

UNIVERSIDADE FEDERAL DE ITAJUBÁ
INST. DE ENG. DE SISTEMAS E TECNOLOGIAS DA INFORMAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

DISSERTAÇÃO DE MESTRADO

**Sintonia Automatizada de Ganhos de Retroação de
Estados em Sistemas de Controle com o Auxílio de
Redes Neurais Artificiais**

Autor: **Bruno Mohallem Paiva**

Orientador: **Prof. Dr. Benedito Isaías Lima Lopes**

Co-orientador: **Prof. Dr. Carlos Alberto Murari Pinheiro**

Itajubá, Junho de 2008

UNIVERSIDADE FEDERAL DE ITAJUBÁ
INST. DE ENG. DE SISTEMAS E TECNOLOGIAS DA INFORMAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

DISSERTAÇÃO DE MESTRADO

**Sintonia Automatizada de Ganhos de Retroação de
Estados em Sistemas de Controle com o Auxílio de
Redes Neurais Artificiais**

Autor: **Bruno Mohallem Paiva**

Orientador: **Prof. Dr. Benedito Isafias Lima Lopes**

Co-orientador: **Prof. Dr. Carlos Alberto Murari Pinheiro**

Curso: **Mestrado em Engenharia Elétrica**

Área de Concentração: **Automação e Sistemas Elétricos Industriais**

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica como parte dos requisitos para obtenção do Título de Mestre em Engenharia Elétrica.

Itajubá, Junho de 2008

MG – Brasil

UNIVERSIDADE FEDERAL DE ITAJUBÁ
INST. DE ENG. DE SISTEMAS E TECNOLOGIAS DA INFORMAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

DISSERTAÇÃO DE MESTRADO

**Sintonia Automatizada de Ganhos de Retroação de
Estados em Sistemas de Controle com o Auxílio de
Redes Neurais Artificiais**

Autor: **Bruno Mohallem Paiva**

Orientador: **Prof. Dr. Benedito Isaías Lima Lopes**

Co-orientador: **Prof. Dr. Carlos Alberto Murari Pinheiro**

Composição da Banca Examinadora:

Dr. Fernando José de Oliveira Moreira – EMBRAER

Prof^a. Dr^a. Helga Gonzaga Martins - UNIFEI

Prof. Dr. Carlos Alberto Murari Pinheiro (Co-orientador) - UNIFEI

Prof. Dr. Benedito Isaías Lima Lopes (Orientador) - UNIFEI

Sumário

Lista de Figuras	iii
Lista de Tabelas	iv
Agradecimentos	v
Resumo	vii
Abstract	viii
Simbologia	ix
Siglas e Abreviaturas.....	xi
Capítulo 1	
INTRODUÇÃO	1
Capítulo 2	
SISTEMAS DE CONTROLE AUTOMÁTICO	4
2.1 SISTEMAS DINÂMICOS	4
2.2 SISTEMAS DE CONTROLE	8
2.3 REALIMENTAÇÃO DE ESTADOS	9
Capítulo 3	
REDES NEURAIS ARTIFICIAIS.....	13
3.1 INTRODUÇÃO.....	13
3.2 MODELO DE NEURÔNIOS ARTIFICIAIS	13
3.3 ARQUITETURA DE REDES NEURAIS ARTIFICIAIS	15
3.4 TREINAMENTO DE UMA REDE NEURAL ARTIFICIAL.....	17

Capítulo 4**SINTONIA DE GANHOS DE RETROAÇÃO DE ESTADOS COM O AUXÍLIO DE REDES NEURAIS ARTIFICIAIS..... 19**

4.1 INTRODUÇÃO.....	19
4.2 VISÃO GERAL.....	20
4.3 DETALHAMENTO DA METODOLOGIA.....	23
4.3.1 Requisitos de projeto.....	23
4.3.2 Avaliação do atendimento aos requisitos de projeto.....	23
4.3.3 Obtenção de um ponto de partida para a sintonia.....	25
4.3.4 Varredura de ganhos em torno do ponto de partida.....	27
4.3.5 Sintonia utilizando os conjuntos de ganhos obtidos pela varredura.....	29
4.3.6 Otimização do algoritmo através de uma rede neural artificial.....	29
4.3.7 Utilização da rede neural artificial.....	31
4.3.8 Validação dos melhores casos e revisão de projeto (<i>design review</i>).....	31

Capítulo 5**EXEMPLO DE APLICAÇÃO, RESULTADOS E DISCUSSÕES 33**

5.1 SISTEMA C* PARA CONTROLE DE AERONAVES.....	33
5.1.1 Introdução.....	33
5.1.2 Modelagem.....	34
5.2 SINTONIA DE UMA MALHA DE CONTROLE TIPO C*.....	37
5.2.1 Requisitos de projeto e respectivas formas de avaliação.....	37
5.2.2 Demais configurações do algoritmo.....	41
5.2.3 Análise dos resultados obtidos.....	42

Capítulo 6**CONCLUSÕES E PERSPECTIVAS FUTURAS..... 50****REFERÊNCIAS BIBLIOGRÁFICAS 52****Apêndice A****MATRIZES DE ESTADO UTILIZADAS NO EXEMPLO DE APLICAÇÃO..... 56****Apêndice B****ILUSTRAÇÕES COM OS EIXOS E REFERÊNCIAS RELATIVOS AO MOVIMENTO DE UMA AERONAVE..... 57****Apêndice C****OBTENÇÃO DA ACELERAÇÃO NORMAL SENTIDA PELO PILOTO..... 59**

Lista de Figuras

Figura 2.1 – Controlador em malha aberta.....	9
Figura 3.1 – Estrutura de um neurônio artificial	14
Figura 3.2 – Neurônio artificial estilizado.....	14
Figura 3.3 – Funções de ativação: (a) Heaviside; (b) sigmoideal; (c) linear.	15
Figura 3.4 – Rede neural do tipo feedforward.....	16
Figura 4.1 – Fluxograma das etapas da metodologia	22
Figura 4.2 – Arquitetura da rede neural tipo MLP proposta	30
Figura 5.1 – Modelagem do sistema C^*	35
Figura 5.2 – Sistema C^* em malha aberta.....	35
Figura 5.3 – Modelo simplificado (malha aberta)	36
Figura 5.4 – Diagrama de Pólos e Zeros	38
Figura 5.5 – Diagrama de Nichols.....	39
Figura 5.6 – Diagrama de Pólos e Zeros	40
Figura 5.7 – GUI desenvolvida em MATLAB.....	42
Figura B1 – Eixos de referência e ângulos.....	57
Figura B2 – Movimentos rotacionais de uma aeronave	58

Lista de Tabelas

Tabela 1 – Conjunto de ganhos inicial obtido pelo algoritmo.....	42
Tabela 2 – Conjuntos de fatores de multiplicação que geraram os menores custos.....	43
Tabela 3 – Respostas obtidas com a sintonia do conjunto inicial de ganhos	44
Tabela 4 – Respostas obtidas com o melhor conjunto de ganhos encontrado.....	46
Tabela 5 – Tempos das etapas para o exemplo de aplicação.....	48
Tabela 6 – Impacto do aumento da resolução no tempo de execução.....	49

Agradecimentos

Agradeço a Deus pelo dom da vida e da capacidade de aprendizagem.

Aos meus pais, Sonia e Luiz Eduardo, que me ensinaram desde cedo a importância dos estudos e da busca constante pela capacitação e superação.

Ao meu irmão, Henrique, pela transmissão de sua experiência técnica, científica e pessoal, pelo constante incentivo e pelas valorosas orientações ao longo deste trabalho.

À minha amada, Joanize, por todo apoio, estímulo e compreensão durante o desenvolvimento deste trabalho.

Ao meu orientador, Prof. Isaías, por ter acreditado na minha capacidade e na viabilidade da pesquisa, pela fundamental orientação no desenvolvimento deste projeto e pelo constante incentivo.

Ao meu co-orientador, Prof. Pinheiro, pelos conhecimentos transmitidos.

Ao meu amigo, Álvaro Polati, pelo apoio a este projeto e pela inestimável orientação sobre sistemas de controle aplicados à aeronáutica.

Ao CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico, pelo suporte financeiro.

*“O importante não é aquilo que fazem de nós,
mas o que nós mesmos fazemos do
que os outros fizeram de nós.”*

Jean-Paul Sartre

Resumo

PAIVA, B. M. (2008), *Sintonia Automatizada de Ganhos de Retroação de Estados em Sistemas de Controle com o Auxílio de Redes Neurais Artificiais*, Itajubá, MG. Dissertação (Mestre em Ciências em Engenharia Elétrica) - Instituto de Engenharia de Sistemas e Tecnologias da Informação, Universidade Federal de Itajubá.

Este trabalho propõe uma metodologia para a sintonia de ganhos de sistemas de controle automático baseados em realimentação de estados. O objetivo da metodologia é encontrar de forma eficiente condições de sintonia que atendam aos requisitos de projeto previamente definidos. O cumprimento destes requisitos é quantificado objetivamente, através de índices de desempenho (custos), que avaliam os diferentes critérios de desempenho do sistema. Um conjunto inicial de ganhos de realimentação é obtido através da técnica LQR (Linear Quadratic Regulator), aplicada a um modelo simplificado do sistema de controle. Em seguida, para cada ganho de realimentação, é definido um fator multiplicador, ao qual são atribuídos valores discretizados dentro de um intervalo determinado, e então é gerada uma série de conjuntos de fatores multiplicadores, contendo todas as combinações possíveis entre esses valores. Os ganhos iniciais são multiplicados por cada um dos conjuntos de fatores multiplicadores, gerando uma série de conjuntos de ganhos de realimentação. Para cada um desses conjuntos de ganhos, realiza-se uma simulação com o modelo completo do sistema em malha fechada, e obtêm-se custos (índices de desempenho) correspondentes. Os conjuntos de ganhos e os respectivos custos associados são utilizados para o treinamento de uma rede neural do tipo MLP (multi-layer perceptron), que estimará as possíveis melhores soluções, que possuem os menores custos associados. Essas soluções são validadas por meio de novas simulações, utilizando o sistema completo e então, por fim, é escolhida a solução que melhor sintoniza o sistema. A metodologia é ilustrada com uma aplicação em um sistema de controle do tipo C^* (“C-star”) controlando um modelo longitudinal linearizado de uma aeronave. Os resultados mostram que os requisitos de projeto são satisfatoriamente atendidos com um custo computacional bastante eficiente e promissor.

Palavras-chave

Ganhos de Realimentação de Estados, Sintonia Automatizada, Redes Neurais Artificiais

Abstract

PAIVA, B. M. (2008), *Automatized State-Feedback Gains Tuning in Control Systems Supported by Artificial Neural Network*, Itajubá, MG. MSc. Degree Dissertation - System Engineering and Information Technology Institute, Federal University of Itajubá.

This work proposes a methodology for automatic tuning of gains of control systems based on state-feedback. The goal of the methodology is to efficiently find tuning solutions which meet pre-defined project requirements. The requirements achievement is objectively quantified through performance indices, which evaluate the different system performance criteria (costs). An initial feedback gain set is obtained by LQR (Linear Quadratic Regulators) technique, using a simplified model of the control system. So, for each feedback gain, it is defined a multiplicative factor, which are assigned to discretized values that lies on a determined interval, and then a series of multiplicative factors set is generated, based on all of the possible combinations between those values. The initial gains are multiplied for each of these multiplicative factors set, generating a series of feedback gains set. For each gain set, a simulation with the complete closed-loop system model is carried out and the corresponding costs (performance indices) are obtained. The feedback gains sets and their corresponding costs are used to train an artificial neural network of type MLP (multi-layer perceptron), which will estimate a set of the possibly best solutions, that have the lowest costs. These solutions are validated through new simulations using the complete system and then solution that best tune the system is finally chose. The methodology is illustrated with an application in a C* ("C-Star") control system, which controls a linearized longitudinal aircraft model. The results show that the project requirements are met with an efficient and reduced computing cost.

Keywords

State-Feedback Gains, Automatized Tuning, Artificial Neural Networks

Simbologia

A, B, C, D	Matrizes de estado de um sistema dinâmico
$x(t), \dot{x}(t)$	Vetor de variáveis de estado e respectiva derivada
$u(t)$	Excitação aplicada ao sistema dinâmico
$y(t)$	Saída do sistema
s	Variável da Transformada de Laplace
$F(s)$	Transformada de Laplace de uma função $f(t)$
$G(s)$	Função de Transferência de um determinado sistema
K	Vetor de ganhos de realimentação
K_i	i -ésimo ganho do vetor K
K_{LQR}	Vetor de ganhos de realimentação obtido pelo método LQR aplicado ao sistema simplificado
K_{iLQR}	i -ésimo ganho do vetor K_{LQR}
K_{K_i}	Fator de multiplicação do ganho K_i
$K_{K_i(i)}$	i -ésimo valor do fator K_{K_i} , obtido pela varredura em um intervalo pré-determinado
$K_{K_{LQR}(i)}$	i -ésimo conjunto de fatores de multiplicação, obtido pela combinação dos valores da varredura
$K'_{(i)}$	i -ésimo conjunto de ganhos gerado pela multiplicação entre o vetor de ganhos K_{LQR} e o i -ésimo conjunto de multiplicadores $K_{K_{LQR}(i)}$

r	Resolução de varredura do fator K_{Ki}
C^*	“C-Star”, sistema de controle proposto inicialmente pela Boeing para o controle do movimento longitudinal de aeronaves
α	Ângulo de ataque (ângulo entre o vetor velocidade do avião e o eixo de estabilidade do avião)
θ	Ângulo de arfagem da aeronave (eixo longitudinal)
q	<i>Pitch rate</i> (velocidade angular de arfagem da aeronave)
u	Velocidade horizontal do avião
$N_z, N_{z_{piloto}}$	Aceleração normal do avião e aceleração normal do avião sentida pelo piloto

Siglas e Abreviaturas

RNA	Rede Neural Artificial
MLP	Arquitetura <i>Multi-Layer Perceptron</i> (rede neural multicamadas)
LQR	<i>Linear-Quadratic Regulator</i> (Regulador linear-quadrático)
GUI	<i>Graphical User Interface</i> (Interface gráfica do usuário)

Capítulo 1

INTRODUÇÃO

O controle de sistemas dinâmicos é um importante ramo da engenharia que permite que se atinjam determinadas características ou especificações desejadas em sistemas dinâmicos. Através de um modelo matemático do sistema a ser controlado, conhecido como planta, e da definição de metas que são esperadas para o comportamento ou resposta final do sistema (requisitos), é possível projetar um sistema de controle que atue sobre a planta e possibilite que os requisitos de projeto sejam alcançados.

Existem na literatura vários métodos (Ogata, 2003) para o projeto e sintonia de malhas de controle, que permitem que o sistema sob controle mantenha respostas satisfatórias. Um método bastante conhecido de controle de sistemas dinâmicos é o método de realimentação de estados. A retroação de estados utiliza como lei de controle uma combinação linear dos estados do sistema (Philips e Harbor, 1997).

Existem técnicas clássicas utilizadas para se calcular os ganhos de sintonia de um sistema de controle com realimentação de estados, como as técnicas LQR e de alocação de pólos (Ogata, 2003). No entanto, cada técnica atende a objetivos específicos de sintonia, sendo mais recomendada para determinados tipos de problemas. Além disso, cada técnica apresenta limitações e restrições de aplicabilidade.

Métodos alternativos têm sido propostos para minimizar limitações de técnicas clássicas (Lien e Yu, 2008) e também para alcançar uma sintonia satisfatória em situações particulares. Alguns desses métodos alternativos utilizam técnicas de inteligência artificial (Darsono e

Labadie, 2007; García, 2005; Chen e Chiang, 2004; Calise e Rysdyk, 1998). Técnicas de inteligência artificial como as redes neurais artificiais, por exemplo, podem ser utilizadas na melhoria dos métodos tradicionais de sintonia de malhas de controle (Zárate e Bittencout, 2008; Singh *et al.*, 2005).

Seguindo esse tipo de abordagem, este trabalho apresenta uma metodologia alternativa para a sintonia de malhas de controle com realimentação de estados, em que atrasos de transporte e alguns estados do sistema não são fornecidos ou são difíceis de serem realimentados na prática e, por isso, não são utilizados na realimentação.

Os principais objetivos da pesquisa foram o desenvolvimento de uma metodologia para contornar limitações de técnicas tradicionais, em especial, aquelas geradas por sistemas em que nem todos os estados são realimentados e que apresentam atrasos de transporte; o desenvolvimento de técnicas na automação dos procedimentos levantados; a melhoria na eficiência computacional; a avaliação e validação dos resultados obtidos com a aplicação da metodologia; e, finalmente, a obtenção de eficácia e eficiência no atendimento dos requisitos de projeto.

A principal contribuição deste trabalho foi a proposição de um algoritmo que: utiliza índices de desempenho associados ao cumprimento dos requisitos de projeto para realizar uma sintonia automatizada de sistemas de controle com realimentação de estados; utiliza redes neurais artificiais para melhorar sua eficiência computacional; permite a avaliação de virtualmente qualquer requisito de projeto.

Este trabalho está dividido em 6 capítulos. No Capítulo 2 é apresentada a teoria de sistemas dinâmicos e de sistemas de controle automático, abordando, entre outros assuntos, modelagens matemáticas desses tipos de sistemas, comportamentos típicos e os objetivos do controle de sistemas dinâmicos. Em seqüência, a arquitetura de realimentação de estados é apresentada e o método de sintonia LQR para esse tipo de arquitetura é descrito. O Capítulo 3 aborda a teoria de redes neurais artificiais (RNA), descrevendo o comportamento e a modelagem de uma arquitetura típica de *multi-layer perceptron* (MLP). O Capítulo 4 apresenta a metodologia proposta neste trabalho para a sintonia de sistemas de controle do tipo realimentação de estados, com o auxílio de redes neurais artificiais. O Capítulo 5 apresenta um exemplo de aplicação da metodologia, cujo objetivo é a sintonia de um sistema do tipo C* (“C-Star”), utilizado para controle do movimento longitudinal de aeronaves. Nesse capítulo é descrita a implementação do sistema a ser sintonizado e da metodologia proposta, juntamente

com análises sobre os resultados obtidos. Finalmente, o Capítulo 6 apresenta as conclusões do trabalho e algumas sugestões para desenvolvimento de trabalhos futuros.

Capítulo 2

SISTEMAS DE CONTROLE AUTOMÁTICO

2.1 SISTEMAS DINÂMICOS

Sistemas dinâmicos são sistemas que apresentam algum tipo de mudança comportamental no transcorrer de um determinado espaço de tempo (seja o tempo considerado discreto ou contínuo). O número de indivíduos de uma população de bactérias ou o movimento de um astro no espaço sideral são exemplos de sistemas que sofrem alterações à medida que o tempo evolui.

A grande maioria dos sistemas dinâmicos, independentemente de sua natureza (elétricos, mecânicos, biológicos, etc.) pode ser modelada por equações diferenciais e as grandezas de saída (respostas) desses sistemas são, então, obtidas pela resolução de tais equações.

As equações diferenciais modelam um sistema físico e constituem um conjunto de equações matemáticas das leis físicas que regem o sistema.

A abordagem matemática compromete-se entre a representatividade e a complexidade do sistema modelado, sempre balizada na relevância dos aspectos inerentes e essenciais. Se a contribuição de um determinado aspecto físico é irrelevante, ou negligenciável, frente aos demais aspectos, pode-se proceder a uma significativa redução na complexidade do modelo matemático obtido.

Em geral, na abordagem de um novo problema, verifica-se ser desejável construir inicialmente um modelo simplificado, de modo a se ganhar um conhecimento básico e geral para a solução. Posteriormente um modelo matemático mais completo poderá então ser elaborado e utilizado para uma análise mais detalhada do sistema (Ogata, 2003).

Uma vez obtido o modelo matemático, várias ferramentas, analógicas ou digitais, podem ser utilizadas para análise e síntese. Com o avanço tecnológico recente, os computadores passaram a ser largamente utilizados para o estudo e análise de modelos matemáticos, uma vez que possuem alta capacidade de processamento (milhões de instruções por segundo).

Um modelo matemático bastante utilizado são os sistemas lineares, que são aqueles cujas equações do modelo matemático também são lineares. Uma equação diferencial é linear quando seus coeficientes são constantes ou funções da variável independente.

A principal característica de um sistema linear (ou linearizado) é o princípio da superposição, isto é, a resposta global de vários agentes externos (excitação) ao sistema pode ser obtida pelo somatório das respostas das ações individuais de cada agente. Assim, em sistemas lineares, a resposta final para várias entradas ou excitações pode ser considerada individualmente.

Sistemas dinâmicos não lineares podem ser reduzidos a sistemas lineares em um ponto (ou faixa) de operação através do processo de linearização. É importante destacar que a utilização de um modelo linearizado é válida apenas para o ponto (ou faixa) de operação destacado.

Em sistemas de controle, para caracterizar as relações entre entrada e saída de um sistema, funções matemáticas, conhecidas como “função de transferência”, são amplamente difundidas. Tais funções são obtidas pela relação entre as Transformada de Laplace (Guidorizzi, 2002) da saída (resposta) e da entrada (excitação) do sistema, considerando as condições iniciais nulas. Ressalta-se que a função de transferência é utilizada para representação de sistemas lineares invariantes no tempo.

Dessa forma, pode-se representar a dinâmica de um sistema por equações algébricas na variável s . O grau n do polinômio em s no denominador da função de transferência é igual à ordem do sistema e , portanto, diz-se que o sistema é de n -ésima ordem.

A principal vantagem na representação em s é que as equações diferenciais são convertidas em equações algébricas, consideravelmente mais fáceis de serem manipuladas e resolvidas. Entretanto, não é possível observar e/ou controlar todos os fenômenos internos que envolvem o processo de controle.

Além da representação por função de transferência, um sistema físico pode ser modelado pelas variáveis de estado do sistema. Essa abordagem é especialmente interessante para sistemas mais complexos, que podem apresentar múltiplas entradas e múltiplas saídas. A teoria moderna de controle é baseada no método do espaço de estados, que fornece uma representação uniforme e poderosa no domínio do tempo de sistemas multi-variáveis de ordem arbitrária.

As variáveis de estado de um sistema são o menor conjunto de variáveis que consegue determinar o comportamento dinâmico do sistema. Conhecendo seus valores iniciais (em $t = t_0$) e a respectiva entrada do sistema para $t \geq t_0$, é possível determinar toda a dinâmica do sistema a partir das variáveis de estado.

A escolha das variáveis de estado é arbitrária e depende somente de considerações práticas. É importante ressaltar, no entanto, que, por essa razão, nem todas as variáveis de estado são fisicamente mensuráveis.

Se n variáveis de estado são necessárias para descrever um sistema, o vetor $x(t)$, contendo as n variáveis de estado, é conhecido como vetor de estado. Conhecendo-se a entrada $u(t)$ do sistema, para $t \geq t_0$, e o valor inicial $x(t_0)$ do vetor de estados, é possível determinar o estado do sistema $x(t)$ para qualquer instante t .

Um sistema dinâmico linear pode ser representado pelas suas equações de estado, conforme mostrado em (1).

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}\tag{1}$$

As matrizes A , B , C e D são chamadas de matrizes de estado do sistema; $u(t)$ é o sinal de entrada do sistema; $x(t)$ é o vetor de variáveis de estado; $\dot{x}(t)$ é o vetor das derivadas das variáveis de estado e $y(t)$ é a saída do sistema dinâmico.

Um sistema representado por matrizes de estado pode ser também representado por uma função de transferência, conforme demonstrado a seguir.

As transformadas de Laplace (Guidorizzi, 2002) das equações apresentadas em (1) são dadas por (2):

$$\begin{aligned} sX(s) - X(0) &= AX(s) + BU(s) \\ Y(s) &= CX(s) + DU(s) \end{aligned} \quad (2)$$

A função de transferência é dada pela relação entre a Transformada de Laplace da função de saída e a Transformada de Laplace da função de entrada, para condições iniciais nulas. Assim, obtém-se, com um pouco de álgebra matricial, a partir das equações expressas em (2), a relação (3), a seguir.

$$Y(s) = [C(sI - A)^{-1}B + D]U(s) \quad (3)$$

Por fim, tomando-se a relação entre saída e entrada, chega-se à expressão (4), que permite a conversão do conjunto de equações do sistema no espaço de estados para a função de transferência.

$$G(s) = \frac{Y(s)}{U(s)} = C(sI - A)^{-1}B + D \quad (4)$$

Um vez realizado o equacionamento matemático do sistema, seja a partir de funções de transferência, seja no espaço de estados, este é, então, utilizado para as análises das características dinâmicas do sistema.

A análise do sistema dinâmico pode ser relativa à resposta transitória do sistema (aquela que o sistema apresenta até chegar ao seu estado final ou de regime permanente, se houver), ou à resposta estacionária ou de regime (que diz respeito à saída quando t tende ao infinito).

A primeira característica que se pode pensar sobre um sistema dinâmico é a resposta temporal característica para um determinado tipo de entrada ou excitação, como por exemplo, para uma entrada tipo degrau ou uma entrada tipo rampa. Outro ponto que se pode analisar é se essa resposta temporal permanece entre determinados valores máximos e mínimos para cada instante de tempo (envoltória temporal). Outros fatores que podem ser levantados na análise são o máximo sobre-sinal para o sistema (*overshoot*), o erro máximo em regime permanente e o fator de amortecimento, entre outros.

Uma característica muito importante na análise de sistemas dinâmicos diz respeito à sua estabilidade. Diz-se que um sistema é estável se, na presença de uma perturbação, ele consegue retornar ao seu estado de equilíbrio no qual se encontrava. Já em um sistema instável, a oscilação da resposta varia indefinidamente ou a resposta diverge sem limite em relação ao seu ponto de equilíbrio. Existem diversas propriedades que estão ligadas à estabilidade do sistema, como margem de fase e margem de ganho, amortecimento, localização dos pólos, etc.

Os sistemas dinâmicos nem sempre apresentam as características desejadas para uma determinada situação que se está pretendendo. Uma abordagem de aproximação às características desejadas para o sistema dinâmico em estudo (ao qual se dá o nome de planta), é a implementação de um sistema de controle a ser diretamente inserido junto ao sistema original. Um sistema de controle tem como objetivo modificar as características da planta de tal forma a levá-la às características desejadas.

2.2 SISTEMAS DE CONTROLE

Um sistema de controle automático é um sistema que tem por objetivo controlar determinadas características de um sistema dinâmico. Ao se projetar um sistema de controle deve-se poder prever o comportamento do sistema a partir da análise de suas componentes.

Em um projeto de controle direto, o controle é feito através de um bloco chamado controlador. A entrada do sistema, em vez de ser diretamente direcionada à planta, passa inicialmente pelo controlador, que atua em cima do sinal de entrada. Em seguida, a saída do controlador é direcionada para a planta do sistema.

Esse tipo de sistema de controle é conhecido como controle em malha aberta. Na malha aberta não há realimentação da informação de saída para a entrada do sistema, isto é, a saída não interfere no ajuste do controlador. A Figura 2.1 a seguir ilustra o diagrama de blocos de um controlador em malha aberta.

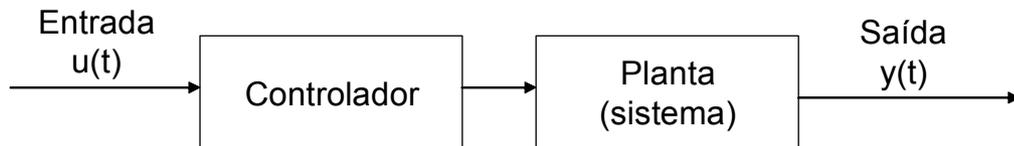


Figura 2.1 – Controlador em malha aberta

Em um controle por realimentação (também conhecido como *feedback*), o sistema de controle passa a utilizar as informações de saída para corrigir as ações implementadas sobre a planta. Nesse tipo de sistema, conhecido como controle em malha fechada, o sinal da saída é realimentado na entrada e comparado com um sinal de referência (*set point*). A diferença entre a saída e a referência é o erro do sistema. Esse erro é então direcionado à entrada do controlador, que atua na planta do sistema baseado em informações da saída desta. A Figura 2.2, mais adiante, apresenta um exemplo de realimentação.

Em controle clássico existem diversas técnicas para o projeto do controlador quando o sistema em geral é descrito por funções de transferência. Já em controle moderno as técnicas de controle são baseadas na representação do sistema no espaço de estados. Dentre as técnicas modernas de controle, a realimentação de estados é muito utilizada.

2.3 REALIMENTAÇÃO DE ESTADOS

Realimentação (empregam-se ainda os termos “retroalimentação”, “retroação” e também o termo em inglês “*feedback*”) é o nome dado ao procedimento através do qual parte do sinal de saída de um sistema (ou circuito) é transferida para a entrada deste mesmo sistema, com o objetivo de diminuir, amplificar ou controlar a saída do sistema. Quando a retroalimentação diminui o nível da saída, tem-se a retroalimentação negativa, e quando a retroalimentação amplifica o nível da saída, tem-se uma retroalimentação positiva. A retroalimentação pode também ter um efeito variável (às vezes positivo, às vezes negativo),

de acordo com as condições, o tempo de transmissão e a inércia do sistema, o que pode provocar efeitos variados (ciclos, comportamento caótico, etc.).

A realimentação de estados pode ser definida como a entrada do sistema em função da combinação linear das variáveis de estado, como apresentado na expressão (5).

$$u(t) = -Kx(t) \quad (5)$$

O vetor K tem dimensão $1 \times n$ (em que n é o número de variáveis de estado do vetor $x(t)$) e contém os ganhos de realimentação do sistema, isto é, os ganhos que irão gerar a entrada $u(t)$ do sistema, ponderando o sinal de sua respectiva variável de estado de $x(t)$.

Aplicando a realimentação de estados apresentada na expressão (5) para o sistema apresentado na equação (1), obtém-se a equação (6), de malha fechada, apresentada abaixo:

$$\dot{x}(t) = (A - BK)x(t), \quad x(0) = x_0 \quad (6)$$

Nos sistemas de controle em que há uma referência a ser seguida, a retroação de estados é subtraída da referência para a composição da nova entrada. A Figura 2.2 mostra o diagrama

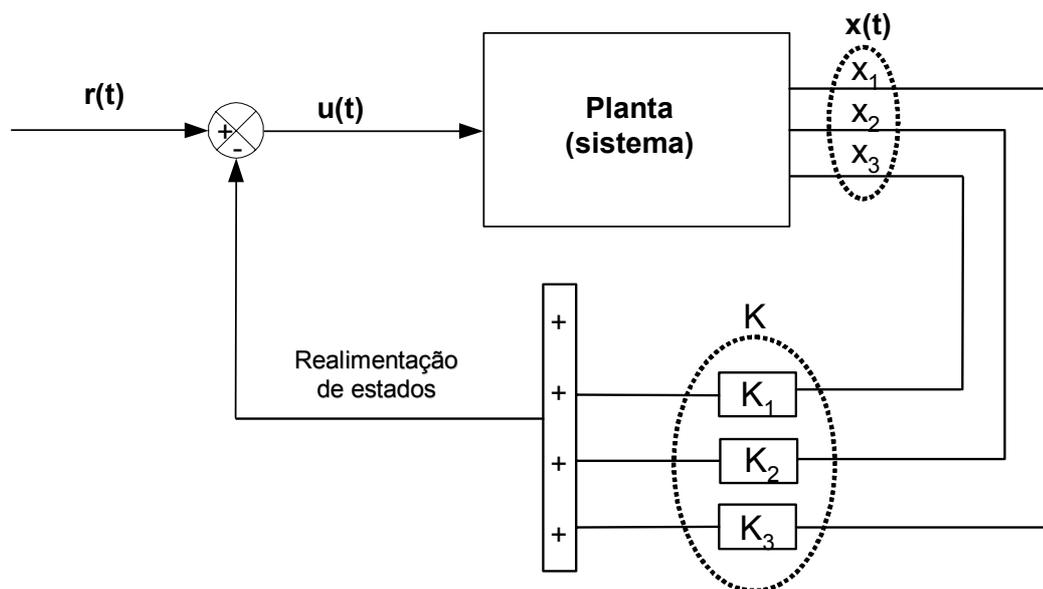


Figura 2.2 – Diagrama de blocos da realimentação de estados

de blocos de um controle baseado em realimentação de estados, com uma referência a ser seguida.

Pode-se empregar também um integrador aplicado ao erro entre referência e saída do sistema, com o objetivo de minimizar o erro em regime permanente (Ogata, 2003). O erro pode ser modelado como um estado adicional do sistema, com a conseqüente alteração das matrizes de estado para incorporar essa nova variável de estado (Philips e Harbor, 1997). Dessa forma, o ganho do integrador pode ser obtido juntamente com os demais ganhos de realimentação do sistema.

Existem diversas técnicas para encontrar os ganhos de realimentação (Gautrey *et al.*, 1997), dentre as quais algumas baseadas em otimização. Em métodos baseados em otimização, uma função custo (índice de desempenho) é definida a partir de um critério de desempenho especificado, e ganhos de realimentação são obtidos de tal forma a minimizar essa função. Cabe ressaltar que a função de custo descrita neste parágrafo não tem qualquer relação com os índices de desempenho (ou custos) propostos na metodologia apresentada neste trabalho.

O conjunto de ganhos que minimiza a função de custo escolhida é, por definição, ótimo. É importante ressaltar ainda que um sistema de controle com um índice de desempenho ótimo para uma determinada condição é, em geral, não ótimo sob outros critérios de desempenho. Além disso, em se tratando de controle ótimo, a aplicação efetiva em um projeto particular pode ser bastante difícil e dispendiosa. Portanto, pode não ter sentido fazer grandes gastos na implementação de um controlador que é ótimo apenas sob condições muito específicas. Um sistema de controle raramente é projetado para realizar um única tarefa especificada totalmente *a priori*. Em vez disso, ele é projetado para realizar uma tarefa selecionada aleatoriamente de um repertório completo de tarefas possíveis. Em sistemas práticos, pode então ser mais aconselhável procurar leis de controle ótimo aproximadas que não estão rigidamente ligadas com um índice de desempenho único. (Ogata, 2003)

O LQR (regulador linear-quadrático) é um método de obtenção de ganhos de realimentação baseado em otimização. O “problema LQR” tenta levar as variáveis de estado, após uma perturbação, de volta à sua condição de equilíbrio ($x(t) \rightarrow x_0$) o mais rápido possível, considerando a variação de energia da entrada do sistema. Mais precisamente, o vetor de ganhos K é escolhido de tal forma a minimizar a função J_{LQR} , definida em (7) (Ogata, 2003).

$$J_{LQR} = \int_0^{\infty} (x^*(t)Qx(t) + u^*(t)Ru(t))dt \quad (7)$$

As matrizes de peso $Q = Q^* \geq 0$ e $R = R^* \geq 0$ são fornecidas. Se os valores de Q são grandes, quando comparados com os valores de R , isso significa que o controle é “barato”, isto é, que a regulação ($x(t) \rightarrow x_0$) é mais importante que o custo de se usar uma quantidade relativamente grande de energia na entrada. Por outro lado, se R é grande em comparação a Q , então o controle da energia é mais importante.

O vetor de ganhos ótimos K é obtido a partir da equação (8), em que P satisfaz a equação de Riccati, apontada pela equação (9).

$$K = R^{-1}B^*P \quad (8)$$

$$A^*P + PA - PBR^{-1}B^*P + Q = 0 \quad (9)$$

A equação de Riccati pode ser resolvida numericamente. Alguns softwares como o MATLAB possuem funções internas que implementam a resolução da equação de Riccati e a obtenção do ganhos de realimentação (MATLAB Control System Toolbox Reference, 2008; Groom e Schaffner, 1990). No ambiente do software citado, tem-se a função implícita $k=lqr(A,B,Q,R)$, cujos parâmetros são as matrizes de estado A e B e as matrizes de peso Q e R , e o retorno é o vetor k , de ganhos ótimos de realimentação .

No entanto, a solução matemática pode produzir respostas não aplicáveis na prática. Além disso, o método LQR pressupõe que os parâmetros do processo sejam constantes, que não haja atrasos de transporte, que todos os estados do sistema possam ser realimentados, que o sistema seja linear, etc. (Ogata, 2003). Como isso nem sempre ocorre, a aplicação dessa técnica é bastante limitada.

Capítulo 3

REDES NEURAIS ARTIFICIAIS

3.1 INTRODUÇÃO

As redes neurais artificiais são modelos matemáticos cujo intuito é simular as capacidades de aprendizagem e inferência do cérebro humano, isto é, o tentar, o errar, o aprender, o descobrir; enfim, adquirir conhecimento. As RNAs apresentam um modelo inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através da experiência. Uma rede neural artificial vale-se de centenas ou milhares de unidades de processamento (neurônios artificiais), enquanto o cérebro de um mamífero pode apresentar bilhões ou trilhões de neurônios biológicos.

3.2 MODELO DE NEURÔNIOS ARTIFICIAIS

O neurônio artificial (perceptron) tenta modelar algumas características do neurônio biológico, desde os estímulos neuronais até a capacidade individual de aprendizado.

A Figura 3.1 mostra o modelo de um perceptron (unidade básica do modelo neuronal), onde x_1 a x_n representam os n terminais de entrada (equivalentes aos dendritos do neurônio biológico) e y_1 a y_m , os m terminais de saída (equivalente aos axônios). Os pesos w_{1j} a w_{nj} são as ponderações nas entradas e representam as conexões sinápticas entre os neurônios. A

função de ativação ou de limiar (*threshold function*) representa a ativação ou inibição de estímulos na saída do neurônio (Chong e Zak, 2001).

Cada um dos sinais de entrada é multiplicado por seu respectivo peso (que pondera o grau de contribuição do estímulo de entrada), cujo somatório final aplicado à função de ativação determinará o nível de atividade do neurônio. Dependendo do nível de atividade resultante, o estímulo é passado à frente (para os neurônios seguintes na cadeia de processamento neural ou para a saída) ou não. As Figuras 3.1 e 3.2 ilustram o modelo de unidade neuronal do tipo perceptron.

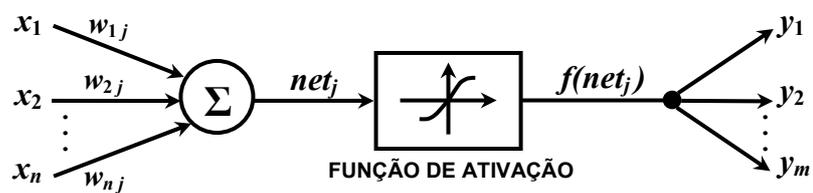


Figura 3.1 – Estrutura de um neurônio artificial (retirada de (Lopes, 2007))

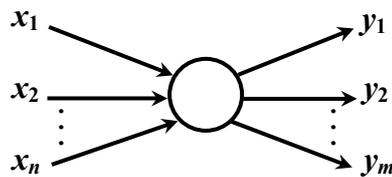


Figura 3.2 – Neurônio artificial estilizado (retirada de (Lopes, 2007))

O somatório de todas as entradas do neurônio j , ponderadas pelos seus respectivos pesos sinápticos, é dado pela expressão (10), abaixo, em que net_j é o estímulo total a ser aplicado na função de ativação do neurônio.

$$net_j = \sum_{i=1}^n x_i w_{ij} \quad (10)$$

Nessa expressão, w_{ij} representa o peso da sinapse do neurônio i para o neurônio j , e o valor resultante net_j (somatório ponderado das entradas) é aplicado à função de ativação, cuja resposta será o nível de atividade de saída do neurônio.

As funções de ativação $f(\cdot)$ mais comuns utilizadas em unidades neuronais artificiais são as do tipo função de *Heaviside* (Figura 3.3(a)), a função sigmoidal (Figura 3.3(b)) e a função linear (Figura 3.3(c)). Para uma função de ativação *threshold* (*Heaviside*), a saída é configurada para um de dois níveis, dependendo se a entrada é maior ou menor do que um valor limite definido (*threshold*). Para uma função de ativação sigmoidal, a saída varia continuamente, mas não linearmente, em função do nível de ativação. Existem vários tipos de função sigmoidal, como a sigmoidal logística e a tangente hiperbólica. Finalmente, para uma função de ativação linear, a saída é proporcional ao valor do nível de ativação.

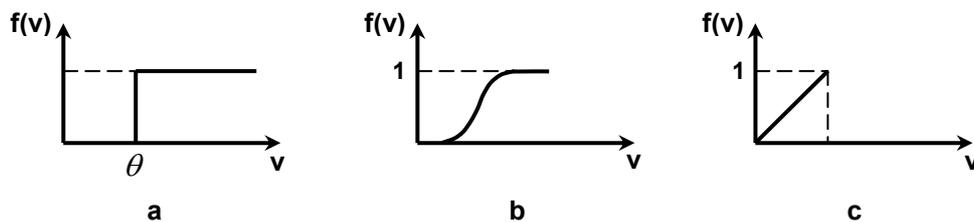


Figura 3.3 – Funções de ativação: (a) *Heaviside*; (b) sigmoidal; (c) linear.
(retirada de (Lopes, 2007))

A função mais utilizada na construção de redes neurais artificiais é a sigmoidal, em especial a do tipo tangente hiperbólica (Haykin, 1999).

3.3 ARQUITETURA DE REDES NEURAIS ARTIFICIAIS

Uma RNA é formada por perceptrons interconectados. As entradas de um perceptron podem ter origem tanto nos nós de entrada como também nas saídas de outros perceptrons ligados a ele. Já a saída do perceptron é sempre uma função de suas próprias entradas.

Os perceptrons podem ser interconectados de diferentes formas, dando origem a diferentes topologias de rede (Rao *et al.*, 2005). As topologias podem ser definidas pelo número de camadas, quantidade de neurônios nas camadas e pelo tipo de conexão entre os neurônios.

Dentre os tipos de configurações existentes, a rede pode ser do tipo *feedforward* (direta) ou do tipo *feedback* (com retro-alimentação). Nas redes do tipo *feedforward*, os perceptrons

são interconectados em camadas e o fluxo de dados ocorre em apenas uma direção (da entrada para a saída) (Chong e Zak, 2001). Nas redes neurais do tipo *feedback*, há pelo menos um ponto de realimentação, isto é, um neurônio recebe informação tanto de neurônios de camadas anteriores quanto posteriores. A primeira camada na rede é denominada camada de entrada e a última é chamada de camada de saída; as possíveis camadas existentes entre estas duas são conhecidas como camadas intermediárias ou ocultas.

Problemas mais complexos podem demandar a utilização de mais camadas intermediárias e/ou a utilização de mais perceptrons por camada. Isso, porém, torna o aprendizado da rede mais lento, já que uma quantidade significativa de cálculos matemáticos é exigida.

A Figura 3.4 ilustra uma rede neural do tipo *feedforward*, com três neurônios na camada de entrada, duas camadas intermediárias, e dois neurônios na camada de saída. Observa-se a intrincada interconexão entre os neurônios, formando uma rede de processamento altamente distribuída e com forte paralelismo entre as unidades perceptrons.

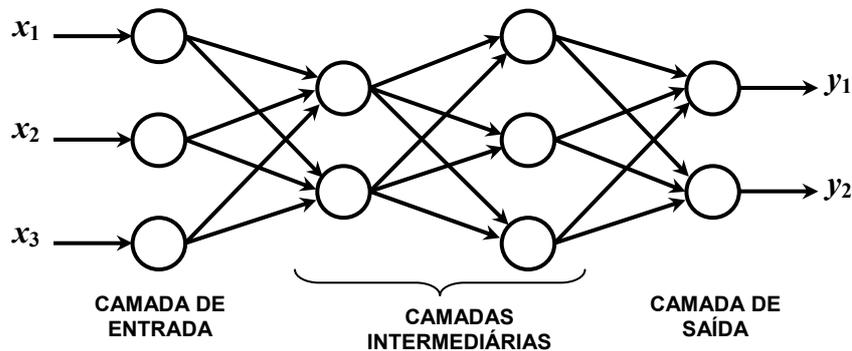


Figura 3.4 – Rede neural do tipo *feedforward*

Em relação às funções de ativação, na prática, a função escolhida é quase sempre não-linear do tipo sigmoidal logística ou do tipo tangente sigmoidal, uma vez que camadas intermediárias com funções de ativação lineares simplesmente podem ser reduzidas a uma única camada, pois os cálculos puramente lineares podem ser reduzidos em apenas uma operação (Beale e Jackson, 1990).

3.4 TREINAMENTO DE UMA REDE NEURAL ARTIFICIAL

O treinamento de uma RNA é um processo iterativo de ajuste dos pesos sinápticos dos perceptrons. Uma rede conclui seu aprendizado quando, depois de processar uma certa quantidade de dados de entrada, pertencentes ao domínio do problema, consegue ter uma estrutura final que conduz a uma generalização do problema proposto, isto é, quando a rede consegue conduzir valores de entrada ainda não apresentados na fase de treinamento, porém pertencentes ao domínio do problema, a valores de saídas satisfatórios dentro uma tolerância admitida.

Existem diversos algoritmos para a realização do treinamento de uma rede neural. Eles diferem-se entre si principalmente pelo modo como os pesos são modificados. Quando se utilizam valores conhecidos ou desejados de saída, indicando à rede uma resposta esperada do problema, o aprendizado é dito supervisionado. Neste tipo de treinamento, o vetor de respostas esperadas da saída é previamente conhecidos. Por outro lado, quando não há conhecimento prévio da resposta esperada, diz-se que o aprendizado é não-supervisionado.

Uma configuração de rede bastante utilizada é a do tipo *backpropagation*. Essa configuração usa a topologia *feedforward* e o algoritmo de aprendizado *backpropagation*. Para esse algoritmo de aprendizado supervisionado, as saídas desejadas são parte importante na fase de treinamento e permitirá o ajuste dos pesos através do processo de retropropagação do erro (valor esperado menos valor fornecido pela rede). Na fase de treinamento, este algoritmo opera em uma seqüência de dois passos. Primeiro, um sinal é apresentado à camada de entrada da rede e este sinal é propagado através da rede até que uma resposta seja produzida pela camada de saída. No segundo passo, ocorre a fase de adaptação da rede, quando a saída obtida pela rede é comparada à saída desejada para o sinal de entrada, produzindo um erro. Este último é retropropagado através da rede para o ajuste dos pesos entre as camadas para produzir a saída correta ou dentro de uma tolerância especificada (Bigus, 1996).

Para as unidades na saída, ajustar o peso é relativamente simples. Contudo para as unidades intermediárias, o ajuste não é tão óbvio. Pode-se deduzir que as unidades intermediárias ligadas a saídas de maior erro precisam de ajustes maiores em seus pesos, enquanto unidades intermediárias ligadas a saídas de menor erro sofrerão ajustes menores. Matematicamente se mostra que os pesos para um determinado nó particular deve ser ajustado

na proporção direta ao erro das unidades às quais está ligado. Assim, retropropagar o erro pela rede permite o ajuste dos pesos entre as camadas, permitindo a redução local e global do erro e que a rede de fato generalize ou aprenda (Beale e Jackson, 1990).

Capítulo 4

SINTONIA DE GANHOS DE RETROAÇÃO DE ESTADOS COM O AUXÍLIO DE REDES NEURAIS ARTIFICIAIS

4.1 INTRODUÇÃO

Projetar um sistema de controle automático pode ser uma tarefa bastante complexa, dependendo da planta que se quer controlar, da acessibilidade às informações da planta e dos requisitos de projeto.

Na escolha da arquitetura do sistema de controle a ser utilizado é necessário um certo grau de conhecimento de características da planta e dos requisitos de comportamento imposto ao sistema. Uma vez decidida a arquitetura, é necessário fazer a sintonia da malha de controle visando a atender os requisitos de projeto.

Um sistema de controle bastante utilizado na prática faz uso da retroação de estados, que utiliza como entrada do sistema uma combinação linear das variáveis de estado. A sintonia desse sistema consiste em determinar os ganhos de realimentação para as variáveis de estado.

Existem algumas técnicas clássicas utilizadas para se calcular os ganhos de um sistema de controle com realimentação de estados, sendo a técnica LQR (Ogata, 2003), por exemplo, largamente explorada na literatura.

O método LQR consiste na obtenção dos ganhos por intermédio da otimização matemática de uma função de custo. No entanto, conforme exposto na seção 2.3, esse método possui uma série de restrições de ordem prática, o que torna a aplicação dessa técnica bastante limitada.

Este trabalho propõe uma metodologia alternativa para a sintonia de malhas de controle com realimentação de estados, voltada, em especial, para sistemas que apresentam atrasos de transporte e estados difíceis de serem mensurados e observados na prática, e que não serão utilizados na realimentação. O objetivo da metodologia proposta é, de forma automatizada, encontrar uma sintonia que atinja determinados requisitos mínimos de projeto, especificados pelo projetista, sem necessariamente atingir uma solução ótima.

Entenda-se por metodologia automatizada aquela que automatiza grande parte das tarefas, necessitando, entretanto, de algumas intervenções humanas, o que se contrapõe ao que seria uma metodologia automática, em que todos os procedimentos seriam realizados pelo computador, sem a necessidade de intervenção humana.

4.2 VISÃO GERAL

A metodologia proposta para a sintonia de um sistema de controle com realimentação de estados é dividida em várias etapas, conforme apresentado na Figura 4.1 e descrita sucintamente a seguir.

Utiliza-se, inicialmente, uma versão simplificada do modelo do sistema a ser controlado, que elimina possíveis restrições à aplicação da técnica LQR. A esse modelo simplificado aplica-se o método LQR, obtendo-se um conjunto de ganhos de realimentação para o sistema. Tal conjunto obtido pode não ser adequados ao controle do sistema original, mas é um bom ponto de partida para se determinar um conjunto de ganhos que sintonize o sistema original de forma satisfatória.

Uma vez obtido o conjunto de ganhos inicial, faz-se uma varredura de valores em torno de cada valor de ganho. Isto é, para cada ganho do conjunto inicial, são selecionados alguns valores dentro de um intervalo pré-determinado em torno do ganho. A partir da combinação

de valores gerados na varredura, obtém-se uma série de conjuntos de ganhos a serem testados e avaliados no projeto de controle da planta original.

Simula-se o sistema completo para cada uma dessas combinações e faz-se uma análise automatizada das respostas do sistema, estabelecendo um índice de desempenho (custo) associado a cada uma dessas combinações. Esse valor é calculado para quantificar o quanto os requisitos de projeto pré-estabelecidos não são atendidos. Com essa série de conjuntos de ganhos e seus respectivos índices de desempenho associados (parciais e total), a rede neural é então treinada.

Uma vez que o processo de treinamento da rede neural esteja finalizado, apresenta-se à rede uma quantidade maior de novos conjuntos de ganhos, obtidos por uma nova varredura de valores em torno do conjunto inicial, e obtém-se como resposta da rede os valores de índice de desempenho estimados para cada conjunto.

Os melhores casos (conjuntos de ganhos que apresentaram os menores custos totais estimados) são selecionados e então validados, a partir da simulação do sistema para cada um deles e cálculo dos custos (índices de desempenho) associados a cada um.

Dessa segunda etapa de simulações, selecionam-se os melhores casos (os conjuntos de ganhos que apresentaram o menor custo associado) para se fazer uma revisão de projeto (*design review*), isto é, verificar se dentre os melhores casos há pelo um que atenda aos requisitos de projeto. Acredita-se que, dentre esses casos, pelo menos um sintonizará o sistema de controle de forma satisfatória, atendendo aos requisitos de projeto.

A Figura 4.1 apresenta um fluxograma ilustrativo com as etapas da metodologia proposta.

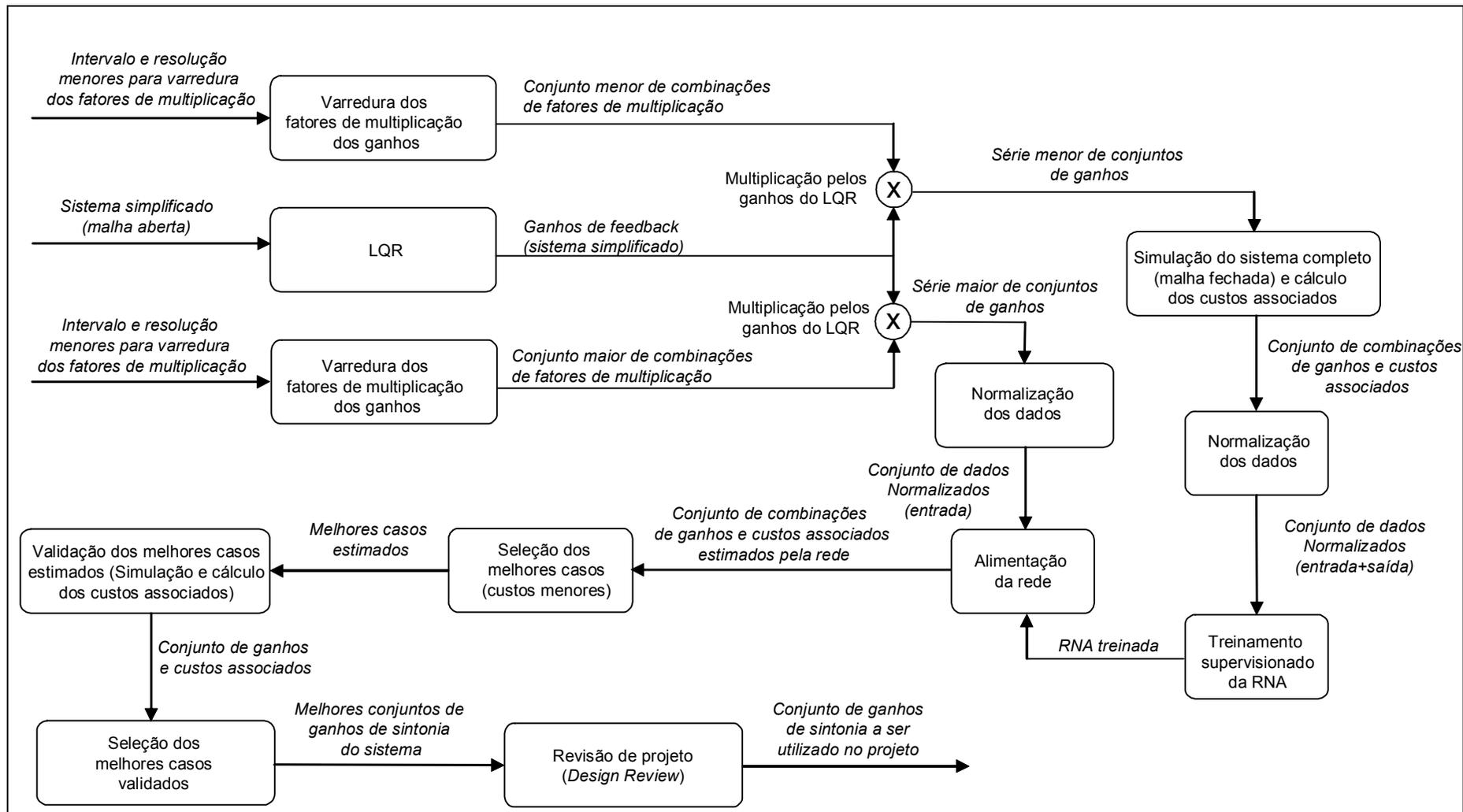


Figura 4.1 – Fluxograma das etapas da metodologia

4.3 DETALHAMENTO DA METODOLOGIA

4.3.1 Requisitos de projeto

O passo inicial para a sintonia em um projeto de sistema de controle é a definição de quais requisitos o projeto deve atender. Uma vez definidos, deve-se determinar qual metodologia utilizar para avaliar o cumprimento ou não de tais requisitos.

Os requisitos de projeto em geral estão relacionados ao comportamento dinâmico do sistema. Exemplos desses requisitos são frequência de cruzamento de ganho aproximada; amortecimento mínimo esperado para os pólos do sistema; margem de ganho e margem de fase mínimas desejadas; resposta temporal do sistema a uma excitação (entrada) pré-definida, (tipo degrau, impulso ou rampa, por exemplo) limitada a valores máximos e mínimos a cada instante; etc.

Ressalta-se que, para a metodologia proposta, os requisitos em geral não impõem uma dada solução particular, mas sim um conjunto de soluções, como por exemplo, sintonias cuja margem de fase resultante é maior ou igual a uma margem de fase mínima desejada; ou sintonias cujo sistema resultante apresente pólos dominantes com um amortecimento maior que um amortecimento mínimo desejado.

4.3.2 Avaliação do atendimento aos requisitos de projeto

Uma vez definidos os requisitos de projeto, é necessário definir formas de quantificar o atendimento (ou não) destes. O cumprimento ou não dos requisitos de projeto pode ser avaliado com a ajuda de diversos diagramas e gráficos propostos na literatura, como o Diagrama de Bode, o Diagrama de Nichols, Diagrama de Pólos e Zeros, gráfico de resposta temporal, etc..

As expressões utilizadas para o cálculo dos índices de desempenho (custos) associados às violações dos requisitos podem ser escolhidas com certa flexibilidade pelo projetista. Contudo, deve-se levar em conta o princípio básico de que os custos devem aumentar à medida que determinada característica avaliada se afasta de um requisito de projeto.

Este ponto apresenta uma grande vantagem para a metodologia proposta em relação às metodologias tradicionais clássicas ou de controle moderno. Em um método tradicional, o projeto do sistema de controle é baseado em um conjunto restrito de características a serem impostas para o sistema. Já na metodologia proposta, qualquer característica pode ser estabelecida como requisito, desde que uma expressão matemática permita quantificar ou mensurar adequadamente a adequação ou cumprimento ao critério adotado.

Uma forma de se quantificar o quanto os requisitos de projeto estão sendo satisfatoriamente observados é estabelecer expressões matemáticas que relacionem as informações contidas nos diagramas e gráficos citados na literatura para análise de sistemas dinâmicos (mencionados anteriormente) e valores numéricos que indiquem o quão distantes de características desejadas se está para o ponto observado.

A partir do diagrama de Nichols, por exemplo, é possível avaliar as margens de fase e de ganho. Um índice de desempenho pode ser associado ao não-cumprimento das margens desejadas e, também, à possibilidade de um distúrbio alterar significativamente as margens de fase e de ganho, reduzido-as abaixo de um limite desejável.

O diagrama de Bode permite verificar as frequências de cruzamento de ganho e, através das informações contidas neste diagrama, associar um valor numérico (custo) para o afastamento da frequência de cruzamento de ganho desejada. Também é possível atribuir um custo à curva de resposta em frequência para a qual a ocorrência de um distúrbio possa alterar significativamente tal frequência de cruzamento de ganho.

No diagrama de pólos e zeros é possível verificar se os pólos dominantes estão com um amortecimento maior ou igual ao amortecimento mínimo esperado e, portanto, estabelecer um índice de desempenho para aqueles pólos que tiverem amortecimento menor do que o desejado.

Calculado um índice de desempenho para cada requisito de projeto (custos parciais), estes devem ser agrupados em um único índice de desempenho global (custo total). Para tornar isso possível, há necessidade de se normalizar os valores de cada critério de desempenho individualmente a partir de uma base comum entre eles, de tal forma a garantir o somatório de índices de desempenhos de diferentes origens.

Não é uma tarefa trivial definir uma base comum para a normalização dos índices adotados, já que esta depende, primeiramente, de como o projetista determinou e adotou esse

e aqueles critérios para avaliação do requisito de desempenho. Além disso, não existe uma relação firmada na literatura que possa relacionar determinada violação em um requisito a uma outra violação qualquer em outro requisito, como por exemplo, uma relação que atribua a violação de 0,1 no amortecimento de um pólo à variação de 0,5 rad/s na frequência de cruzamento de ganho. Na prática, o projetista observa os diagramas e avalia, de acordo com seus conhecimentos, o quanto uma determinada variação em relação à especificação inicial é importante ou não.

O projetista pode estar disposto a ceder, em partes, em um dado requisito de projeto (por exemplo, a margem de fase de um dado estado do sistema estar um pouco abaixo do esperado), em face ao cumprimento global de desempenho dos demais requisitos. Com uma ponderação entre índices de desempenho, o projetista diz ao algoritmo quais são os requisitos em que ele está mais disposto ou menos disposto a ceder, em prol de uma sintonia geral.

Assim, a normalização e a ponderação dos índices de desempenho para cada requisito avaliado (custos parciais), que compõem o custo total final, ficam a cargo do projetista, de acordo com o cálculo dos índices de desempenho e da relevância de cada requisito para o projeto. Essa é outra grande vantagem em relação às metodologias tradicionais, já que, além de poder estipular uma gama de requisitos diferentes a serem abordados, o projetista pode determinar a relevância de cada um deles e o quanto está disposto a ceder em cada um.

Essa forma empírica, baseada na experiência do projetista, é uma maneira prática de se obter as ponderações. Simulando algumas condições que representem a atuação do indivíduo na elaboração do projeto de controle de sistemas, o projetista vai ajustando as ponderações, de tal forma a se atingir valores de custo coerentes com a situação apresentada.

4.3.3 Obtenção de um ponto de partida para a sintonia

Inicialmente, para a sintonia do sistema de controle, deve-se obter um conjunto inicial de ganhos de realimentação, em cima do qual o algoritmo proposto será aplicado. É importante que este ponto de partida faça com que a sintonia não esteja muito longe daquela esperada.

As metodologias clássicas, como LQR e alocação de pólos com realimentação de estados, apresentam algumas limitações e, por isso, nem sempre podem ser utilizadas em

sintonia de um sistema de controle. No entanto, quando aplicadas ao modelo reduzido de um sistema, eliminando possíveis restrições à aplicação dessas metodologias, pode-se obter um bom ponto de partida para a sintonia final do sistema original.

Um ponto de partida pode ser obtido por tentativa e erro, testando algumas combinações dos parâmetros para a metodologia tradicional e, então, utilizando-se no sistema original os ganhos resultantes, observando o quão perto dos requisitos de projeto encontram-se as características do sistema sintonizado.

Neste trabalho, optou-se por utilizar o método LQR na obtenção de um conjunto inicial de ganhos para a sintonia, já que, ao se aplicar o método LQR, é necessário apenas definir as matrizes de custo R e Q, sendo que uma delas é pode ser definida como a matriz identidade e a outra definida como a matriz identidade multiplicada por um escalar. Ressalta-se que para a aplicação do método não se faz necessária a determinação analítica das matriz R e Q, considerando rigorosamente uma formulação matemática estrita das características de projeto pretendidas. O que se espera é que seja possível alcançar um ponto de partida razoável em poucas tentativas.

Conforme mencionado anteriormente, o método LQR necessita que todos os estados estejam disponíveis para realimentação e que não haja atrasos de transporte. No entanto, em muitos casos, os estados não são facilmente mensuráveis ou observáveis na prática.. Além disso, os sistemas muitas vezes possuem atraso de transporte não desprezíveis. Portanto, para tornar possível a aplicação do método LQR, utiliza-se um modelo simplificado do sistema, que contém apenas os estados de fato utilizados na realimentação e que ignora quaisquer atrasos de transporte.

Ao sistema simplificado (em malha aberta), aplica-se o processo LQR, para obtenção dos ganhos de realimentação, representados pelo vetor K_{LQR} em (11).

$$K_{LQR} = [K1_{LQR} \quad K2_{LQR} \quad \dots \quad Kn_{LQR}] \quad (11)$$

4.3.4 Varredura de ganhos em torno do ponto de partida

A partir do conjunto inicial de ganhos obtido, pode-se obter um conjunto K' , definido em (13), que fornece valores de ganhos para a sintonia do sistema original. Tal conjunto é formado pelo produto, termo a termo, entre o conjunto inicial K_{LQR} , definido em (11), e um conjunto de fatores multiplicativos $K_{K_{LQR}}$, definido em (12).

$$K_{K_{LQR}} = [K_{K_1} \quad K_{K_2} \quad \dots \quad K_{K_n}] \quad (12)$$

$$K' = [K_{K_1} \cdot K_{1_{LQR}} \quad K_{K_2} \cdot K_{2_{LQR}} \quad \dots \quad K_{K_n} \cdot K_{n_{LQR}}] \quad (13)$$

Uma vez obtido o conjunto de ganhos de realimentação de estados através da aplicação do método LQR para o sistema simplificado (ponto de partida da sintonia), deve-se encontrar pelo menos um conjunto de fatores $K_{K_{LQR}}$ cuja sintonia resultante atenda satisfatoriamente o sistema original.

Um modo de se obter tal conjunto de fatores é, para cada um dos n fatores K_{K_i} do conjunto $K_{K_{LQR}}$, fazer uma varredura dos valores dentro de um intervalo pré-estabelecido. Adotado um intervalo para cada fator, estes são varridos com uma resolução r , isto é, são discretizados r valores dentro do intervalo. Em seguida, são feitas combinações entre os valores obtidos pela varredura, tomando-se um valor de cada intervalo. Obtém-se, com isso, uma série de r^n combinações de conjuntos de fatores de multiplicação $K_{K_{LQR}(i)}$, que levam a r^n conjuntos de ganhos $K'_{(i)}$.

A varredura de um fator de multiplicação de ganho K_{K_i} , ilustrada em (14), é feita em um intervalo de valores definidos, expresso em (15), com uma resolução r . O primeiro termo $K_{K_i(1)}$ da seqüência de varredura é o limite inferior do intervalo de varredura, enquanto o último termo $K_{K_i(r)}$ é o limite superior, conforme expresso em (16) e (17), respectivamente.

Como se trata de um fator multiplicador, a varredura no intervalo é feita de forma exponencial, com uma razão incremental q , calculada a partir dos extremos do intervalo e da resolução r , conforme a relação apontada em (18).

O i -ésimo termo da varredura do fator K_{K_i} é obtido de acordo com a expressão (19).

$$K_{Ki} = \{K_{Ki(1)} \quad K_{Ki(2)} \quad \dots \quad K_{Ki(i)} \quad \dots \quad K_{Ki(r)}\} \quad (14)$$

$$\text{Intervalo} = [K_{Ki\min} \quad K_{Ki\max}] \quad (15)$$

$$K_{Ki(1)} = K_{Ki\min} \quad (16)$$

$$K_{Ki(r)} = K_{Ki\max} \quad (17)$$

$$q = \sqrt[r-1]{\frac{K_{Ki\max}}{K_{Ki\min}}} \quad (18)$$

$$K_{Ki(i)} = K_{Ki\min} \cdot q^{i-1} \quad (19)$$

Como exemplo, admita-se que, para um sistema com três ganhos $K1_{LQR}$, $K2_{LQR}$ e $K3_{LQR}$, obtidos pelo processo LQR, deseja-se fazer a varredura no intervalo entre a metade de seu valor a até duas vezes esse valor, conforme intervalo apresentado em (20), e, por simplicidade, adotam-se os mesmos intervalos para os três fatores. Tomando-se três valores nesse intervalo, conforme ilustrado em (21), obtêm-se os valores possíveis para cada fator, mostrados em (22).

$$\text{Intervalo} = [0,5; 2,0] \quad (20)$$

$$r = 3 \quad (21)$$

$$\begin{aligned} K_{K1} &= \{0,5 \quad 1,0 \quad 2,0\} \\ K_{K2} &= \{0,5 \quad 1,0 \quad 2,0\} \\ K_{K3} &= \{0,5 \quad 1,0 \quad 2,0\} \end{aligned} \quad (22)$$

Fazendo a combinação desses valores, têm-se 27 combinações de conjuntos de fatores K_{Ki} , fornecendo 27 conjuntos de ganhos, conforme exemplificado em (23):

$$\begin{aligned} K_{K_{LQR}(1)} &= \{0,5 \quad 0,5 \quad 0,5\} \\ K_{K_{LQR}(2)} &= \{0,5 \quad 0,5 \quad 1,0\} \\ K_{K_{LQR}(3)} &= \{0,5 \quad 0,5 \quad 2,0\} \\ K_{K_{LQR}(4)} &= \{0,5 \quad 1,0 \quad 0,5\} \\ K_{K_{LQR}(5)} &= \{0,5 \quad 1,0 \quad 1,0\} \\ K_{K_{LQR}(6)} &= \{0,5 \quad 1,0 \quad 2,0\} \\ &\dots \\ K_{K_{LQR}(27)} &= \{2,0 \quad 2,0 \quad 2,0\} \end{aligned} \quad (23)$$

Para a combinação de fatores $K_{K_{LQR}} = \{0,5 \quad 2,0 \quad 1,0\}$, por exemplo, tem-se o seguinte conjunto de novos ganhos, ilustrado em (24):

$$K' = [0,5 \cdot K1_{LQR} \quad 2,0 \cdot K2_{LQR} \quad 1,0 \cdot K3_{LQR}] \quad (24)$$

4.3.5 Sintonia utilizando os conjuntos de ganhos obtidos pela varredura

Para cada conjunto de ganhos de realimentação obtido pela varredura, sintoniza-se o sistema completo, em malha fechada, utilizando esse conjunto de ganhos, simula-se o sistema e obtém-se o índice de desempenho associado à violação dos requisitos, conforme a metodologia de avaliação discutida na seção 4.3.2.

A quantidade de combinações a serem testadas e avaliadas pode se tornar elevada, com alto esforço computacional de projeto. Admita-se que o sistema a ser sintonizado tenha 5 ganhos e que cada fator K_{Ki} seja combinado com uma resolução r de 8 valores possíveis. Tem-se então 32768 (8^5) combinações possíveis que devem ser testadas e seu respectivo custo associado calculado. Agora, se cada simulação e cálculos demande 0,5 segundo para ser executado, apenas esse passo demandaria 4,6 horas de esforço computacional. Ressalta-se, assim, a necessidade de se desenvolver alguma forma de otimizar a busca e reduzir o esforço computacional envolvido.

4.3.6 Otimização do algoritmo através de uma rede neural artificial

Uma solução factível para se obter uma otimização do desempenho computacional é a utilização de uma rede neural artificial (RNA), treinada com as informações advindas de uma série inicial de simulações feitas com determinadas combinações de conjuntos de ganhos, para se estimarem os custos das demais combinações.

Uma RNA do tipo MLP (*multi-layer perceptron*) é computacionalmente simples de ser implementada e tem excelente capacidade de generalização a partir de um conjunto de dados representativos do problema em estudo (Haykin, 1999).

Como o custo é uma medida arbitrária, que tem objetivo de dar uma noção de quão perto se está dos requisitos de projeto, não é necessária uma precisão grande na estimativa. Assim, uma rede neural tipo MLP é uma alternativa interessante para estimar os índices de desempenho associados aos parâmetros de sintonia e diminuir consideravelmente o esforço computacional demandado pelo algoritmo.

O treinamento da rede é feito com os dados gerados pela primeira etapa de simulação. A entrada da rede é o conjunto de ganhos de realimentação e a saída são os custos (parciais e total) associados a esse conjunto.

A Figura 4.2 ilustra a arquitetura genérica da rede neural proposta. A rede neural é treinada com validação cruzada (Haykin, 1999; Fahlman, 1988), isto é, a rede é treinada e validada sequencialmente para cada conjunto de padrões apresentado em cada iteração ou época. Com esse propósito, o conjunto de dados iniciais é dividido entre vetores de treinamento e vetores de validação. Antes de tal divisão, no entanto, os dados são normalizados e embaralhados, para que a rede consiga adquirir conhecimento e generalização adequadas a partir das informações fornecidas (Haykin, 1999).

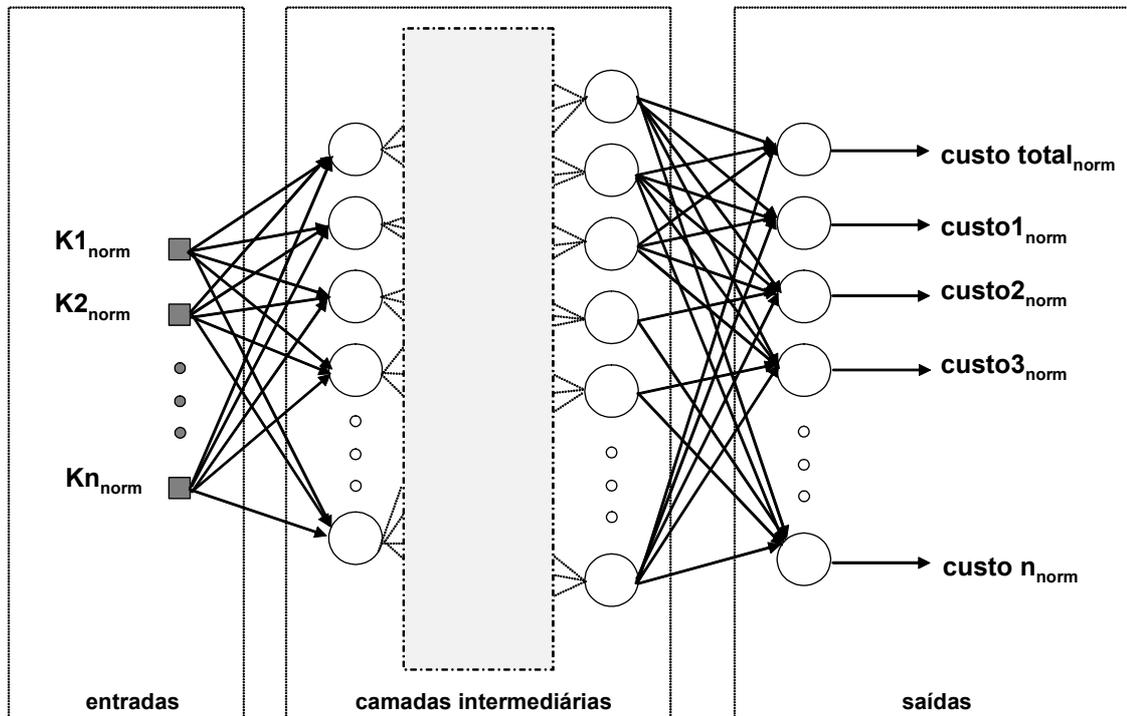


Figura 4.2 – Arquitetura da rede neural tipo MLP proposta

4.3.7 Utilização da rede neural artificial

Após as fases de treinamento e validação, a rede neural MLP está pronta para fornecer estimativas para os custos de novas combinações de ganhos apresentadas a ela. Assim, gera-se uma nova série de conjuntos de fatores $K_{K_{LQR}}$, conforme discutido na seção 4.3.4, mas desta vez com uma resolução e/ou um intervalo de varredura maior, aumentando a chance de se encontrar um conjunto de ganhos que satisfaça a sintonia desejada.

Essa nova série de conjuntos de ganhos é normalizada e então apresentada à rede neural, que fornece uma estimativa de custos associados para cada conjunto. Nesse processo, os vetores de entrada da rede neural são os conjuntos de ganho gerados e normalizados e os vetores de saída são as estimativas da rede para os custos normalizados (parciais e total) de cada conjunto apresentado.

Selecionando os melhores casos, ou seja, aqueles que apresentam os menores custos totais, segundo estimativa da rede, tem-se uma coleção de candidatos à solução do problema. Na seleção dos melhores valores, adota-se um critério prévio de seleção, por exemplo, aqueles que tiverem um custo total associado menor que um dado valor.

4.3.8 Validação dos melhores casos e revisão de projeto (*design review*)

Para verificar quais os valores estimados pela rede são realmente soluções satisfatórias, deve-se simular o sistema para cada um desses candidatos e obter o custo associado a cada um, conforme discutido na seção 4.3.5.

Dentre os casos simulados, são selecionados aqueles que apresentam custos satisfatórios dentro de um critério previamente definido para seleção de candidatos. Espera-se que, dentre esses casos, exista pelo menos um conjunto de ganhos que seja satisfatório para a sintonia do sistema e atendam finalmente aos requisitos de projeto.

A partir dessa etapa, o projetista realiza uma revisão de projeto (ou *design review*, termo em inglês comumente utilizado para esse procedimento), isto é, repassa os melhores casos selecionados pelo algoritmo e faz uma análise crítica em cima desses casos, selecionando o conjunto de ganhos de sintonia a ser utilizado no projeto do sistema.

Caso o custo de uma determinada solução selecionada nesta etapa final esteja dentro de um valor adequado, segundo critério levantado pelo projetista e, mesmo assim, os requisitos de projeto não sejam atendidos de forma satisfatória, há necessidade de se avaliar novamente as expressões utilizadas no cálculo do custo associado. Deve-se verificar quais situações não estão sendo contempladas pelas expressões adotadas e então modificá-las e/ou acrescentar novas expressões, de forma a se contemplar as situações não cobertas.

Caso os custos para as soluções propostas pelo algoritmo estejam coerentes, porém excessivamente altos, segundo o julgamento do projetista, uma revisão dos intervalos de varredura, com novas resoluções, pode aumentar a chance de se encontrar uma solução satisfatória de projeto.

Cabe ainda ressaltar que se o algoritmo está corretamente configurado, de acordo com os requisitos de projeto, e ainda assim a obtenção de uma solução satisfatória não está sendo possível, deve-se verificar se não se está tentando impor requisitos de projeto rígidos demais e difíceis de serem alcançados. Assim, cabe ao projetista viabilizar alternativas para se adequar desempenho com flexibilidade de projeto.

Capítulo 5

EXEMPLO DE APLICAÇÃO, RESULTADOS E DISCUSSÕES

5.1 SISTEMA C* PARA CONTROLE DE AERONAVES

5.1.1 Introdução

Um sistema de controle automático típico de uma aeronave é o sistema que controla seu movimento longitudinal, isto é, o movimento de inclinação do nariz da aeronave para cima e para baixo. Para esse sistema, pode-se utilizar uma malha de controle com retroação de estados com referência a ser seguida.

Em um sistema de controle de movimento longitudinal mais simples, a variável de controle (referência) pode ser definida como o ângulo do profundor, superfície de comando que controla o movimento longitudinal do avião.

Já o sistema C* (“C-Star”) estipula uma variável de controle dada por uma combinação entre a aceleração normal do avião sentida pelo piloto ($N_{z_{\text{piloto}}}$) e a velocidade angular longitudinal da aeronave (q), conhecida pelo termo em inglês *pitch rate*. Essa variável de controle C* é calculada pela expressão (25).

$$C^* = K_{Nz} \cdot Nz_{piloto} + K_{qc} \cdot q \quad (25)$$

O comportamento físico dessas duas variáveis é sentido diretamente pelo piloto, facilitando, assim, a sensibilidade deste no comando do movimento longitudinal da aeronave.

Essa arquitetura foi inicialmente desenvolvida pela Boeing nos anos 60 para aviões militares (Tobie *et al.*, 1966) e utilizada a partir da década de 80 pela Airbus em aviões comerciais (Airbus, 1988; Favre, 1994), o que mostra sua relevância. Resultados práticos (Field, 1993; Siqueira e Moreira, 2007) mostraram a eficiência desse tipo de sistema para o controle longitudinal de aeronaves.

O sistema C^* foi escolhido como exemplo de aplicação para a metodologia proposta neste trabalho por apresentar restrições que dificultam a aplicação de técnicas tradicionais de sintonia de ganhos de realimentação de estados, o que torna essa tarefa desafiadora.

5.1.2 Modelagem

Uma implementação básica do sistema C^* – existem variações (Field, 1993; Najmabadi *et al.*, 1999) – é um sistema de controle com realimentação de estados que possui uma referência a ser seguida. Tal referência de controle é dada por uma combinação linear de duas variáveis físicas do sistema: Nz_{piloto} (aceleração normal, sentida pelo piloto) e q (pitch-rate).

Nessa arquitetura, além da realimentação de estados, emprega-se também um integrador, aplicado ao erro entre referência e saída do sistema, e um ganho de *feedforward*. Além de minimizar o erro em regime permanente, o integrador tem a função de reter a atitude (ângulo de arfagem) após a retirada da referência. Já o ganho de *feedforward* é usado para melhorar a resposta em baixa frequência da malha de controle resultante (Siqueira e Moreira, 2007). A esse ganho geralmente é atribuído um valor fixo, previamente estipulado, de acordo com o quanto se deseja transferir do sinal da referência do sistema diretamente para a entrada da planta.

A Figura 5.1 mostra o diagrama de blocos da implementação da malha de controle utilizada neste trabalho. A Figura 5.2 apresenta o diagrama de blocos de malha aberta desse sistema.

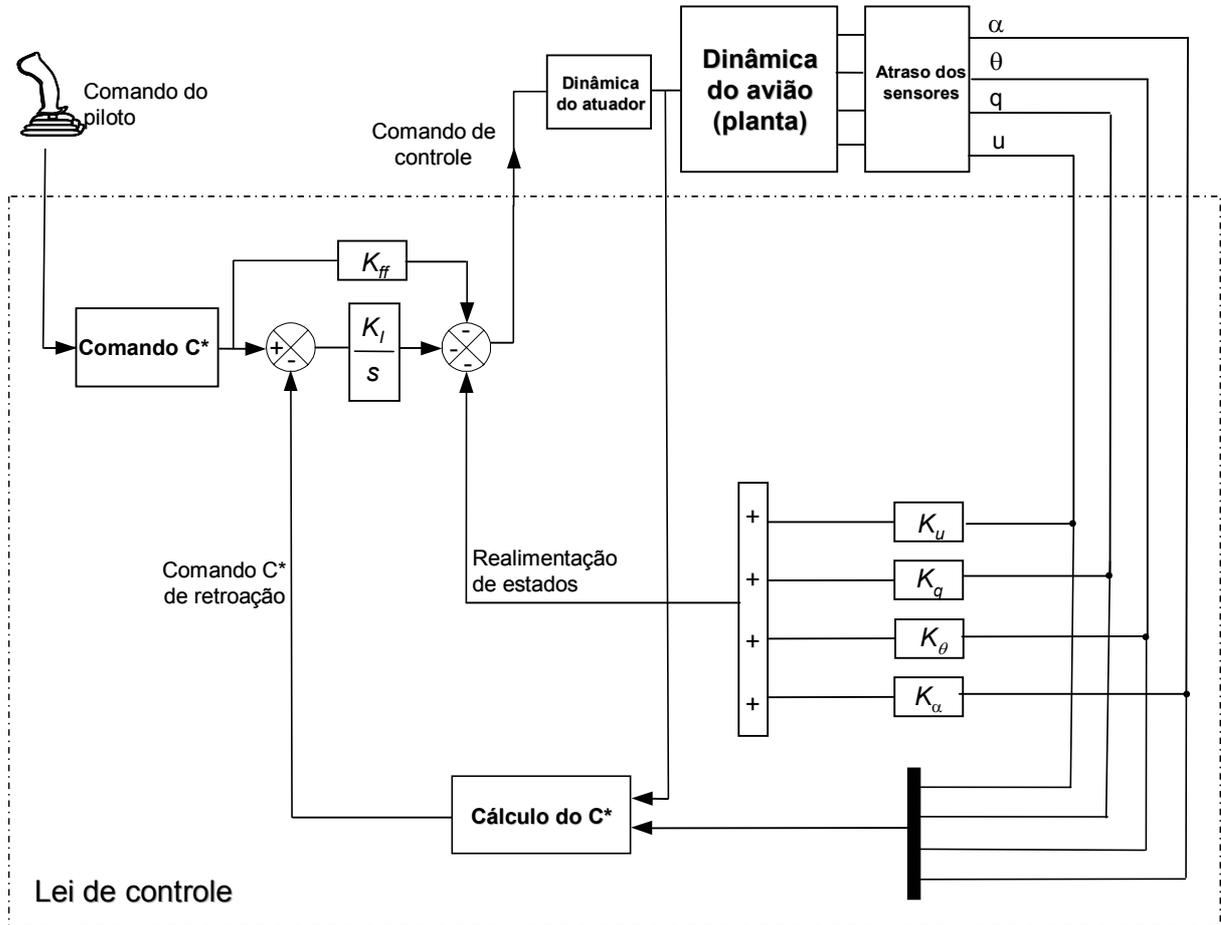


Figura 5.1 – Modelagem do sistema C*

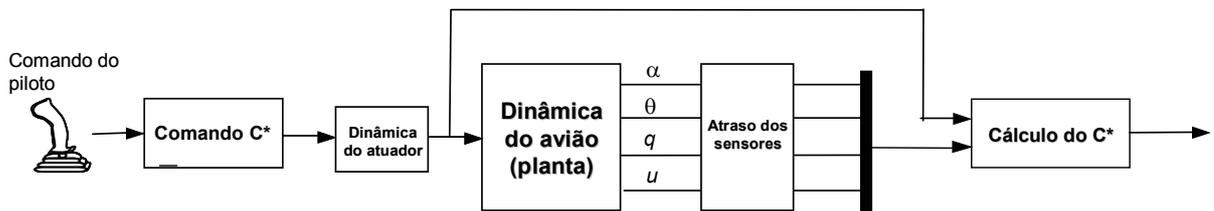


Figura 5.2 – Sistema C* em malha aberta

Para a versão simplificada do sistema, utilizou-se o modelo em malha aberta apresentado na Figura 5.3. Diferentemente do que ocorre na modelagem completa (Figura 5.2), a versão reduzida não considera os modelos do atuador e os atrasos dos sensores.

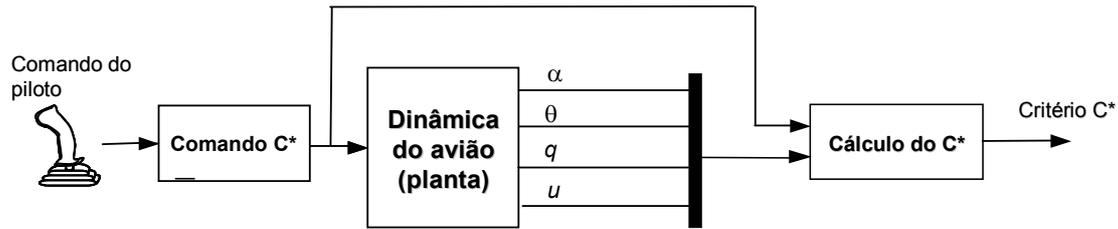


Figura 5.3 – Modelo simplificado (malha aberta)

O equacionamento das matrizes de estado de uma aeronave (planta do sistema) pode ser obtido em literaturas específicas de sistemas aeronáuticos (Helmersson *et al.*, 1997; Steven e Lewis, 1992). As variáveis de estado do modelo da aeronave são: α (ângulo de ataque), θ (ângulo de arfagem), q (velocidade angular longitudinal) e u (velocidade horizontal).

As matrizes de estado utilizadas no estudo são apresentadas no Apêndice A. Já no Apêndice B encontram-se ilustrações que indicam os eixos de referência e principais ângulos relativos aos movimentos de uma aeronave.

Para o cálculo do sinal C^* é necessário conhecer a velocidade angular (q) da aeronave e a aceleração normal sentida pelo piloto (Nz_{piloto}). Como q é um dos estados do sistema utilizados na realimentação, este sinal pode ser obtido diretamente. Já Nz_{piloto} não pode ser mensurado diretamente, mas pode ser obtido em função da entrada da planta e das variáveis de estado do sistema. No Apêndice C está indicada a expressão utilizada para o cálculo de Nz_{piloto} .

Os atrasos dos sensores podem ser modelados pelo método de Padé (Steven e Lewis, 1992). Além disso, por simplicidade, a dinâmica do atuador pode ser modelada por uma função de transferência de primeira ordem (Helmersson *et al.*, 1997).

O comando C^* dado pelo piloto é modelado por uma função que relaciona o deslocamento do *joystick* (ou manche) com o comando C^* (combinação entre aceleração normal e *pitch rate*) desejado para o movimento do *joystick* (*command shaping*).

5.2 SINTONIA DE UMA MALHA DE CONTROLE TIPO C*

Uma vez desenhada a arquitetura do sistema, é necessário, então, sintonizá-lo, isto é, achar os ganhos de realimentação do sistema que atendam aos critérios de desempenho de projeto.

O objetivo da sintonia é achar o conjunto $[K_a, K_\theta, K_q, K_v, K_i]$ de ganhos de realimentação que possibilite o atendimento dos requisitos especificados. Contudo, conforme mencionado, a presença de atrasos de transporte e estados não-realimentados oferece limitações à aplicação de técnicas tradicionais para a obtenção desses ganhos. Por outro lado, a sintonia desse sistema apresenta-se como uma grande oportunidade para a aplicação do método proposto.

5.2.1 Requisitos de projeto e respectivas formas de avaliação

Como requisitos de projeto, para o sistema em malha fechada, optou-se pela adoção de uma frequência de cruzamento de ganho ω_c de aproximadamente 3,0 rad/s e um fator de amortecimento mínimo $\zeta_{\min} = 0,4$ para os pólos dominantes da dinâmica rápida da aeronave. Para a análise desses dois requisitos, a função de transferência foi obtida pela relação entre a variável de saída (ângulo de arfagem da aeronave) e a variável de entrada (comando C* do piloto).

Para a análise de malha quebrada (*broken loops*) (Civita *et al.*, 2002), definiram-se como requisitos uma margem de fase mínima de desejável $Mf_{des} = 60^\circ$ e requerida de $Mf_{req} = 30^\circ$ e uma margem de ganho mínima desejável de $Mg_{des} = 6$ e requerida de $Mg_{req} = 4$. Para a avaliação das margens de fase e de ganho, as funções de transferência foram obtidas pela malha quebrada nas variáveis de estado da aeronave e na saída do atuador.

O critério de desempenho de amortecimento mínimo para os pólos dominantes foi avaliado através das informações obtidas pelo diagrama de Pólos e Zeros. Para os pólos com parte real negativa, foram atribuídos custos aos que estavam acima da reta que demarca o lugar geométrico dos pólos com amortecimento igual ao amortecimento mínimo esperado, isto é, aqueles que apresentaram amortecimentos menores que o mínimo requerido. Quanto maior o afastamento do pólo em relação à reta citada, maior o custo.

Para pólos com componentes reais positivas não desprezíveis foi atribuído um custo fixo, bem superior àqueles atribuídos para os que possuíam parte real negativa, para penalizar fortemente essa condição, já que ela leva à instabilidade do sistema.

A Figura 5.4 mostra um exemplo de pólos avaliados através do Diagrama de Pólos e Zeros. Três pólos estão dentro das especificações, sendo dois pólos com amortecimento acima do amortecimento mínimo e um pólo na origem. O quarto pólo está acima da linha de amortecimento mínimo, e a ele é atribuído um custo.

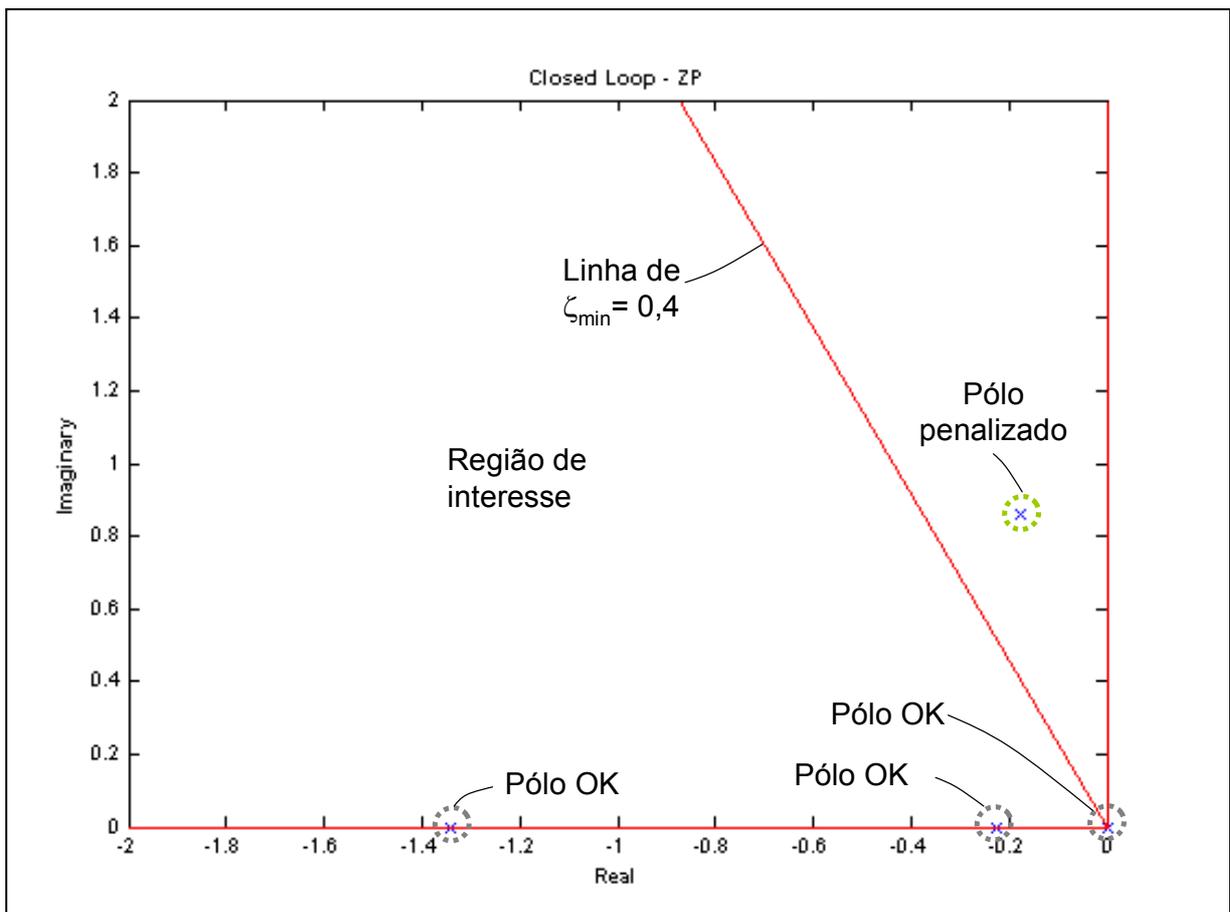


Figura 5.4 – Diagrama de Pólos e Zeros

Os critérios de desempenho relativos às margens de ganho e de fase foram determinados considerando as informações do diagrama de Nichols. Para a avaliação desse critério, fez-se uso de dois losangos (um interno e outro externo), nos quais a curva do diagrama não deveria penetrar. Os quatro vértices do losango são os pontos do diagrama que representam os valores positivos e negativos das margens de ganho e margem de fase. O losango interno representa a região definida pelos valores de margem requeridos, enquanto o losango externo representa a

região definida pelos valores desejáveis. A Figura 5.5 ilustra um exemplo da utilização do Diagrama de Nichols para a análise desse critério de desempenho.

Foram atribuídos custos para os trechos da curva do diagrama que penetravam dentro dos losangos. Quanto maior a penetração dentro do losango, maior o custo. Além disso, os trechos da curva situados dentro do losango interno (exemplo destacado em preto na Figura 5.5) foram consideravelmente mais penalizados do que os trechos da curva situados entre o losango externo e o interno (destacados em verde na figura).

Aos custos assim obtidos, também foram somados custos proporcionais à diferença entre a margem (de ganho e/ou de fase) desejável e a margem avaliada, caso a primeira fosse maior do que a segunda.

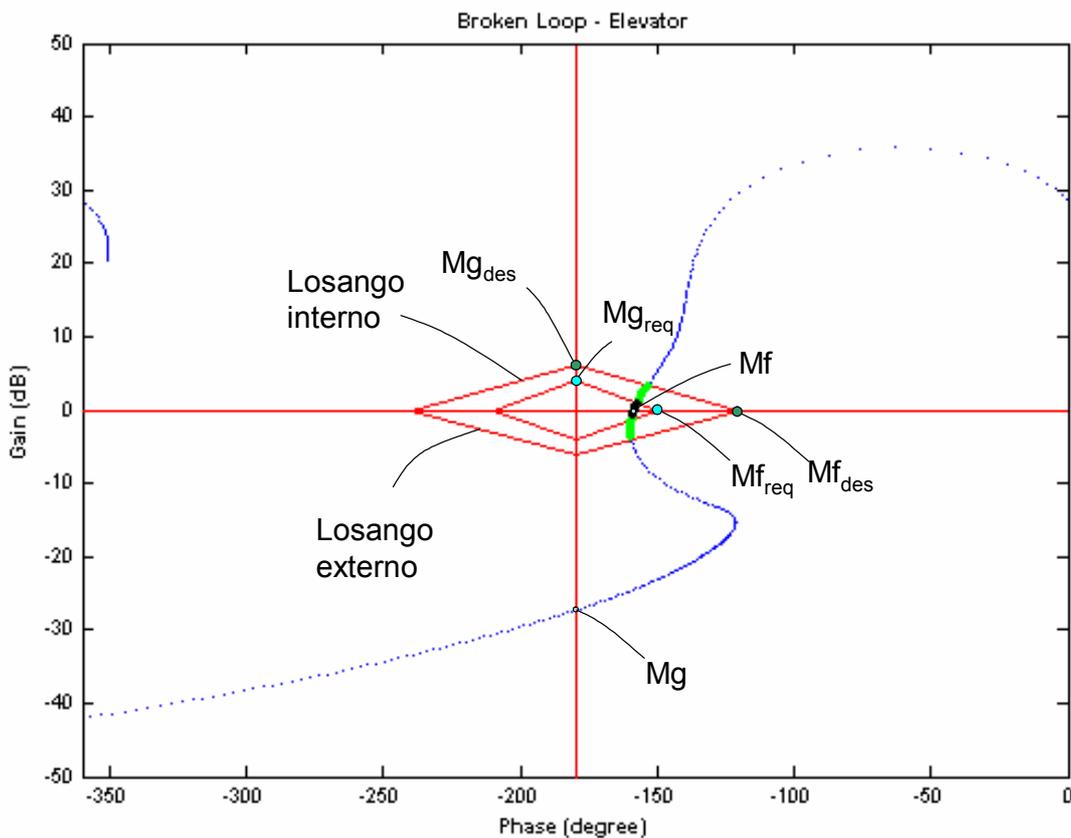


Figura 5.5 – Diagrama de Nichols

O critério de desempenho associado à frequência de cruzamento de ganho foi avaliado através de informações do diagrama de Bode, conforme descrito a seguir. Foram definidas duas regiões de interesse dentro das quais a curva deve permanecer: para frequências

inferiores à frequência de cruzamento de ganho esperada, a curva deve situar-se acima da reta horizontal de ganho unitário (0 dB) (Região 1). Já para frequências superiores àquela, a curva deve situar-se abaixo da linha de ganho unitário (Região 2).

Assim, para os trechos da curva de magnitude fora da região de interesse (região indicada com a cor cinza), foram atribuídos custos. Quanto maior o trecho da curva fora da área de interesse, maior o custo associado.

Ao custo assim obtido é agregado um custo proporcional à diferença entre a frequência de cruzamento de ganho esperada e aquela obtida.

Na Figura 5.6 é indicado um exemplo de uma análise de resposta em frequência. Na figura são indicadas as regiões de interesse onde se deseja que a curva permaneça. Os dois trechos da curva destacados estão fora da região de interesse e são penalizados. Na figura também estão destacadas a frequência de cruzamento obtida e aquela desejada, a cuja diferença também é atribuído um custo.

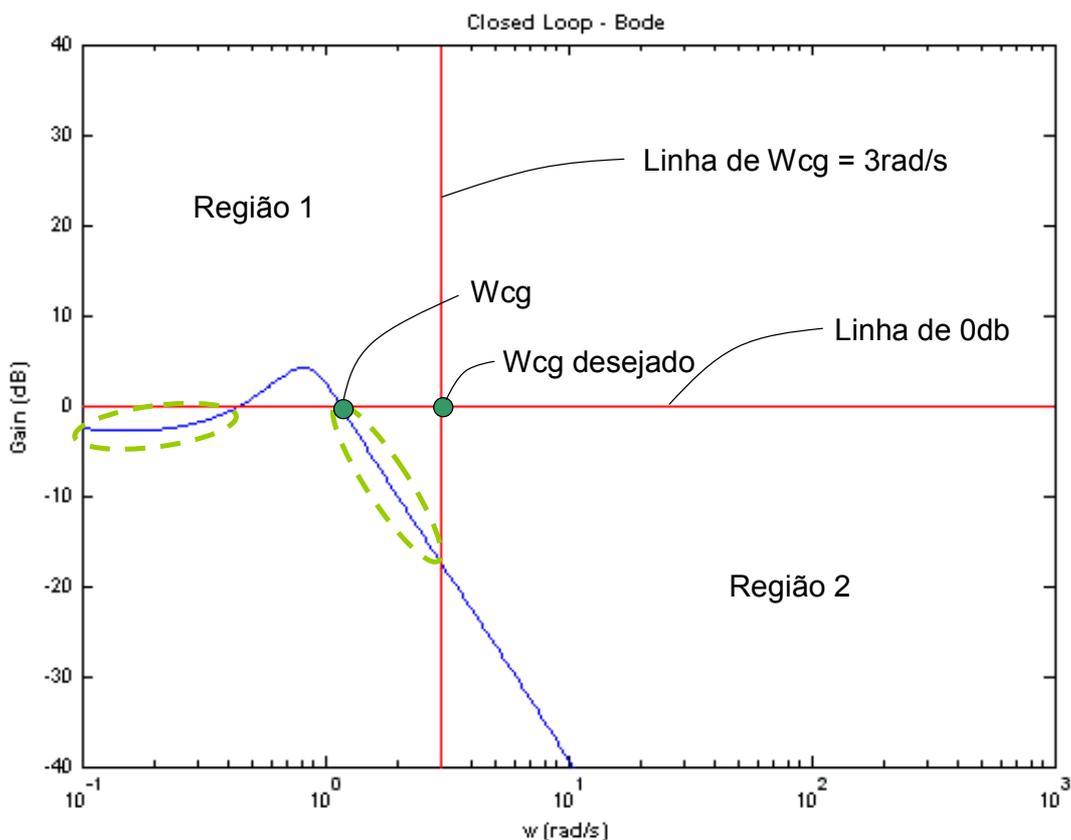


Figura 5.6 – Diagrama de Pólos e Zeros

Para este trabalho não se impôs nenhum requisito diretamente relacionado à resposta temporal e, por este motivo, não se tece nenhuma análise em relação a esse tipo de resposta.

Na ponderação dos índices de desempenho individuais, para a composição do custo total, priorizaram-se aqueles referentes às margens de custo e de fase, isto é, o não atendimento a essas margens foram mais penalizados do que o não atendimento aos demais requisitos de projeto.

5.2.2 Demais configurações do algoritmo

A faixa de frequência de análise ficou compreendida entre 0,1 rad/s e 100 rad/s. Durante as análises, notou-se que a frequência de cruzamento de ganho foi pouco sensível às variações nos ganhos de realimentação. Por esse motivo, o ganho de *feedforward* foi ajustado de forma a levar frequência de cruzamento de ganho para um valor próximo do valor desejado.

Os parâmetros de otimização para a técnica LQR foram ajustados em $Q=0,001*I$ e $R=I$, onde I é a matriz identidade.

A rede neural artificial foi configurada com duas camadas intermediárias, com 8 e 10 neurônios, respectivamente, e com uma taxa de aprendizagem de 0,01.

A varredura inicial dos cinco fatores K_{ki} , correspondentes aos cinco ganhos a serem sintonizados, utilizada para a geração dos conjuntos de ganhos a serem utilizados na primeira etapa de simulação, foi realizada para cada fator no intervalo compreendido entre 0,25 e 4, com resolução $r = 4$, totalizando 1024 combinações. Na segunda etapa de varredura, utilizada para a geração dos conjuntos de ganhos cujos índices de desempenho são estimados pela rede neural, o intervalo também foi mantido entre 0,25 e 4, porém com $r = 16$ (totalizando 1048576 combinações).

Dos casos estimados pela rede neural, foram escolhidos os 100 melhores (aqueles que apresentavam menor custo estimado). Esses casos foram, então, utilizados nas simulações do sistema, escolhendo-se as 40 melhores condições.

Os modelos foram implementados em Simulink™ e o algoritmo, implementado em MATLAB™. Para facilitar a utilização e visualização do processo proposto, uma interface de usuário (Graphical User Interface - GUI) foi desenvolvida, permitindo o ajuste de alguns

parâmetros e o acompanhamento dos resultados intermediários e finais. A Figura 5.7 mostra a GUI desenvolvida.

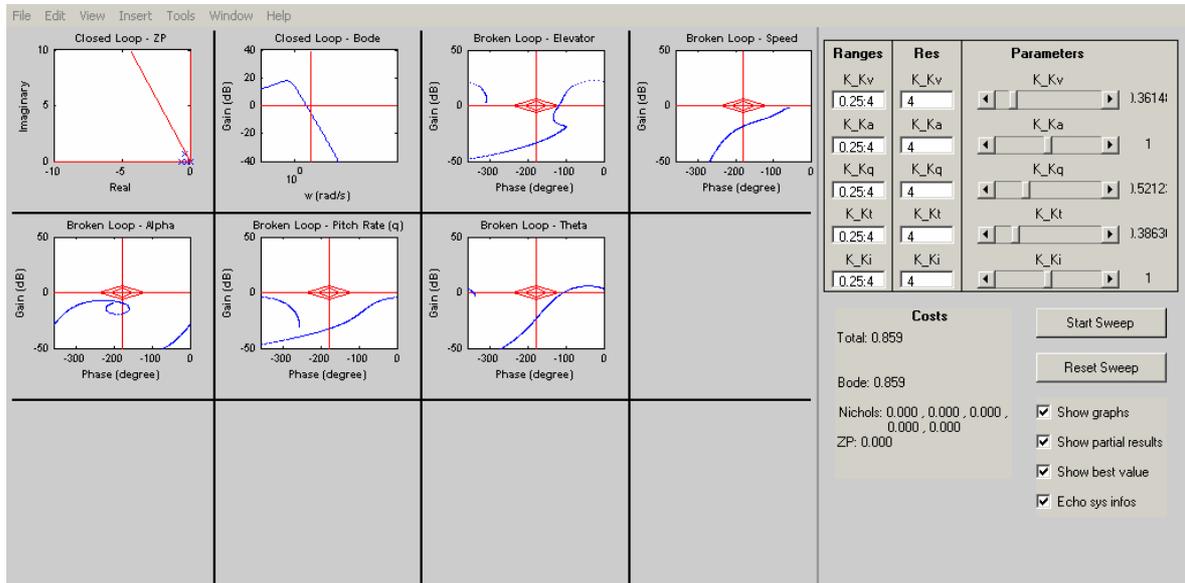


Figura 5.7 – GUI desenvolvida em MATLAB

Além disso, os parâmetros configuráveis foram separados em arquivos *.m* (scripts em MATLAB), gerenciando a configuração inicial do algoritmo.

5.2.3 Análise dos resultados obtidos

Na Tabela 1 são indicados os ganhos de realimentação obtidos pelo método LQR. Na Tabela 2 são indicados os 40 conjuntos de fatores de multiplicação para esses ganhos que geraram os menores custos.

Tabela 1 – Conjunto de ganhos inicial obtido pelo algoritmo

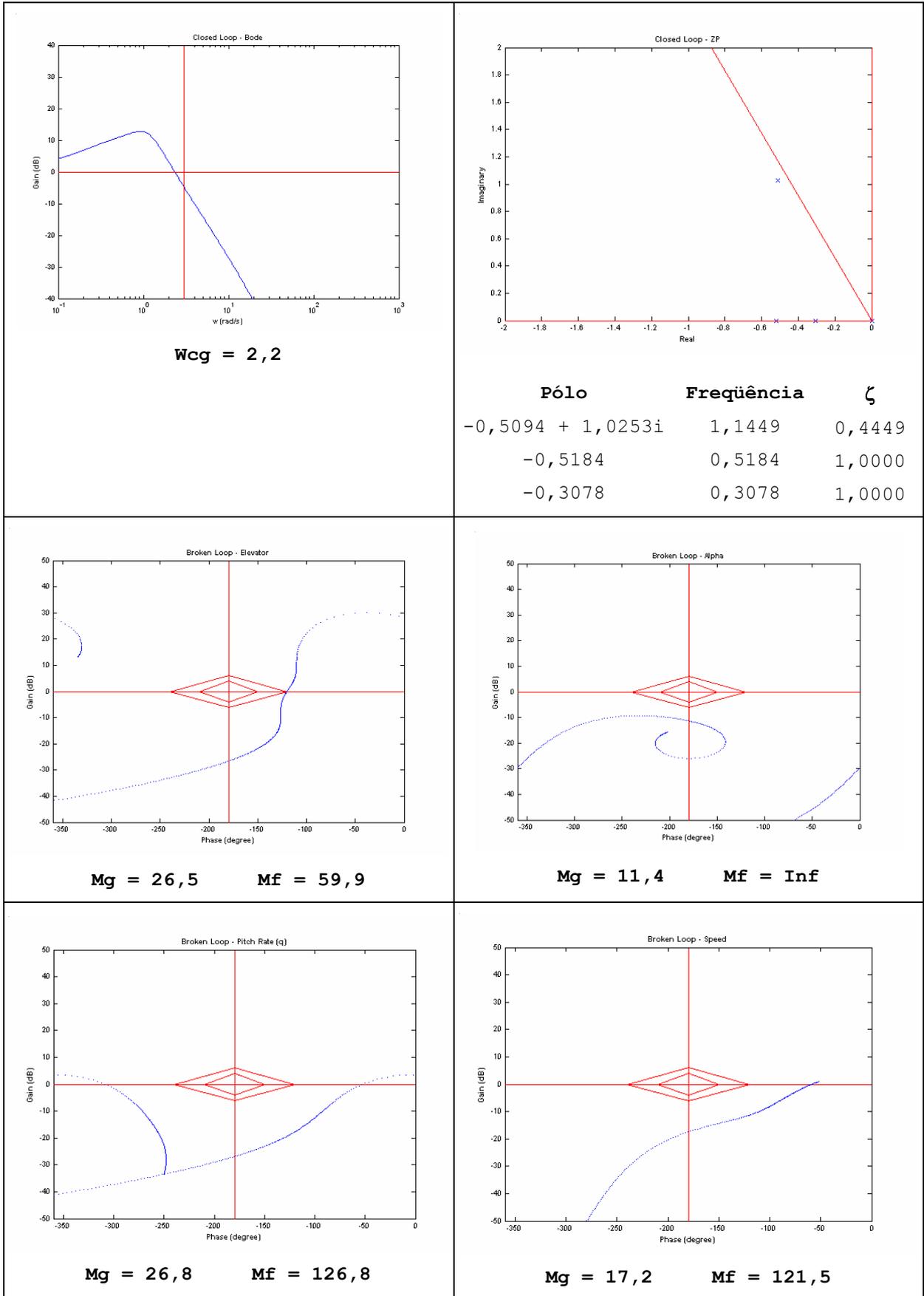
$K\alpha_{LQR}$	$K\theta_{LQR}$	Kq_{LQR}	Ku_{LQR}	Ki_{LQR}
0,0216	0,5018	-0,5229	-1,0901	0,0316

Tabela 2 – Conjuntos de fatores de multiplicação que geraram os menores custos

$K_{K\alpha_{LQR}}$	$K_{K\theta_{LQR}}$	$K_{Kq_{LQR}}$	$K_{Ku_{LQR}}$	$K_{Ki_{LQR}}$	Custo
0,3008	0,5236	1,3195	0,2500	0,3008	0,9211
0,2500	0,5236	1,3195	0,2500	0,3008	0,9216
0,3618	0,5236	1,3195	0,2500	0,2500	0,9230
0,3008	0,5236	1,3195	0,2500	0,2500	0,9235
0,2500	0,5236	1,3195	0,2500	0,2500	0,9239
0,2500	0,6300	1,3195	0,2500	0,5236	0,9277
0,2500	0,6300	1,3195	0,2500	0,4353	0,9319
0,2500	0,6300	1,3195	0,2500	0,3618	0,9355
0,2500	0,6300	1,3195	0,2500	0,3008	0,9385
0,2500	0,6300	1,3195	0,2500	0,2500	0,9410
0,3008	0,6300	1,5874	0,2500	0,6300	0,9599
0,2500	0,6300	1,5874	0,2500	0,6300	0,9604
0,3618	0,5236	1,5874	0,2500	0,2500	0,9607
0,3008	0,5236	1,5874	0,2500	0,2500	0,9614
0,2500	0,5236	1,5874	0,2500	0,2500	0,9620
0,3008	0,6300	1,5874	0,2500	0,5236	0,9649
0,2500	0,6300	1,5874	0,2500	0,5236	0,9655
0,2500	0,7579	1,5874	0,2500	0,9117	0,9669
0,3618	0,6300	1,5874	0,2500	0,4353	0,9685
0,2500	0,7579	1,5874	0,3008	0,4353	0,9686
0,3008	0,6300	1,5874	0,2500	0,4353	0,9693
0,2500	0,6300	1,5874	0,2500	0,4353	0,9699
0,3008	0,7579	1,5874	0,3008	0,3618	0,9717
0,3618	0,6300	1,5874	0,2500	0,3618	0,9722
0,2500	0,7579	1,5874	0,3008	0,3618	0,9722
0,3008	0,6300	1,5874	0,2500	0,3618	0,9729
0,2500	0,6300	1,5874	0,2500	0,3618	0,9735
0,2500	0,7579	1,5874	0,2500	0,7579	0,9744
0,3008	0,7579	1,5874	0,3008	0,3008	0,9747
0,3618	0,6300	1,5874	0,2500	0,3008	0,9752
0,2500	0,7579	1,5874	0,3008	0,3008	0,9753
0,3008	0,6300	1,5874	0,2500	0,3008	0,9759
0,2500	0,6300	1,5874	0,2500	0,3008	0,9765
0,3008	0,7579	1,5874	0,3008	0,2500	0,9772
0,3618	0,6300	1,5874	0,2500	0,2500	0,9777
0,2500	0,7579	1,5874	0,3008	0,2500	0,9778
0,3008	0,6300	1,5874	0,2500	0,2500	0,9784
0,2500	0,6300	1,5874	0,2500	0,2500	0,9790
0,2500	0,7579	1,5874	0,2500	0,6300	0,9808
0,2500	0,7579	1,5874	0,2500	0,5236	0,9860

A Tabela 3 apresenta as respostas do sistema sintonizado com o conjunto inicial de ganhos e a Tabela 4 apresenta as respostas do sistema sintonizado com o melhor conjunto de ganhos encontrado.

Tabela 3 – Respostas obtidas com a sintonia do conjunto inicial de ganhos



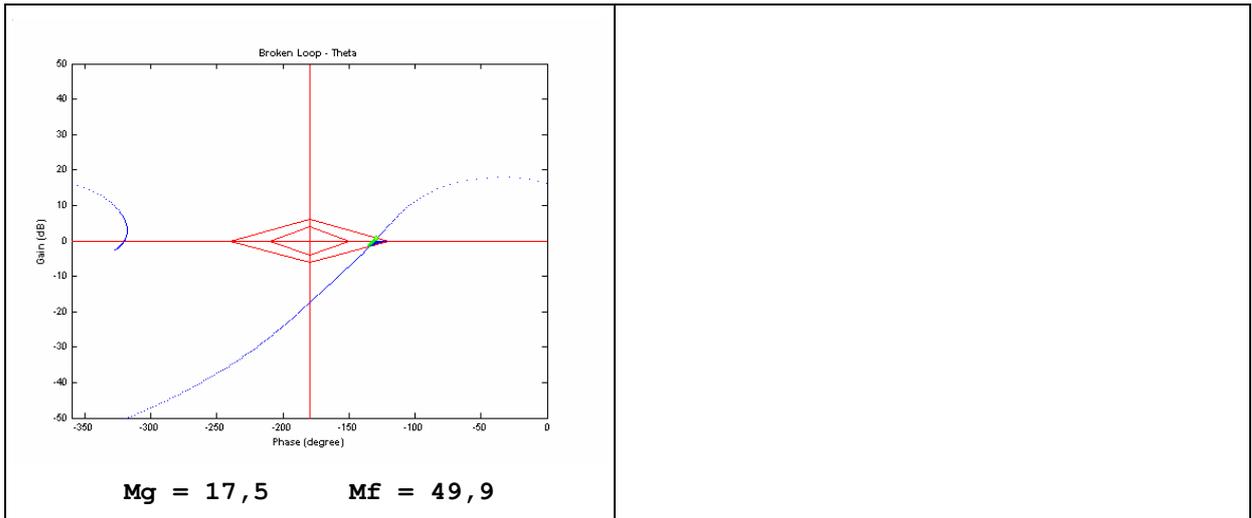
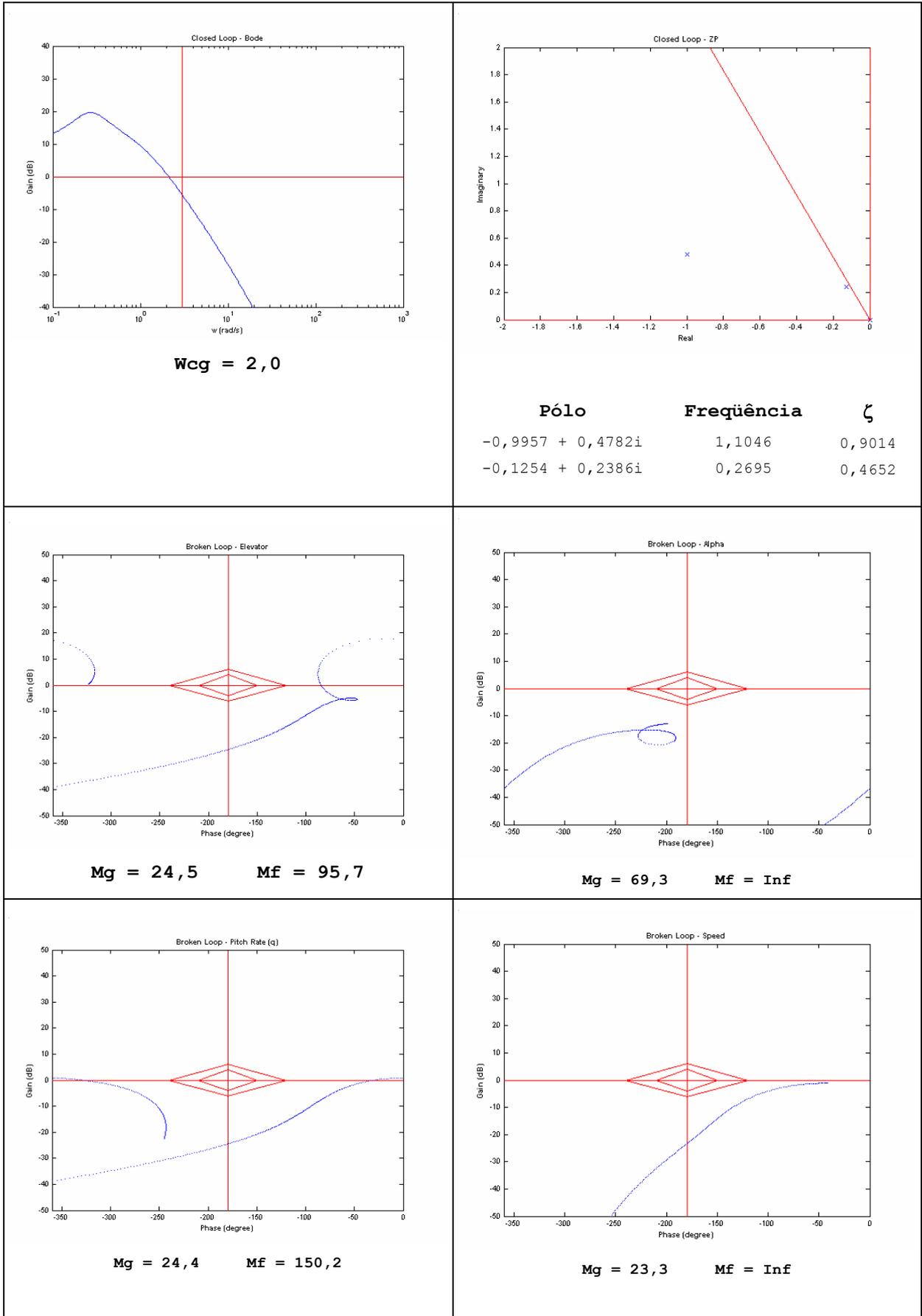
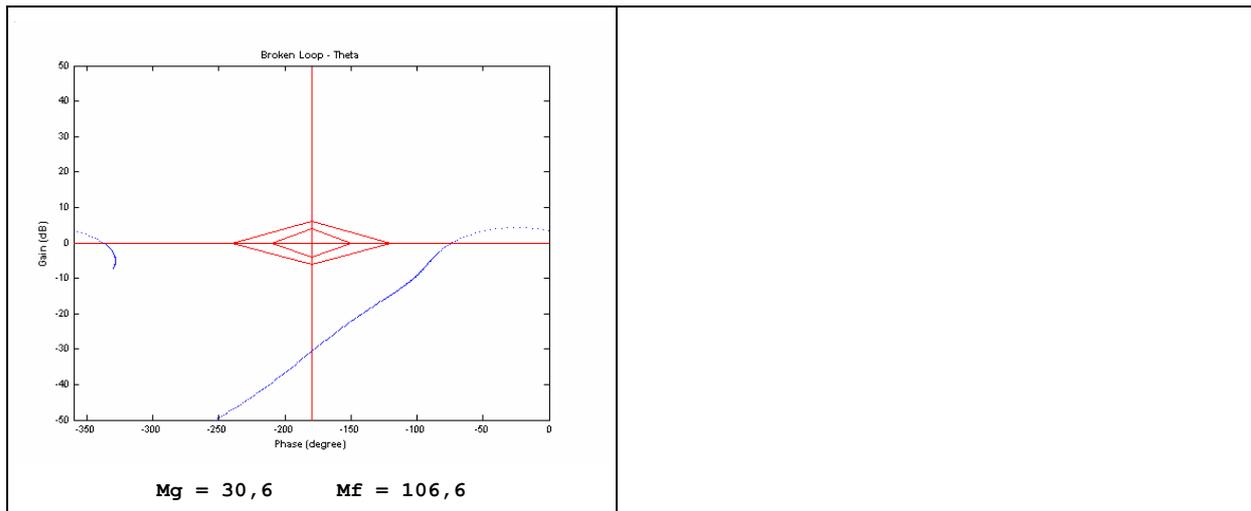


Tabela 4 – Respostas obtidas com o melhor conjunto de ganhos encontrado





Na Tabela 3 nota-se que as respostas do sistema sintonizado com os ganhos iniciais já se encontram próximas aos valores requeridos para o projeto, à exceção da margem de fase de malha quebrada em θ , que ficou abaixo de 50° .

Conforme mencionado anteriormente, um bom ponto de partida para a sintonia é aquele em que a sintonia não se encontra muito afastada dos requisitos de projeto e, neste caso apresentado, um bom ponto de partida foi facilmente encontrado com poucas tentativas.

Na Tabela 4 observa-se que os requisitos de projeto foram atendidos de forma satisfatória pela melhor sintonia encontrada. Os amortecimentos dos pólos dominantes foram maiores do que 0,4, as margens de ganho ficaram acima de 6 e as margens de fase ficaram acima de 60° . Apenas a frequência de cruzamento de ganho ficou ligeiramente abaixo de 3,0 rad/s (obteve-se 2,0 rad/s), mas tal valor ficou dentro da tolerância admitida, já que o requisito de projeto era uma frequência de cruzamento de ganho aproximada e não havia uma necessidade de grande acurácia para tal grandeza.

Comparando-se as respostas das Tabelas 3 e 4, nota-se que houve uma piora nos amortecimentos dos pólos dominantes e na frequência de cruzamento de ganho da sintonia final encontrada, em comparação com a sintonia inicial, em prol de uma significativa melhora das margens de ganho e de fase. Isso aconteceu conforme esperado, já que, conforme mencionado, o não-atendimento a essas margens foi bastante penalizado.

O custo associado à sintonia inicial foi de aproximadamente 12, enquanto o custo da melhor sintonia encontrada foi de aproximadamente 1. Para efeito de comparação de ordem de grandeza, um sistema com uma sintonia ou pouco mais afastada dos requisitos de projeto

apresentaria, por exemplo, um custo associado de 100. Já um sistema com uma sintonia muito distante dos requisitos de projeto apresentaria, por exemplo, um custo associado de 1000.

O tempo gasto em processamento pela CPU para o computador utilizado na execução completa do algoritmo proposto foi de 17 minutos (processador Athlon® 2300 XP+, 512 MB de memória RAM, sistema operacional Windows XP, MATLAB® 6.5), dos quais aproximadamente 10 minutos foram gastos apenas na primeira etapa de simulação. A Tabela 5 indica, para o exemplo de aplicação apresentado, os tempos aproximados obtidos das principais etapas do algoritmo e também o tempo total de execução do algoritmo.

Tabela 5 – Tempos das etapas para o exemplo de aplicação

Etapa	Tempo aproximado
Primeira etapa de varredura e simulação dos conjuntos de ganhos	12 minutos
Treinamento da rede neural	Menos de 1 minuto
Utilização da rede neural para estimativa	2 minutos
Validação (simulação) dos melhores casos estimados pela rede	Menos de 1 minuto
Tempo total de execução do algoritmo	17 minutos

Conforme se observa na Tabela 5, a primeira etapa de simulação é a etapa que demanda mais tempo. Por isso, recomenda-se um cuidado especial na seleção da resolução r de varredura dos fatores K_{ki} nesta etapa. Se, por um lado, quanto maior o valor de r na varredura, maior a quantidade de padrões de entrada para o conjunto de dados a ser apresentado à rede neural e, portanto, melhor o treinamento e acurácia na estimativa, por outro lado, o aumento de r pode implicar um aumento significativo no tempo de execução do algoritmo.

No exemplo, optou-se pela varredura com resolução $r = 4$ para cada fator K_{ki} . No total, houve $4^5=1024$ casos a serem simulados e avaliados na primeira etapa. Se a opção fosse $r = 5$ para cada fator K_{ki} , haveria $5^5 = 3125$ casos a serem simulados e avaliados, ou seja, 3 vezes mais simulações nessa etapa, com um tempo 3 vezes maior de processamento. Para este exemplo, significaria mais do que dobrar o tempo total de execução do algoritmo.

A Tabela 6 mostra, para efeito de comparação, algumas resoluções possíveis de serem adotadas e o tempo esperado para a execução da primeira etapa de simulação, considerando o número de conjuntos de ganhos obtidos pela varredura realizada com as respectivas resoluções.

Tabela 6 – Impacto do aumento da resolução no tempo de execução

Resolução (r)	Número de conjuntos de ganhos (r^n)	Tempo aproximado para primeira etapa de simulação
4	$4^3=1024$	12 minutos
5	$5^3=3125$	36 minutos
6	$6^3=7776$	91 minutos
7	$7^3=16807$	196 minutos

Na Tabela 5, anterior à Tabela 6, nota-se que, embora se tenha apresentado uma quantidade realmente muito grande de conjunto de ganhos (mais de um milhão) para a rede neural, para que esta estimasse os custos associados aos conjuntos de ganhos, o tempo gasto nessa etapa de estimativa foi pequeno (2 minutos), em comparação com o tempo total de execução do algoritmo. Assim, não é necessária nessa etapa uma preocupação com as resoluções, diferentemente do que ocorre na varredura dos conjuntos de ganho simulados.

Ressalta-se, no entanto, que, para alcançar nessa etapa significativa melhora de desempenho computacional do algoritmo implementado, os conjuntos de ganho foram apresentados à rede neural em blocos de 20 mil conjuntos, e não todos de uma vez. Observou-se que a rede neural artificial utilizada é consideravelmente melhor em desempenho quando os padrões são apresentadas em pequenas quantidades à rede.

Capítulo 6

CONCLUSÕES E PERSPECTIVAS FUTURAS

Este trabalho apresentou uma proposta de metodologia para a sintonia de malhas de controle com realimentação de estados. Partindo do método LQR, amplamente conhecido, mas que apresenta algumas limitações, a metodologia baseia-se na busca de um conjunto de ganhos de sintonia que cumpra os requisitos mínimos de projeto, em contraposição a algumas metodologias existentes que buscam uma solução ótima.

Através da atribuição de custos associados ao não cumprimento dos requisitos de projeto e na utilização de uma rede neural artificial, realiza-se uma busca de ganhos de realimentação em torno de um conjunto de ganhos inicial encontrado pelo método LQR aplicado a um modelo reduzido ou simplificado do sistema de controle original.

Para demonstrar a aplicabilidade da metodologia, utilizou-se um sistema de controle implementando a técnica C^* , de controle do movimento longitudinal de uma aeronave.

Como resultado, obteve-se mais de um conjunto de ganhos que sintonizaram o sistema adequadamente, fazendo com que o sistema atingisse os requisitos de projeto. Tal objetivo foi alcançado com um baixo custo computacional, o que mostrou a eficácia e a eficiência do método proposto.

A principal contribuição deste trabalho foi a proposição de um algoritmo que: utiliza índices de desempenho associados ao cumprimento dos requisitos de projeto para realizar uma sintonia automatizada de sistemas de controle com realimentação de estados; utiliza redes neurais artificiais para melhorar sua eficiência computacional; permite a avaliação de virtualmente qualquer requisito de projeto.

Como continuação dos estudos, sugere-se a abordagem de outras formas de avaliação de critérios de desempenho associados ao não cumprimento dos requisitos de projeto, bem como a proposição de técnicas para a composição do índice de desempenho global do sistema.

Propõe-se, também, o estudo e a análise de mudanças nas técnicas propostas ou a utilização de outras técnicas que sejam equivalentes, como por exemplo:

- Aplicação do método LQR ao sistema original, considerando-se, do resultado obtido, apenas os ganhos dos estados que são realimentados, em detrimento da aplicação desse método a um sistema simplificado, que possui só os estados que serão realimentados;
- Utilização da técnica de alocação de pólos, em vez do método LQR, para a obtenção de um conjunto inicial de ganhos;
- Utilização, como conjunto inicial de ganhos, em detrimento da utilização daquele obtido pelo LQR, dos ganhos de sintonia encontrados para condições de linearização adjacentes, em sistemas não-lineares linearizados em diversos pontos de operação,
- Utilização, para a obtenção da estimativa dos índices de desempenho baseada nos dados iniciais obtidos, de outras arquiteturas para a rede neural artificial e de outros algoritmos de aprendizado, e, ainda, a utilização de técnicas híbridas (Lin e Lee, 1996).

Por fim, propõe-se a adaptação da técnica proposta para aplicação a outras arquiteturas de sistemas de controle, além da retroação de estados, abordada neste trabalho. Sugere-se a análise das limitações de técnicas já existentes e o desenvolvimento de uma forma de se obter um ponto de partida para a sintonia que consiga contornar tais limitações.

REFERÊNCIAS BIBLIOGRÁFICAS

- AIRBUS (1988)**, A320 Fly By Wire and Handling Qualities, 5th Performance and Operations Conference, Airbus Industrie Engineering Directorate Flight Division
- BEALE, R., JACKSON, T. (1990)**, Neural Computing: An Introduction, IOP Publishing Ltd., Bristol, UK
- BIGUS, J. P. (1996)**, Data Mining with Neural Networks: Solving Business Problems from Application Development to Decision Support, McGraw-Hill
- CALISE, A.J., RYSDYK, R.T. (1998)**, Nonlinear adaptive flight control using neural networks, Control Systems Magazine – IEEE, 18 (6), pp. 14-25
- CARDENUTO, N.C. (2003)**, Aeronaves configuradas por controle do tipo preditivo neural”. Dissertação de Doutorado, Laboratório de Computação e Matemática Aplicada, INPE, São José dos Campos, SP
- CHONG, E. K. P., ZAK, S. H. (2001)**, An Introduction to optimization, 2nd Edition, John Wiley & Sons, Inc.
- CHEN, L. H. E CHIANG, C. H. (2004)**, An intelligent control system with a multi-objective self-exploration process, Fuzzy Sets and Systems, 143 (2), pp. 275-294
- CIVITA, M. L., KANADE, T., PAPAGEORGIOU, G., MESSNER, W. (2002)**, Design and Flight Testing of a High-Bandwidth H-Infinity Loop Shaping Controller for a Robotic Helicopter, AIAA Guidance, Navigation, and Control Conference and Exhibit, Monterey, California

- DARSONO, S. E LABADIE. J. W. (2007)**, Neural-optimal control algorithm for real-time regulation of in-line storage in combined sewer systems, *Environmental Modelling & Software*, 22 (9), pp. 1349-1361
- ETKIN, B., REID, L. D. (1996)**, *Dynamics of Flight, Stability and Control*. 3^a ed., John Wiley, 1996
- FAHLMAN, S.E. (1988)**, An empirical study of learning speed in back-propagation networks, Technical Report CMU-CS-88-162, School of Computer Science - Carnegie Mellon University, Pittsburgh, PA.
- FAVRE, C. (1994)**, Fly-by-wire for commercial aircraft: the Airbus experience. *International Journal of Control*, 59(1), pp. 139–157
- FIELD, E. (1993)**, The application of a C* flight control law to large civil transport aircraft, College of Aeronautics Reports
- GARCÍA, C. (2005)**, Artificial intelligence applied to automatic supervision, diagnosis and control in sheet metal stamping processes, *Journal of Materials Processing Technology*, Vol. 164-165, pp. 1351-1357
- GAUTREY, J.E. ET AL. (1997)**, RCAM Design Challenge Presentation Document: the Classical Control Approach, GARTEUR/TP-088-13
- GROOM, N. J., SCHAFFNER, P. R. (1990)**, An LQR Controller Design Approach for a Large Gap Magnetic Suspension System (LGMSS), NASA Technical Memorandum 101606.
- GUIDORIZZI, H. L. (2002)**, *Um Curso de Cálculo*, 5^a ed., Vol. 4, Ed. Livros Técnicos e Científicos
- HAYKIN, S. (1999)**, *Neural Networks: A Comprehensive Foundation*, 2nd ed., Prentice-Hall, Inc.

- HELMERSSON, A. ET AL. (1997)**, Robust Flight Control Design Challenge, Problem Formulation and Manual: the Research Civil Aircraft Model (RCAM), GARTEUR/TP-088-03
- LIEN, C.H. E YU, K. W. (2008)**, LMI optimization approach on robustness and H_{∞} control analysis for observer-based control of uncertain systems, Chaos, Solitons & Fractals, 36 (3), pp. 617-627
- LIN, C.T, LEE, C.S.G. (1996)**, Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent System, Prentice-Hall International, Inc.
- LOPES, P. S (2007)**, Detecção de danos em estruturas por meio de técnicas de redes neurais artificiais e algoritmos genéticos, Dissertação de Mestrado, Universidade Federal de Itajubá
- MATLAB Control System Toolbox Reference (2008)**, The MathWorks Inc. Disponível em http://www.mathworks.com/access/helpdesk/help/pdf_doc/control/reference.pdf . Acesso em Maio/2008
- NAJMABADI, K., EVANS, M. R., BLEEG,R. J. , BREUHAUS, R.S. (1999)**, Aircraft pitch-axis stability and command augmentation system, US Patent 5979835. Disponível em <http://www.patentstorm.us/patents/5979835-description.html>. Acesso em Maio/2008
- OGATA, K. (2003)**, Engenharia de Controle Moderno, 4ª ed., Pearson Brasil
- PHILIPS, C.L., HARBOR, R.D. (1997)**, Sistemas de Controle e Realimentação, Makron Books
- RAO, H. S., GHORPADE, V. G., MUKHERJEE, A. (2005)**, “A genetic algorithm based back propagation network for simulation of stress-strain response of ceramic-matrix-composites”, Computers and Structures, v 84, pp 330–339
- SINGH, V., GUPTA, I. E GUPTA, H.O. (2005)**, ANN based estimator for distillation - inferential control, Chemical Engineering and Processing, 44 (7), pp. 785-795

SIQUEIRA, D., MOREIRA, F. (2007), Robust Flight Control Design Supported by Flying Qualities Analysis. AIAA Guidance, Navigation and Control Conference and Exhibit. Hilton Head, USA

STEVEN, B. L., LEWIS, F. L. (1992), Aircraft Control and Simulation, John Wiley & Sons

TOBIE, H.N., ELLIOT, E.M., MALCOLM, L.G. (1966), A new longitudinal handling qualities criterion, Boeing Commercial Airplane Company

ZÁRATE, L. E. E BITTENCOUT, F. R. (2008), Representation and control of the cold rolling process through artificial neural networks via sensitivity factors , Journal of Materials Processing Technology, 197 (1-3), pp. 344-362

Apêndice A

MATRIZES DE ESTADO UTILIZADAS NO EXEMPLO DE APLICAÇÃO

```
A = [  
  -0.08896      1.257      -0.2306      -9.793  
  -0.008065     -0.535       0.9558       0.00942  
  -0.0009221    -0.6563      -0.6617      -0.00137  
           0           0           1           0  
];
```

```
B = [  
  0.04565  
  -0.0627  
  -1.13  
  0  
];
```

```
C = [  
      1      0      0      0  
      0      1      0      0  
      0      0      1      0  
      0      0      0      1  
];
```

```
D = [  
  0  
  0  
  0  
  0  
];
```

Apêndice B

ILUSTRAÇÕES COM OS EIXOS E REFERÊNCIAS RELATIVOS AO MOVIMENTO DE UMA AERONAVE

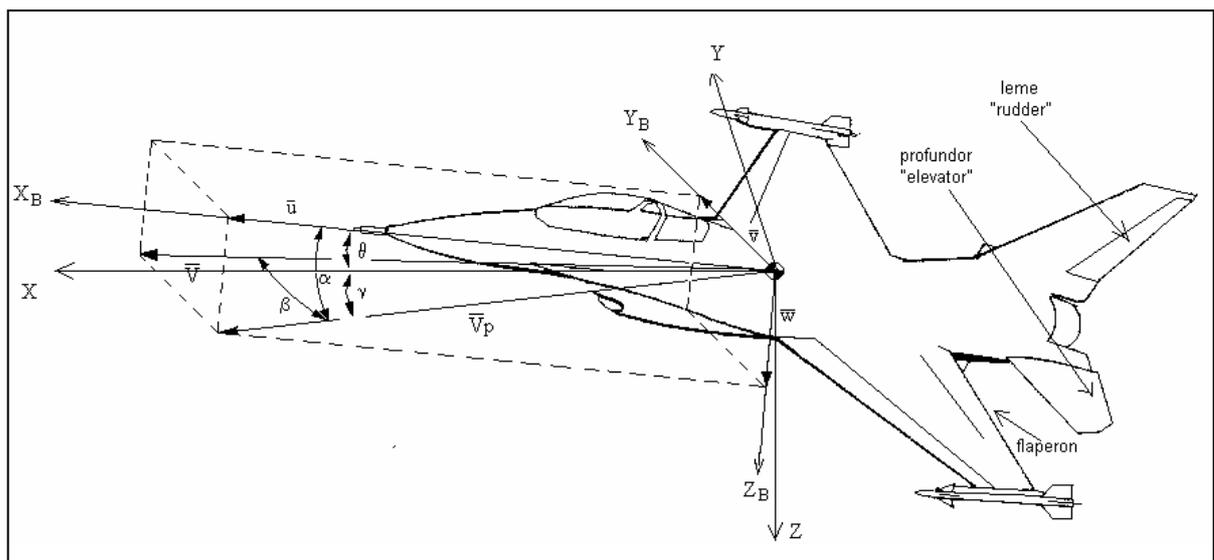


Figura B1 – Eixos de referência e ângulos (Retirada de (Cardenuto, 2003))

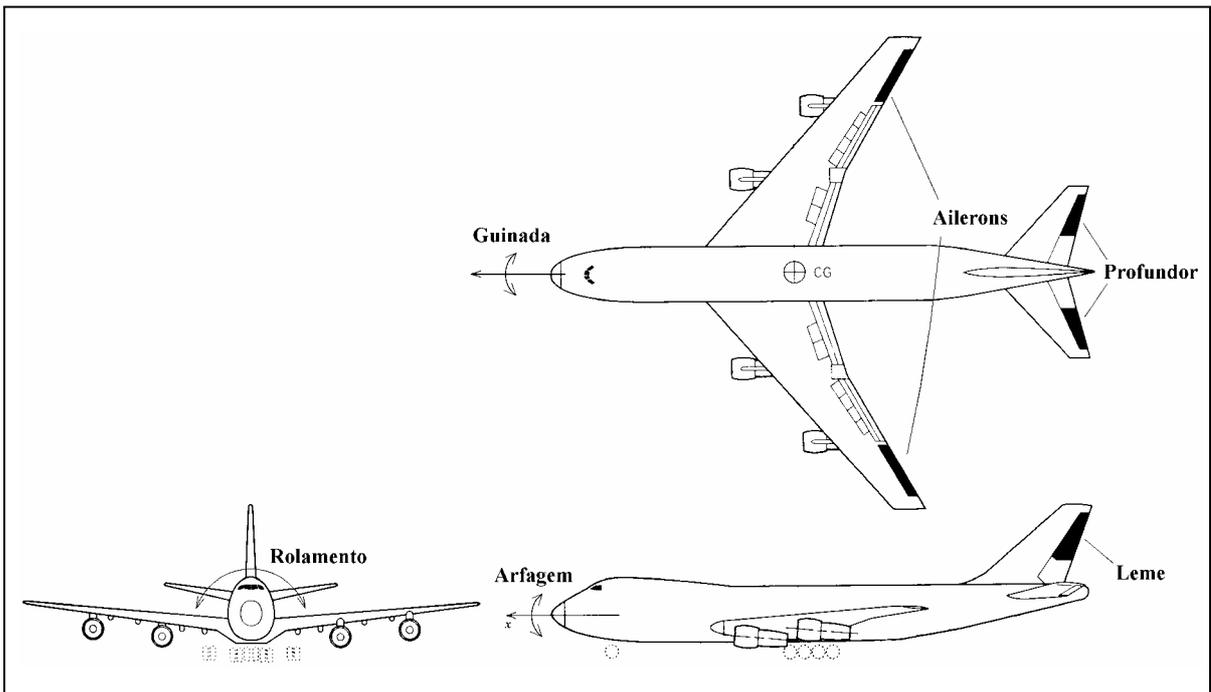


Figura B2 – Movimentos rotacionais de uma aeronave (Adaptada de (Etkin e Reid, 1996))

Apêndice C

OBTENÇÃO DA ACELERAÇÃO NORMAL SENTIDA PELO PILOTO

Para se obter a aceleração normal, sentida pelo piloto, Nz_{piloto} , em função das variáveis de estado do sistema e da entrada da planta, é utilizada a expressão (A1) (Helmersson *et al.*, 1997; Steven e Lewis, 1992).

$$Nz_{\text{piloto}} = \frac{V_0}{g_0} (\dot{\theta} - \dot{\alpha}) + \frac{l}{g_0} \dot{q} \quad (\text{A1})$$

Nessa expressão, os valores de V_0 (velocidade da aeronave na condição de linearização) e g_0 (gravidade local na condição de linearização) são constantes para um dado modelo linearizado. O valor de l (distância entre o centro de gravidade da aeronave e a posição do piloto) é constante para uma dada aeronave. Como θ , q e α são estados do sistema dinâmico em questão, $\dot{\theta}$, \dot{q} e $\dot{\alpha}$ são obtidos diretamente na equação de estados do movimento longitudinal.