as últimas observações quanto menor for o fator de esquecimento, é evidente que o aprendizado é mais rápido neste caso. No entanto, para evitar que as observações anteriores sejam descartadas muito rapidamente, deve-se empregar um fator relativamente grande. Se o fator for unitário, as observações anteriores não são descartadas. Pode-se estabelecer uma relação entre o fator de esquecimento, λ , e o número $N_{1/2}$ de observações processadas cujos ganhos superam 50% do ganho da última medida, na forma:

$$N_{1/2} = -\frac{\ln 2}{\ln \lambda} \quad (\text{D.5})$$

O número de medidas dentro do tempo de meia-vida referentes a cada um dos fatores de esquecimento da Figura D.7 são: 34 medidas para $\lambda = 0.98$, 69 para $\lambda = 0.99$ e 138 para $\lambda = 0.995$. Veja também que nesta figura prosseguiu-se com o treinamento até que este divergisse, o que levou a um erro de $5 \cdot 10^{-17}$ para um fator de esquecimento de 0.995. Os demais parâmetros da rede permaneceram iguais ao da Figura D.3.

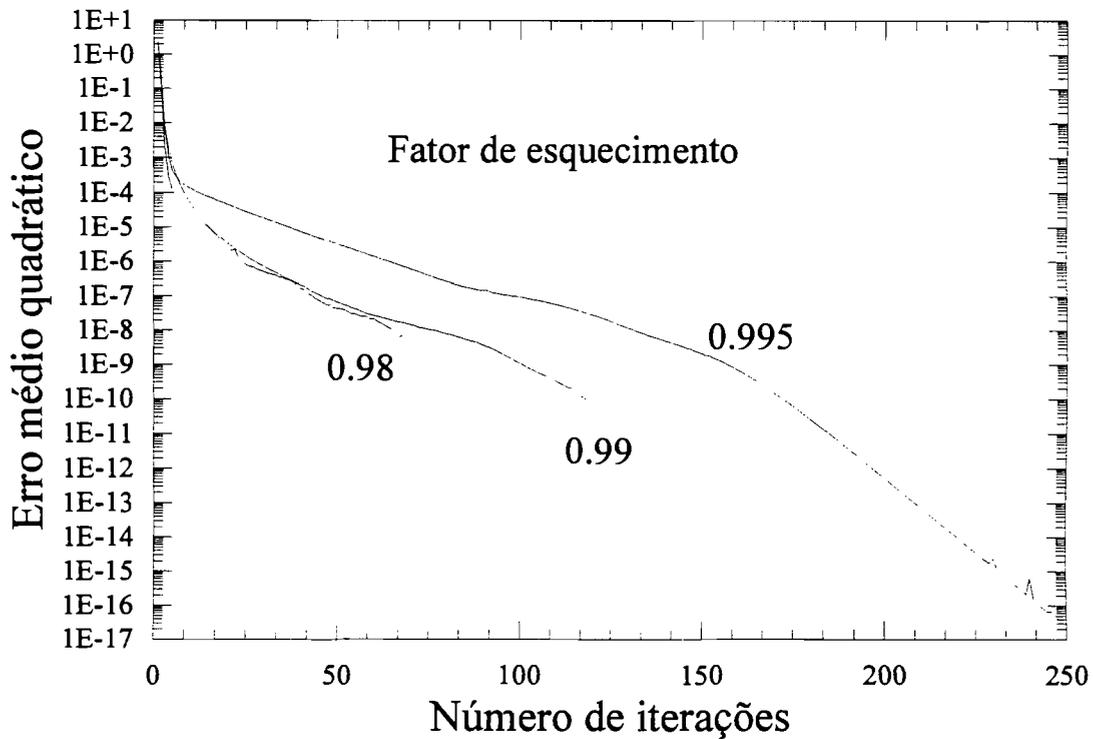


Fig. D.7 - Erro apresentado pela rede sob diferentes fatores de esquecimento.

A matriz de covariância é inicializada na forma diagonal, dada por:

$$P(0) = \sigma_0^2 \mathbf{1} \quad (\text{D.6})$$

onde $\mathbf{1}$ é a matriz identidade e σ_0^2 é um escalar ajustado para assegurar uma convergência inicial do treinamento. O impacto da magnitude deste valor foi analisado na Figura D.8. Constatou-se que o aprendizado é pouco influenciado pelo valor de σ_0^2 .

Valores diferentes devem ser utilizados, no entanto, para evitar convergências para mínimos locais (no caso de $\sigma_0^2 = 10$ e $\sigma_0^2 = 1000$), e também como tentativa para evitar saturações dos neurônios.

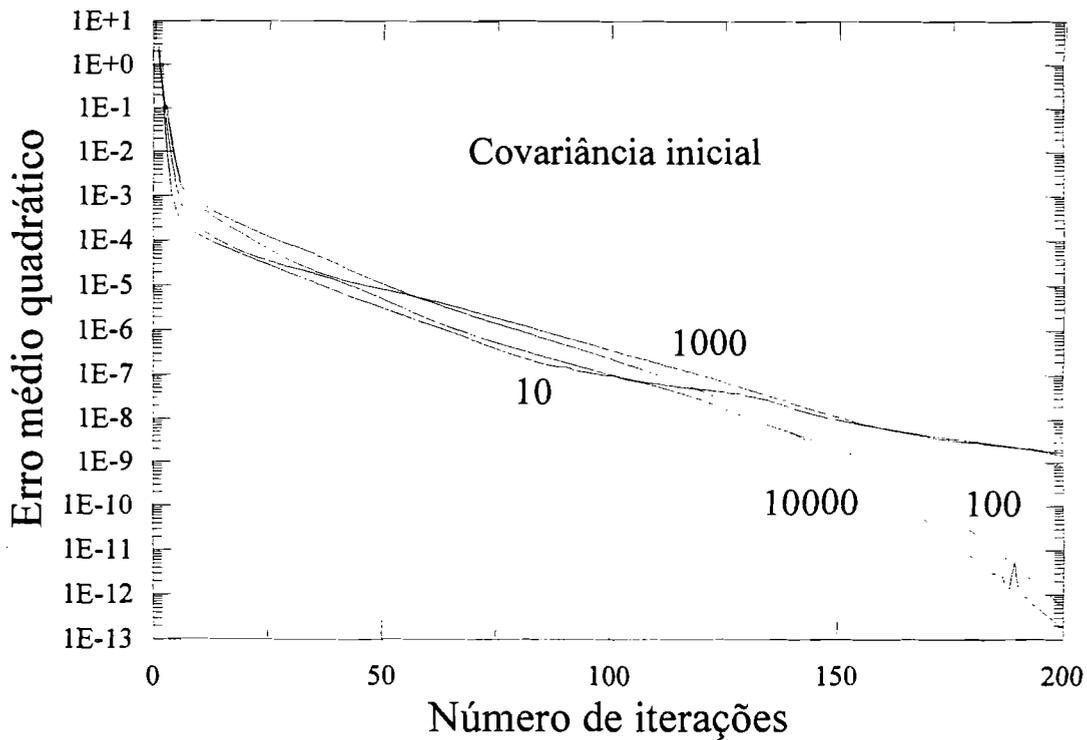


Fig. D.8 - Impacto da matriz de covariância inicial no aprendizado da rede.

Tanto a velocidade do treinamento, quanto o erro mínimo obtido pela rede dependem da quantidade de informações que é fornecida à rede para efetuar o aprendizado. Geralmente um aumento na quantidade de pontos provoca uma diminuição na velocidade (já que há mais pontos a serem processados) e um aumento do erro final atingido pela rede, conforme a Figura D.9. Com 50 pontos amostrados ($T = 50$), obteve-se um erro de $5 \cdot 10^{-9}$, antes que o processo de treinamento divergisse. Este erro é significativamente maior que o erro conseguido com 10 e 20 pontos, porém há que se destacar que o erro obtido na interpolação é semelhante para todos eles. Uma vez definido o número de pontos em que a rede será treinada, deve-se procurar ajustar os demais parâmetros de forma a evitar a saturação de neurônios e aumentar a precisão atingida.

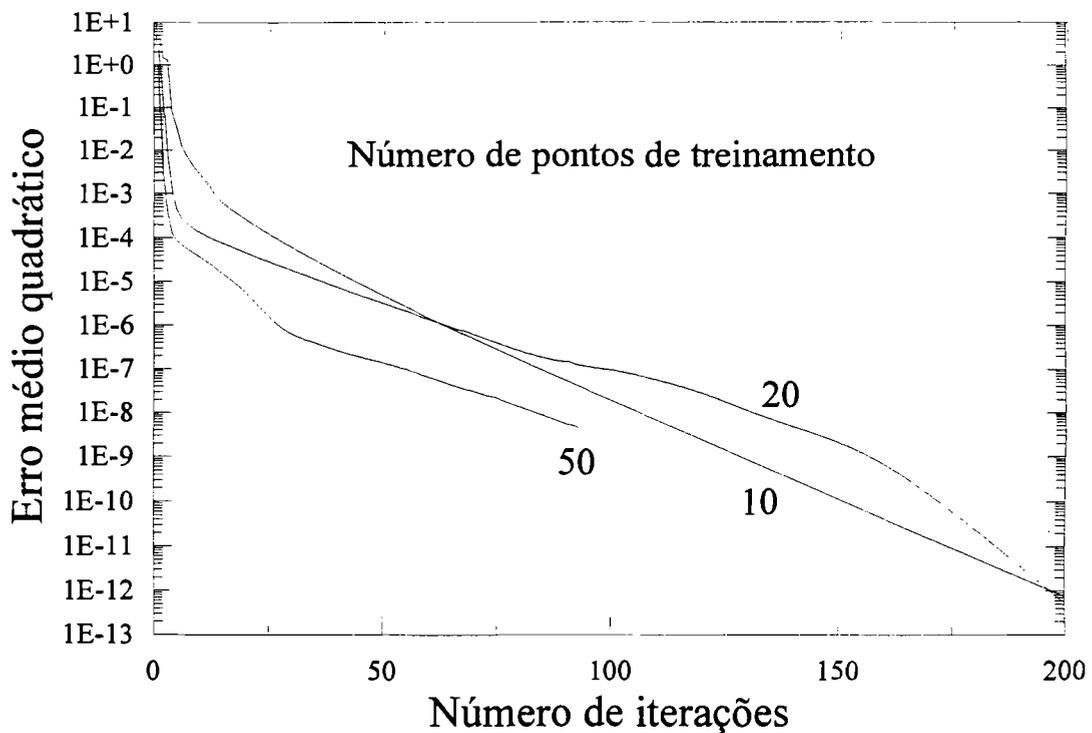


Fig. D.9 - Erros obtidos com quantidades diferentes de pontos treinados.

D.3 A ANÁLISE DOS ERROS DE TREINAMENTO

Uma vez definidos os parâmetros do treinamento e realizado o aprendizado da rede, o erro pode ser computado para cada ponto considerado. No caso da função seno, este erro permanece abaixo de $2 \cdot 10^{-6}$, como visualizado na Figura D.10, onde foram consideradas as seguintes condições iniciais: número máximo de iterações igual a 200, covariância inicial igual à matriz diagonal com elementos iguais a 100, fator de esquecimento λ de 0.995. A rede treinada possui 10 neurônios na camada oculta, e 3 entradas atrasadas (o instante atual e mais 2 entradas em instantes anteriores). Os pesos foram iniciados com semente e amplitude unitárias. Foram utilizados 20 pontos amostrados em um período (2π).

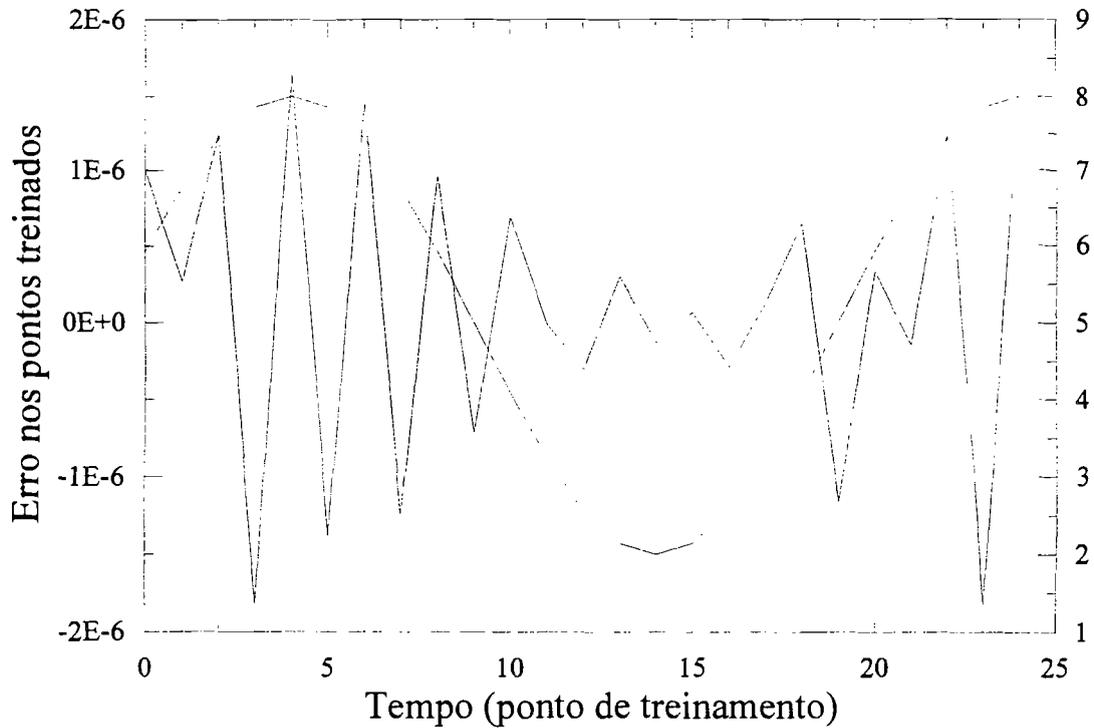


Fig. D.10 - Erro ao final do treinamento nos pontos amostrados. A função seno foi sobreposta de forma ilustrativa, com escala à direita.

A curva mostrada na Figura D.11 apresenta a capacidade da rede de interpolar valores entre os pontos de treinamento. A figura mostra o erro obtido pela rede em pontos intermediários, nos quais ela não foi treinada. O erro mantém-se menor que $6 \cdot 10^{-4}$, diminuindo bastante ao redor dos pontos que foram utilizados no treinamento. Compare estes erros com os erros obtidos nos pontos utilizados no treinamento (Figura D.10), da ordem de $2 \cdot 10^{-6}$. Os erros de interpolação são portanto 100 vezes maiores.

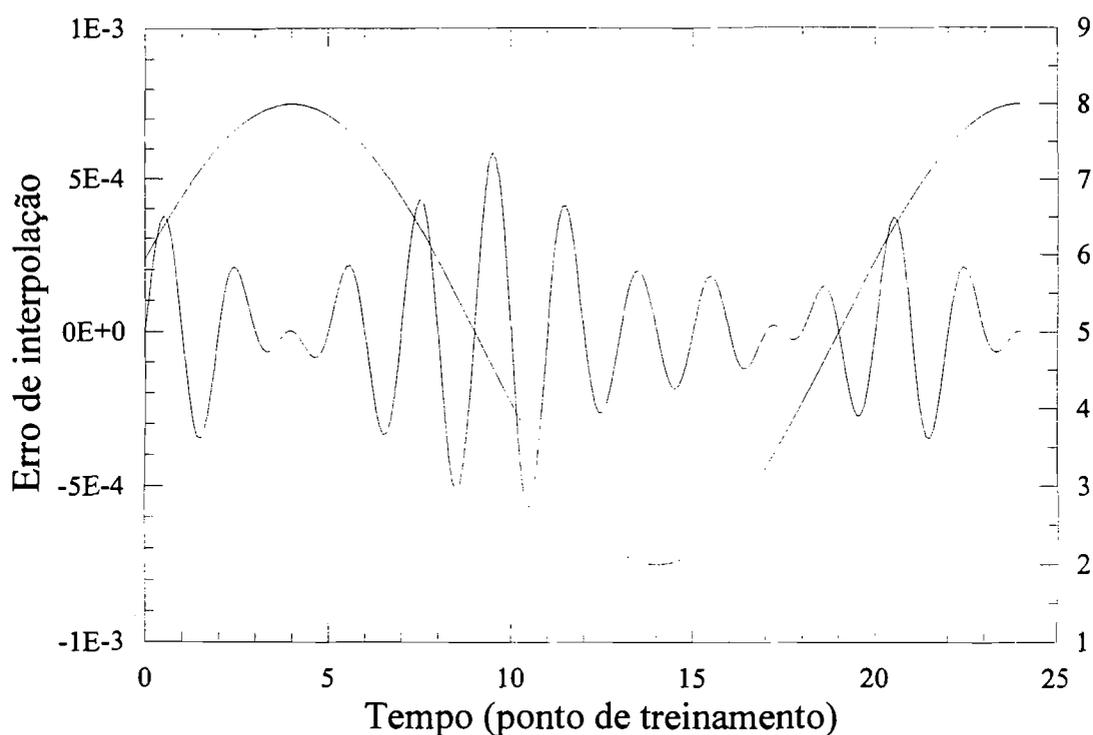


Fig. D.11 - Erro de interpolação entre os pontos amostrados. A função seno foi sobreposta com escala à direita.

Nos gráficos anteriores, a rede foi alimentada com entradas fornecidas pela função senoidal. Já na Figura D.12, a rede foi alimentada inicialmente com valores da função seno, substituídos pela saída da própria rede, conforme esta gerava novos pontos. Neste caso a rede tem a seguinte característica:

$$y^m(t+1) = f_m(y^m(t), y^m(t-1), y^m(t-2), \dots) \quad (D.7)$$

Embora o erro, com relação à função senoidal aumente exponencialmente com o tempo de propagação (devido à acumulação sucessiva), ainda assim este erro permanece menor que $4 \cdot 10^{-4}$ (quadrático) após 6 períodos completos. Isto garante que a rede consegue aproximar muito bem um sistema dinâmico.

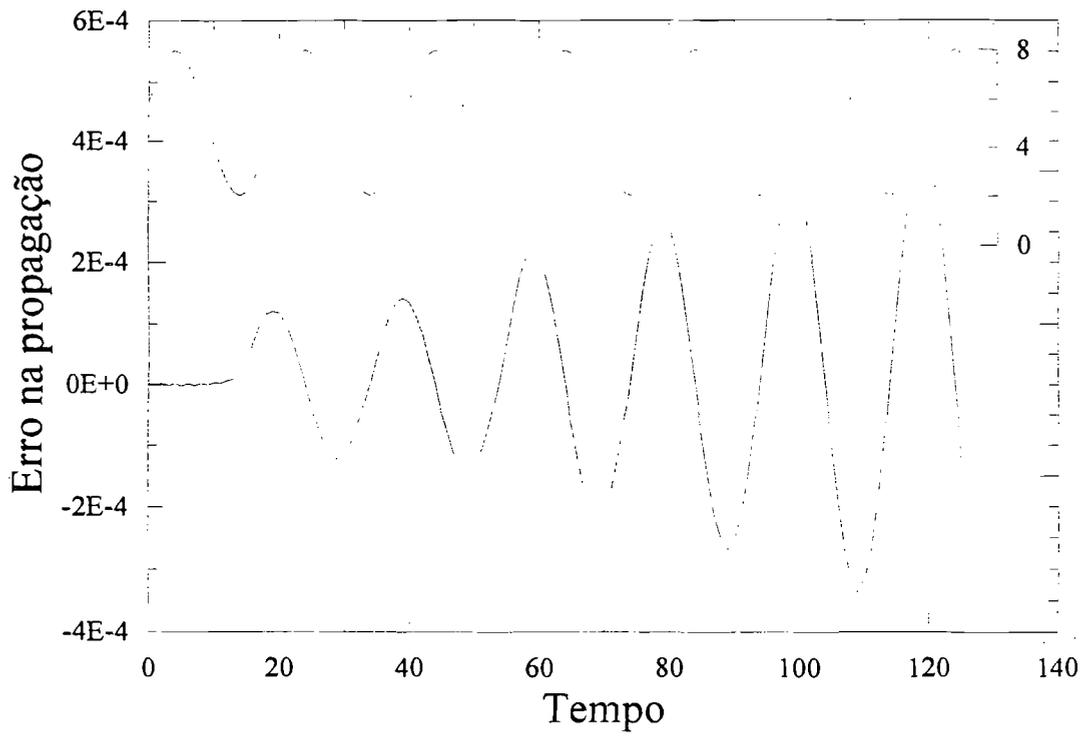


Fig. D.12 - Erro de propagação da rede neural. A função seno foi sobreposta com finalidade ilustrativa.

**PUBLICAÇÕES TÉCNICO-CIENTÍFICAS
EDITADAS PELO INPE**

RELATÓRIOS DE PESQUISA

- Reportam resultados de pesquisa tanto de natureza técnica quanto científica.

NOTAS TÉCNICO-CIENTÍFICAS

- Incluem resultados preliminares de pesquisa, descrição de equipamentos, software, sistemas e experimentos, apresentação de testes, dados e atlas, e documentação de projetos de engenharia.

MANUAIS TÉCNICOS

- Descrevem normas, procedimentos, instruções e orientações.

PUBLICAÇÕES DIDÁTICAS

- Apostilas, notas de aula e manuais didáticos.

TESES E DISSERTAÇÕES

- Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

PUBLICAÇÕES SERIADAS

- Periódicos Técnico-científicos: Boletim de Sensoriamento Remoto, Climanálise: Boletim de Monitoramento e Análise Climática.
- Anais de Eventos