

Madson Cruz Machado

**Sintonia RNA-RBF para o Projeto *Online* de
Sistemas de Controle Adaptativo**

São Luís

2017

Madson Cruz Machado

Sintonia RNA-RBF para o Projeto *Online* de Sistemas de Controle Adaptativo

Dissertação apresentada ao Programa de Pós Graduação em Engenharia de Eletricidade da UFMA como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia de Eletricidade. Área de concentração: Automação e Controle.

Universidade Federal do Maranhão
Centro de Ciências Exatas e Tecnologia
Programa de Pós Graduação em Engenharia de Eletricidade

Orientador: Prof. Dr. João Viana da Fonseca Neto

São Luís

2017

Machado, Madson Cruz.

Sintonia RNA-RBF para o Projeto *Online* de Sistemas de Controle Adaptativo/
Madson Cruz Machado. – 2017

88 p.

Orientador: João Viana da Fonseca Neto.

Dissertação (Mestrado) – Programa de Pós-graduação em Engenharia de
Eletricidade/CCET, Universidade Federal do Maranhão, São Luís, 2017.

1. Redes neurais artificiais. 2. Controle adaptativo. 3. Projeto *online*. I. Fonseca
Neto, João Viana da. II. Título

Madson Cruz Machado

Sintonia RNA-RBF para o Projeto *Online* de Sistemas de Controle Adaptativo

Dissertação apresentada ao Programa de Pós Graduação em Engenharia de Eletricidade da UFMA como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia de Eletricidade. Área de concentração: Automação e Controle.

Trabalho aprovado. São Luís, 26 de maio de 2017:

Prof. Dr. João Viana da Fonseca Neto
Orientador

Prof. Dr. Francisco das Chagas de Souza
Examinador Interno

Prof. Dr. Raimundo Carlos Silvério Freire
Examinador Externo

São Luís
2017

*Este trabalho é dedicado à minha família,
pois em todos os momentos sempre estiveram ao meu lado.*

Agradecimentos

Agradeço primeiramente a Deus pelo fôlego de vida e por permitir com que este trabalho seja concluído.

Agradeço ao prof. Dr. João Viana da Fonseca Neto pela orientação, apoio e discussões durante a realização deste trabalho. Agradeço também a todos os participantes do LAC e LABSECI pelo companheirismo, amizade e por proporcionarem um ambiente de debates acadêmicos. Agradeço a todos os funcionários da UFMA e terceirizados com os quais tive contato, professores, administrativos, pessoal da limpeza, etc.

Por fim, mas não menos importante, aos meus familiares por acreditarem em mim e pelo apoio incondicional.

“Connaître, ce n’est point démontrer, ni expliquer. C’est accéder à la vision.”
Antoine de Saint-Exupéry

*“Whoever loves discipline loves knowledge,
but whoever hates correction is stupid.”*
The Holy Bible – NIV; Proverbs 12,1

Resumo

A necessidade de aumentar a produtividade industrial, associada com os requisitos de qualidade e baixo custo, gerou uma demanda para o desenvolvimento de controladores de alto desempenho. Motivado por esta demanda, apresentou-se neste trabalho modelos, algoritmos e uma metodologia para o projeto *online* de sistemas de controle de alto desempenho. Os modelos apresentam características de adaptabilidade por meio de arquiteturas de sistemas de controle adaptativo. O desenvolvimento de modelos, baseia-se em redes neurais artificiais (RNA), do tipo função de base radial (RBF, *radial basis function*), para o projeto *online* de sistemas de controle adaptativo do tipo modelo de referência associado com o controle de modos deslizantes (SMC, *sliding mode control*). Os algoritmos e o sistema embarcado desenvolvidos para o projeto *online* são avaliados para o rastreamento de alvos móveis, neste caso, o rastreamento da radiação solar. O sistema de controle tem o objetivo de manter a superfície do módulo fotovoltaico perpendicular à radiação solar, pois dessa forma a energia gerada pelo módulo será a maior possível. O processo consiste de um painel fotovoltaico acoplado em uma estrutura que gira em torno de um eixo paralelo à superfície da terra, posicionando o painel de forma a capturar a maior radiação solar em função de seu deslocamento ao longo do dia.

Palavras-chave: Redes neurais artificiais. controle adaptativo. projeto *online*.

Abstract

The need to increase industrial productivity coupled with quality and low cost requirements has generated a demand for the development of high performance controllers. Motivated by this demand, we presented in this work models, algorithms and a methodology for the online project of high-performance control systems. The models have characteristics of adaptability through adaptive control system architectures. The models developed were based on artificial neural networks of radial basis function type, for the online project of model reference adaptive control systems associated with the of sliding modes control. The algorithms and the embedded system developed for the online project were evaluated for tracking mobile targets, in this case, the solar radiation. The control system has the objective of keeping the surface of the photovoltaic module perpendicular to the solar radiation, in this way the energy generated by the module will be as high as possible. The process consists of a photovoltaic panel coupled in a structure that rotates around an axis parallel to the earth's surface, positioning the panel in order to capture the highest solar radiation as function of its displacement throughout the day.

Keywords: Artificial neural networks. adaptive control. online project.

Lista de ilustrações

Figura 1 – Arquitetura do sistema de controle de alto desempenho	17
Figura 2 – Diagrama de blocos de um sistema com escalonamento de ganho	22
Figura 3 – Baseado no regulador autoajustável indireto	23
Figura 4 – Arquitetura de uma rede perceptron de múltiplas camadas com duas camadas ocultas	30
Figura 5 – Estrutura de uma rede de funções de base radial	37
Figura 6 – Diagrama de blocos de um sistema adaptativo por modelo de referência	41
Figura 7 – Conceitos de estabilidade	42
Figura 8 – Superfície de deslizamento	44
Figura 9 – Controle adaptativo neural	46
Figura 10 – Controle neuro-adaptativo aplicado ao PV	53
Figura 11 – Visão geral do sistema	54
Figura 12 – PV como carga no eixo do motor	54
Figura 13 – Diagrama de blocos atuador-carga	56
Figura 14 – Sistema em malha fechada	58
Figura 15 – Resposta ao degrau do sistema em malha aberta	59
Figura 16 – Resposta ao degrau do sistema em malha fechada	59
Figura 17 – Resposta ao degrau unitário do modelo de referência	60
Figura 18 – Gráficos da posição	61
Figura 19 – Gráficos da velocidade	62
Figura 20 – Lei de controle	62
Figura 21 – Gráfico de $\hat{f}(x)$ e $f(x)$	63
Figura 22 – Sistema de controle em malha fechada	64
Figura 23 – Microcontrolador do sistema de controle	65
Figura 24 – Montagem do sistema de acionamento	65
Figura 25 – Diagrama esquemático do circuito dobrador de tensão	66
Figura 26 – Diagrama esquemático circuito de disparo	67
Figura 27 – Diagrama esquemático circuito da ponte H	67
Figura 28 – Posição do seguidor	69
Figura 29 – Velocidade do seguidor	69
Figura 30 – Lei de controle – seguidor	70
Figura 31 – Gráfico de $\hat{f}(x)$ e $f(x)$ – seguidor	70
Figura 32 – Modelo de um neurônio não linear com múltiplas entradas	82
Figura 33 – Função de Heaviside	83
Figura 34 – Função linear por partes	84
Figura 35 – Função logística	84

Figura 36 – Função tangente hiperbólica	85
Figura 37 – Função gaussiana	86
Figura 38 – Modelo do conjunto atuador-carga	87

Lista de abreviaturas e siglas

LMS	<i>Least Mean Square</i>
LQR	<i>Linear Quadratic Regulation</i>
MIMO	<i>Multiple-Input Multiple-Output</i>
MLP	<i>Multilayer Perceptron</i>
MPP	<i>Maximum Power Point</i>
MRAC	<i>Model Reference Adaptive Control</i>
MRAS	<i>Model Reference Adaptive System</i>
PV	<i>Photovoltaic</i>
PWM	<i>Pulse Width Modulation</i>
RBF	<i>Radial Basis Function</i>
RLS	<i>Recursive Least Squares</i>
RNA	Rede Neural Artificial
SC	Sistema de Controle
SISO	<i>Single-Input Single-Output</i>
SMC	<i>Sliding Mode Control</i>
STR	<i>Self Tuning Regulator</i>
ZOH	<i>Zero Order Hold</i>

Lista de símbolos

α	Constante de momento
b_j	Bias
c	Centro da função gaussiana
e_j	Sinal de erro
η	Taxa de aprendizado
\mathcal{E}	Erro total
J	Função de custo
K	Vetor de ganhos
P	Matriz de covariância
σ	Desvio padrão da função gaussiana em relação ao seu centro
θ	Parâmetro ajustável
φ	Saída da função gaussiana
v_j	Campo local induzido
w_{kj}	Pesos sinápticos
x_j	Sinal de entrada
y_i	Sinal de saída

Sumário

1	INTRODUÇÃO	15
1.1	Problema do Alvo Móvel	16
1.1.1	Caracterização do Problema do Alvo Móvel	16
1.2	Justificativa	17
1.3	Objetivos	18
1.3.1	Objetivo Geral	18
1.3.2	Objetivos Específicos	18
1.4	Organização da Dissertação	18
2	PRELIMINARES	20
2.1	Controle Adaptativo	20
2.1.1	Escalonamento de Ganho	21
2.1.2	Regulador Autoajustável	21
2.1.2.1	Regulador Autoajustável Determinístico	22
2.1.2.2	Algoritmo RLS para Estimção Paramétrica	23
2.1.2.3	Projeto por Alocação de Polos	24
2.1.3	Algoritmo MDPP	25
2.1.4	<i>Timeline</i> e Estado da Arte em Controle Adaptativo	26
2.2	Arquitetura RNA do tipo MLP	29
2.2.1	Redes MLP	29
2.3	<i>Timeline</i> e Estado da Arte em Redes Neurais	31
2.3.1	RNA e Controle Adaptativo	34
2.4	Comentários	34
3	SISTEMAS DE CONTROLE BIOINSPIRADOS	36
3.1	Redes Neurais do tipo RBF	37
3.1.1	Comparação Entre Redes do Tipo RBF e MLP	39
3.1.2	Sistema Adaptativo por Modelo de Referência	40
3.1.2.1	A regra MIT	40
3.1.2.2	Teoria de Estabilidade	41
3.2	Controle de Modos Deslizantes	43
3.3	Comentários	45
4	MODELOS NEURO-ADAPTATIVOS PARA PROJETO <i>ON-LINE</i> DE SC	46
4.1	Modelos Neuro-Adaptativos	47

4.1.1	Controle Adaptativo Baseado em RBF	47
4.1.2	Análise da Estabilidade	48
4.2	Algoritmos <i>Online</i> do SC Proposto	49
4.2.1	Algoritmos do Estimador	50
4.2.2	Algoritmo de Controle	50
4.3	Comentários	52
	5 PROJETO <i>ONLINE</i> DO SC-NEURO-ADAPTATIVO	53
5.1	Problema do Seguidor Solar	54
5.2	Formulação Matemática do Sistema de Controle	55
5.2.1	Modelo Discretizado	56
5.3	Projeto do SC-Neuro-Adaptativo	57
5.3.1	Modelagem da Não-Linearidade do Sistema	58
5.3.2	Identificação da Planta	58
5.3.3	Modelo de Referência	59
5.3.4	Modelo do Atuador-Carga	60
5.3.5	Experimentos Computacionais	61
5.4	Desenvolvimento e Montagem do SC-Neuro-Adaptativo	63
5.4.1	Unidades SC-Neuro-Adaptativo	63
5.4.1.1	Unidade de Sensoriamento	64
5.4.1.2	Unidade de Controle	64
5.4.2	Unidade de Potência	65
5.4.2.1	Circuito Dobrador de Tensão	66
5.4.2.2	Circuito de Disparo	66
5.4.2.3	Circuito da Ponte H	67
5.5	Experimentos em <i>Hardware</i>	68
5.6	Comentários	71
	6 CONCLUSÃO	72
6.1	Trabalhos Futuros	73
	REFERÊNCIAS	74
	APÊNDICES	80
	A – REDES NEURAIS ARTIFICIAIS E SINTONIA	81
A.1	Modelo do Neurônio Não Linear	81
	B – ATUADORES ELETROMECCÂNICOS	87

Introdução

A necessidade de aumentar a produtividade industrial, associada com a qualidade e baixo custo, gerou uma demanda para o desenvolvimento de controladores de alto desempenho. Motivado por esta demanda, apresenta-se nesta dissertação modelos e uma metodologia para o projeto de sistemas de controle de alto desempenho. Tay, Mareels e Moore (1998) apresentam os fundamentos e as aplicações de sistemas de controle de alto desempenho. Os modelos apresentam características de adaptabilidade por meio de arquiteturas de sistemas de controle adaptativo e os parâmetros das estruturas algébricas das citadas arquiteturas são computados pelos métodos de computação bioinspirada em redes neurais artificiais.

De forma a firmar o compromisso com o desenvolvimento de sistemas de controle de alto desempenho, apresenta-se o desenvolvimento de um modelo e procedimentos para o projeto *online* de sistemas de controle adaptativo do tipo *model reference adaptive control* (MRAC). O ajuste ou sintonia para o cálculo da lei de controle é guiado por uma rede neural artificial do tipo *radial basis function* (RBF)

A computação bioinspirada, acrônimo de computação biologicamente inspirada, é constituída de estratégias e algoritmos para serem implementados em computadores a fim de modelar o comportamento dos seres vivos constituindo, dessa forma, um importante subconjunto da computação natural (DUAN; LI, 2014). Todos os algoritmos bioinspirados tais como redes neurais, algoritmos genéticos ou inteligência em enxames tentam reproduzir a forma como os organismos biológicos e bio-organismos (como neurônios e bactérias) operam para atingir um alto nível de eficiência, mesmo se, algumas vezes, a melhor solução real não é alcançada (KAR, 2016).

Neste trabalho optou-se por utilizar a computação bioinspirada em redes neurais artificiais (RNA), pois elas possuem características adicionais aos métodos convencionais para o projeto de sistemas de controle que são baseados na teoria de controle clássica. Segundo Haykin et al. (2009), uma rede neural é um processador com alto grau de paralelismo distribuído constituído de unidades de processamento simples, que têm a

propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso. As principais vantagens do uso de redes neurais são o seu poder computacional, em virtude da sua estrutura maciçamente paralelamente distribuída, e da sua habilidade de aprender e conseqüentemente generalizar.

O controlador aplicado é o controle adaptativo por modelo de referência (MRAC, *model reference adaptive control*). Esse tipo de controlador ajusta automaticamente os seus parâmetros para obter as propriedades desejadas em malha fechada (ÅSTRÖM; WITTENMARK, 2013; LANDAU et al., 2011). O desenvolvimento proposto tem por base as técnicas de medição baseada em modelo, adotada na realimentação das variáveis controladas, e inteligência computacional, sendo empregada no ajuste dos parâmetros.

Os modelos bioinspirados em redes neurais artificiais são desenvolvidos para sintetizar as estruturas algébricas de controladores por realimentação de saída. O treinamento da rede é efetuado inicialmente pelos mínimos quadrados recursivos (RLS, *recursive least squares*) e suas variantes para a sintonia dos parâmetros dos blocos funcionais de um sistema de controle adaptativo proposto. Em seguida o controle é embarcado em um microcontrolador que deve satisfazer as especificações de desempenho em malha fechada estabelecida pelo modelo de referência. O desempenho do sistema é avaliado de modo que a saída do sistema em malha fechada acompanha a dinâmica estabelecida pelo modelo de referência, levando em consideração o intervalo de amostragem de forma a não comprometer o desempenho produtivo e a segurança operacional do sistema.

A rede neural, juntamente com o controle adaptativo, são implementados com o intuito de realizar o controle de um sistema constituído por um painel fotovoltaico. O controle é realizado no sentido de que a trajetória do painel fotovoltaico acompanhe a trajetória de uma fonte luminosa atendendo os requisitos de desempenho em malha fechada

1.1 Problema do Alvo Móvel

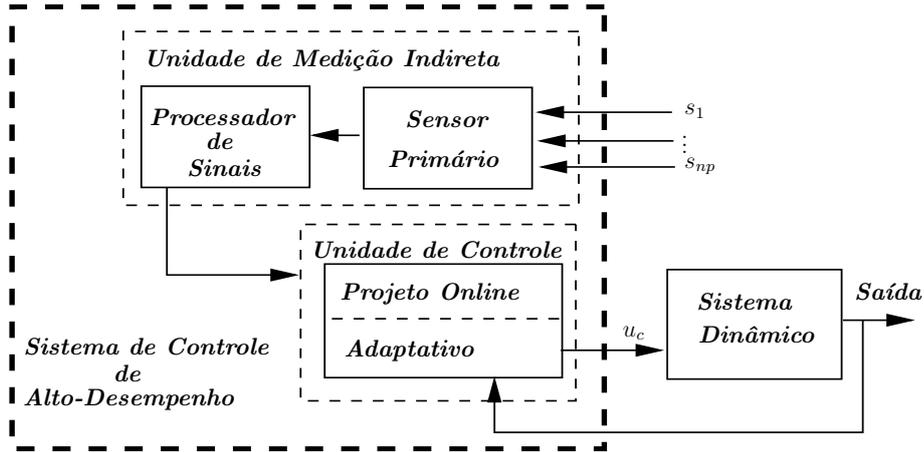
A descrição do problema do alvo móvel e suas soluções são apresentados no contexto de caracterização e solução do problema. A abrangência e a importância do problema são expostos na caracterização do problema do alvo móvel. Os métodos usados para solucionar este problema são apresentados de forma abrangente e de baixo custo.

1.1.1 Caracterização do Problema do Alvo Móvel

O problema consiste em posicionar os dispositivos, tais como: painéis solares, câmeras, para rastrear ou interagir com alvos móveis. Dessa forma, tem-se as seguintes situações: painéis solares que devem ser posicionados para capturar o máximo de energia e câmeras para rastrear invasores em perímetros circulares.

Na Figura 1, apresenta-se o diagrama de blocos do sistema de controle de alto desempenho para rastreamento de alvos móveis. O sinal de referência, proveniente do processador de sinais, é gerado de acordo com a posição do alvo que é dada pelas variações dos sinais s_1, s_2, \dots, s_{np} dos sensores.

Figura 1 – Arquitetura do sistema de controle de alto desempenho



Fonte: Produzindo pelo autor

O problema é formulado a partir dos mapeamentos compostos do espaço das entradas e do espaço de saída que são dados por

$$G : \{s_1, s_2, \dots, s_{np}, T\} \rightarrow (T, x, y, z) \quad (1.1)$$

$$\Theta : (T, x, y, z) \rightarrow \theta_m^{ref} \quad (1.2)$$

em que s_1, s_2, \dots, s_{np} os sinais obtidos no instante T pelos sensores primários, estes sinais são associados com coordenadas x, y e z cartesianas para posicionamento do sistema. As coordenadas x, y e z são processadas para estabelecer a posição de referência, θ_m^{ref} , para o atuador, que compõe o sistema dinâmico.

1.2 Justificativa

O projeto e implementação de controle adaptativo para sistemas com parâmetros variantes no tempo e/ou desconhecidos está imbuído de alto grau de complexidade, pois é necessário escolher a estrutura do modelo e, além disso, existe a presença de variações paramétricas, incertezas de modelagem e perturbações do ambiente. Em vista disto, faz-se uso de técnicas de identificação de sistemas para modelar a relação entrada-saída da planta. O modelo obtido pode ser usado para projetar um controlador adaptativo a ser usado em sistemas com parâmetros desconhecidos e/ou variantes no tempo que atenda métricas de

qualidade operacional, além das que são proporcionadas pelos sistemas em malha fechada. Desta forma, propõe-se o desenvolvimento de sistemas de controle adaptativo sintonizado por redes neurais artificiais.

Em vista da necessidade e da importância do ensino das aplicações de conhecimento teórico em uma situação do mundo real, ainda considerando, as dificuldades de se adquirir plataformas comerciais, apresenta-se como justificativa o desenvolvimento e montagem de uma plataforma para pesquisa e validação experimental para métodos de controle inteligente que estão associados com as abordagens para o projeto *online* de sistemas de controle adaptativo. As abordagens de identificação de sistemas de controle e inteligência artificial compõem uma visão mais ampla, em termos de contextualização, universo no qual pertence esta proposta.

1.3 Objetivos

Os objetivos a serem alcançados neste trabalho estão listados a seguir e agrupados em objetivos gerais e específicos.

1.3.1 Objetivo Geral

De maneira geral, o objetivo é desenvolver modelos para sintonia do projeto *online* de sistemas de controle que são baseados em inteligência computacional. Posteriormente, planeja-se sintetizar o controle dos mesmos utilizando métodos de controle adaptativo.

1.3.2 Objetivos Específicos

- Desenvolver rede neural artificial do tipo RBF para sintonia de sistemas de controle adaptativo;
- Desenvolver algoritmos para projeto *online* de sistemas de controle adaptativo;
- Levantar o modelo da planta por meio de técnicas de identificação de sistemas a partir de dados de entrada e saída da planta;
- Avaliar o desempenho dos modelos matemáticos reduzidos de sistemas dinâmicos do mundo real.

1.4 Organização da Dissertação

A dissertação está organizada em capítulos que apresentam os conceitos básicos dos paradigmas de redes neurais artificiais (RNA), controle adaptativo, *timeline*, estado

da arte e o desenvolvimento da pesquisa juntamente com os resultados obtidos para o projeto *online* de sistema de controle adaptativo para seguir alvos móveis.

No Capítulo 2, conceitos, *timeline* e estado da arte dos paradigmas de controle adaptativo e redes neurais artificiais (RNA) são expostos e comentados no contexto da pesquisa desenvolvida em projeto *online* de um sistema de controle neuro-adaptativo. O referido capítulo é finalizado com o estado da arte de cada paradigma citado, a saber: aprendizado *online* de RNAs, sistemas de controle adaptativo, projeto *online* de sistemas de controle e a interação entre sistemas de controle e RNA.

Os principais métodos dos paradigmas de RNAs e o controle adaptativo são expostos no Capítulo 3. Estes métodos são o sustentáculo teórico das funções de controle e treinamento da RNA para o projeto *online* de sistemas de controle neuro-adaptativo. Apresenta-se, ainda, no Capítulo 3, a integração dos paradigmas de controle adaptativo por modelo de referência e RNA para o projeto *online* de sistemas de controle.

Os modelos propostos para o projeto *online* de sistemas de controle (SC) neuro-adaptativos são apresentados no Capítulo 4. O desenvolvimento do método (procedimento) e algoritmos tem por base a integração entre os paradigmas de rede de base radial e controle, associados com estratégias de controle dos modos deslizantes e aprendizado RBF e RLS.

No Capítulo 5, apresenta-se o desenvolvimento da aplicação dos modelos, desenvolvidos no Capítulo 4, para projeto *online* do SC-neuro-adaptativo. O método é aplicado a um sistema que tem por objetivo seguir um alvo móvel.

No Capítulo 6, são apresentadas as conclusões, comentários e trabalhos futuros que são resultantes desta pesquisa em controle adaptativo e redes neurais para o projeto *online* de sistema de controle.

No Apêndice A, é mostrado o modelo do neurônio não linear juntamente com as principais funções de ativação usadas em redes neurais artificiais. No Apêndice B, é apresentada a modelagem matemática do conjunto atuador-carga no sentido de se obter a função de transferência que relacione a saída e a entrada do sistema.

Preliminares

Os principais conceitos acerca das teorias de controle adaptativo e redes neurais artificiais (RNA) empregadas no projeto *online* de sistemas de controle adaptativo são abordados neste capítulo. A sintonia dos parâmetros deste sistema é baseada em redes neurais artificiais do tipo *radial basis function* (RBF). Estes conceitos originados nas teorias de controle adaptativo e RNA são inseridos no contexto do sistema de controle de alto desempenho para rastreamento de alvos móveis, conforme apresentado na Figura 1.

A primeira técnica de controle adaptativo mostrada é a de escalonamento de ganho. Essa técnica visa selecionar um certo número de pontos de operação que cubram toda a faixa de operação do sistema. Em seguida, é realizada uma discussão sobre o regulador autoajustável, proposto por Åström e Wittenmark (2013), bem como o algoritmo de alocação de polos para o polinômio de grau mínimo.

Expõe-se, ainda, neste capítulo as redes perceptron de múltiplas camadas. Além disso, é apresentado o algoritmo *backpropagation* com a regra de aprendizagem de Widrow-Hoff. Esse algoritmo é usado em redes do tipo perceptron de múltiplas camadas para ajustar os seus parâmetros livres.

2.1 Controle Adaptativo

De acordo com o dicionário Houaiss “adaptar” significa ajustar, acomodar, tornar ou ser adequado a. De forma intuitiva, um controlador adaptativo é um controlador que pode modificar o seu comportamento em resposta à modificações na dinâmica do processo e na natureza do distúrbio. De modo formal, um controlador adaptativo é um controlador com parâmetros ajustáveis e com um mecanismo para ajustar esses parâmetros (ÅSTRÖM; WITTENMARK, 2013).

A técnica de controle adaptativo proporciona o ajuste automático dos parâmetros de controle, em tempo real, a fim de atingir ou manter um nível desejado de desempenho do sistema de controle quando os parâmetros dinâmicos do processo a ser controlado

são desconhecidos e/ou variáveis no tempo. A principal característica desta técnica é a capacidade de extrair informações significativas, a partir de dados reais, a fim de ajustar o controlador e apresentar um mecanismo de ajuste dos parâmetros, seja do modelo de planta ou do controlador (LANDAU et al., 2011)(LANDAU et al., 2017).

Dentre os tipos de sistemas de controle adaptativos mais utilizados três se destacam: escalonamento de ganho, regulador autoajustável e controle adaptativo por modelo de referência, sendo este apresentado no Capítulo 3.

2.1.1 Escalonamento de Ganho

A ideia do escalonamento de ganho é selecionar um certo número de pontos de operação que cubram toda a faixa de operação do sistema. Então, em cada um desses pontos, o projetista faz uma aproximação linear invariante no tempo da dinâmica da planta e projeta um controlador linear para cada planta linearizada. Entre os pontos de operação os parâmetros do compensador podem ser então interpolados, ou escalonados, resultando assim em um compensador global. O escalonamento de ganhos é conceitualmente simples e, de fato, bem-sucedido para várias aplicações. O principal problema com escalonamento de ganho é que ele tem somente garantia teórica limitada de estabilidade em operação não linear. Outro problema é a carga computacional envolvida em um projeto de escalonamento de ganho, devido à necessidade de calcular muitos controladores lineares.

Na Figura 2, tem-se o diagrama de blocos de um sistema com escalonamento de ganho. O escalonamento de ganho pode ser visto como um sistema de controle em malha fechada em que os ganhos da realimentação são ajustados usando compensação no elo de alimentação direta.

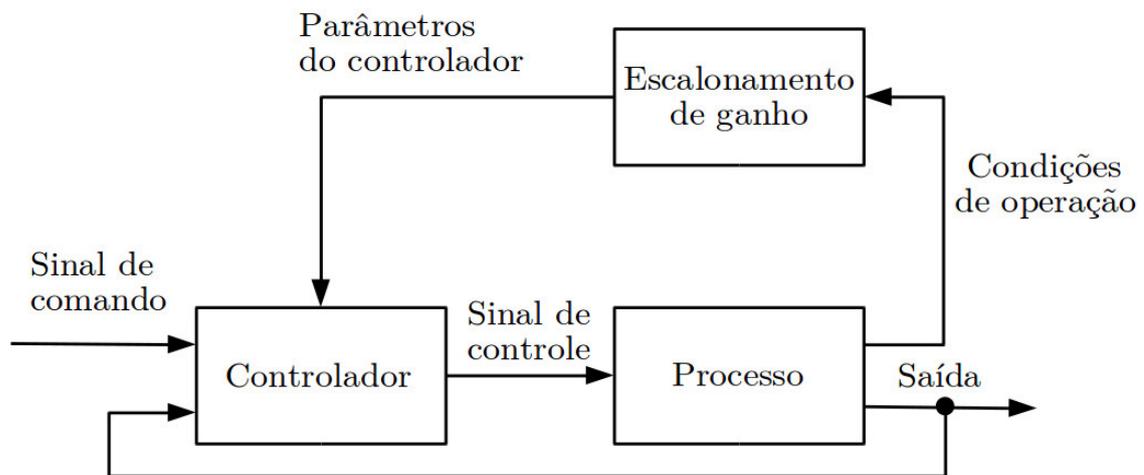
O controlador é sintonizado para cada condição de operação. Devido aos vários valores presentes na tabela de escalonamento deve-se dedicar uma atenção especial à transição entre as diferentes condições de operação.

O escalonamento de ganhos tem a vantagem de que os parâmetros do controlador podem ser alterados muito rapidamente em resposta a mudanças do processo. Como não ocorrem estimativas dos parâmetros, os fatores limitantes dependem da velocidade com que as medições auxiliares respondem às mudanças do processo (ÅSTRÖM; WITTENMARK, 2013).

2.1.2 Regulador Autoajustável

No projeto de sistemas de controle adaptativo é assumido que a estrutura do modelo do processo é especificada. Os parâmetros do modelo são estimados de forma *online* e a estimação pode ser realizada de maneira contínua ou em batelada. A estimação paramétrica pode ser efetuada de diversas formas. Existem também várias técnicas que

Figura 2 – Diagrama de blocos de um sistema com escalonamento de ganho



Fonte: Adaptado de Åström e Wittenmark (2013)

podem ser usadas para o projeto do sistema de controle. Nesta abordagem os parâmetros estimados são tratados como os parâmetros verdadeiros no projeto do controlador. Isto é chamado de princípio da equivalência à certeza (ÅSTRÖM; WITTENMARK, 2013).

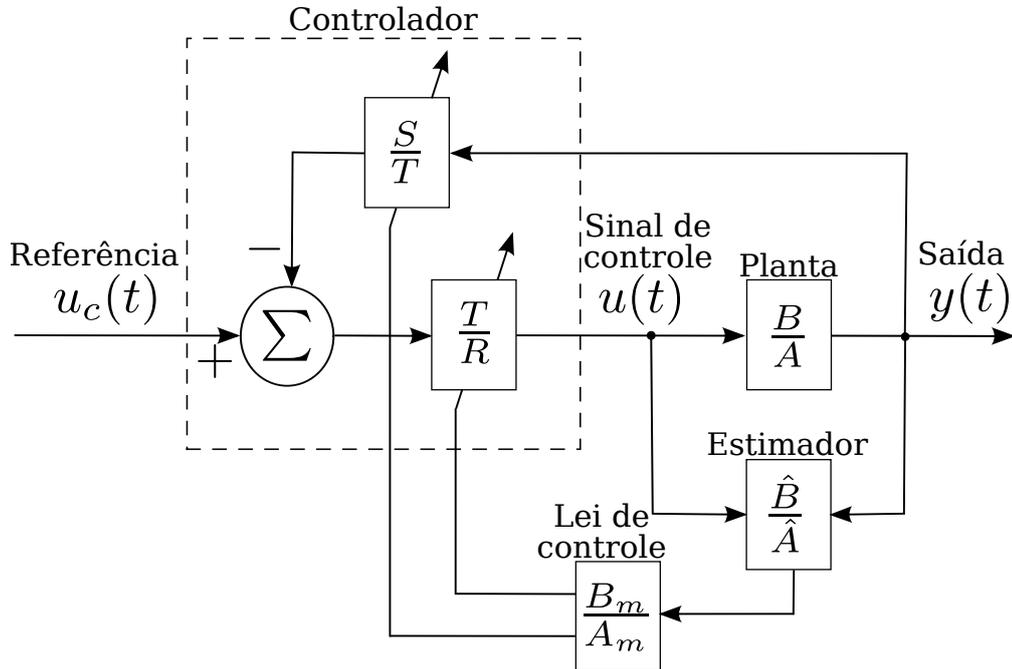
2.1.2.1 Regulador Autoajustável Determinístico

O regulador autoajustável (STR, *self turning regulator*) ajusta automaticamente os seus parâmetros para obter as propriedades desejadas do sistema em malha fechada. A escolha da estrutura do modelo e sua parametrização são importantes práticas para reguladores autoajustáveis. Frequentemente, o modelo pode ser reparametrizado tal que os parâmetros do controlador podem ser estimados diretamente. Isto é, um algoritmo adaptativo direto é obtido. Essa abordagem tem sido chamada de implícita.

Uma abordagem indireta é estimar os parâmetros da função de transferência do processo. Isto resulta em um algoritmo adaptativo indireto. Os parâmetros do controlador não são atualizados diretamente, mas indiretamente via a estimação do modelo do processo. Esse método é chamado frequentemente de explícito. Na Figura 3, é apresentado o diagrama de blocos baseado no regulador autoajustável indireto. É assumido que a estrutura do modelo do processo é especificada.

Os parâmetros do modelo são estimados recursivamente de forma *online* e o bloco “Estimador” fornece uma estimativa dos parâmetros do processo. O bloco “Lei de controle” contém cálculos que são exigidos para realizar o projeto de um controlador com um método especificado e alguns parâmetros que podem ser escolhidos externamente. O bloco “Controlador” é uma implementação do controlador cujos parâmetros são escolhidos do

Figura 3 – Baseado no regulador autoajustável indireto



Fonte: Produzido pelo autor

projeto de controle. Considerações sobre os parâmetros dos blocos que constituem a Figura 3 são realizadas na subseções seguintes.

2.1.2.2 Algoritmo RLS para Estimação Paramétrica

Dentre as várias técnicas existentes de estimação paramétrica, a mais utilizada é a dos mínimos quadrados recursivo (RLS). As equações do RLS são dadas por

$$K(k) = \frac{P(k-1)\varphi(k-1)}{1 + \varphi^T(k-1)P(k-1)\varphi(k-1)} \quad (2.1)$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)(y(k) - \varphi^T(k-1)\hat{\theta}(k-1)) \quad (2.2)$$

$$P(k) = (I - K(k)\varphi^T(k-1))P(k-1) \quad (2.3)$$

em que K é o vetor de ganhos; P a matriz de covariância do erro; φ o vetor de regressores e $\hat{\theta}$ é a estimação do vetor de parâmetros θ .

Considerando que o sistema dinâmico é modelado por uma função de transferência da seguinte forma

$$G(z) = \frac{Y(z)}{X(z)} = \frac{b_0z + b_1}{z^2 + a_1z + a_2} \quad (2.4)$$

em termos de equação a diferenças o sistema dinâmico é modelado por

$$y(k)z^2 + a_1zy(k) + a_2y(k) = b_0zu(k) + b_1u(k) \quad (2.5)$$

Multiplicando a Equação (2.5) por q^{-2} resulta em

$$y(k) + a_1y(k-1) + a_2y(k-2) = b_0u(k-1) + b_1u(k-2) \quad (2.6)$$

e a Equação (2.6) pode ser reescrita como

$$y(k) = \varphi^T(k-1)\theta \quad (2.7)$$

em que

$$\theta^T = (a_1 \ a_2 \ b_0 \ b_1)$$

e

$$\varphi^T(k-1) = (-y(k-1) \ -y(k-2) \ u(k-1) \ u(k-2))$$

2.1.2.3 Projeto por Alocação de Polos

Um dos métodos de projeto de controle usado no STR é o de alocação de polos. A ideia é projetar um controlador que forneça os polos desejados em malha fechada. O processo pode ser descrito por um sistema de entrada única e saída única (SISO) expresso por

$$Ay(t) = B(u(t) + v(t)) \quad (2.8)$$

em que A e B são polinômios em z e são relativamente primos.

De acordo com a Figura 3, pode-se escrever um controlador linear geral da seguinte forma

$$Ru(t) = Tu_c(t) - Sy(t) \quad (2.9)$$

em que R , S e T são polinômios que compõem o bloco do controlador na Figura 3.

A eliminação da lei de controle, u , entre as Equações (2.8) e (2.9) resulta nas equações para o sistema em malha fechada, que são dadas por

$$y(t) = \frac{BT}{AR + BS}u_c(t) + \frac{BR}{AR + BS}v(t) \quad (2.10)$$

e

$$u(t) = \frac{AT}{AR + BS}u_c(t) - \frac{BS}{AR + BS}v(t) \quad (2.11)$$

O polinômio característico em malha fechada é dado por

$$AR + BS = A_c \quad (2.12)$$

A ideia chave no método de projeto é especificar o polinômio característico desejado em malha fechada A_c . Os polinômios R e S podem ser resolvidos da Equação (2.12) que é chamada de equação de Diophantine ou equação de Bezout.

Para determinar o polinômio T , na Equação (2.9), é requerido que a resposta do sinal de comando, u_c , para a saída seja descrita pela dinâmica desejada em malha fechada (modelo de referência) expressa por

$$A_m y_m(t) = B_m u_c(t) \quad (2.13)$$

em que, segue das Equações (2.10) e (2.11), que as condições resultam na relação dada por

$$\frac{BT}{AR + BS} = \frac{BT}{A_c} = \frac{B_m}{A_m} \quad (2.14)$$

em que o modelo de referência depende do modelo, sistema e do sinal de comando.

A Equação (2.14) implica que existe cancelamento de fatores de BT e A_c . Fatorando o polinômio B como

$$B = B^+ B^- \quad (2.15)$$

em que B^+ é um polinômio mônico cujos zeros são estáveis e tão bem amortecidos que podem ser cancelados pelo controlador e B^- corresponde a fatores instáveis ou fracamente amortecido que não pode ser cancelado. Segue que B^- deve ser um fator de B_m expresso por

$$B_m = B^- B'_m \quad (2.16)$$

desde que B^+ é cancelado, ele deve ser um fator de A_c . O polinômio característico em malha fechada é dado por

$$A_c = A_o A_m B^+ \quad (2.17)$$

Uma vez que B^+ é um fator de B e A_c , segue da Equação (2.12) que

$$R = R' B^+ \quad (2.18)$$

e a equação de Diophantine é reduzida a

$$AR' + B^- S = A_o A_m = A'_c \quad (2.19)$$

Introduzindo as Equações (2.15), (2.16) e (2.17) na Equação (2.9) resulta no polinômio T , que é dado por

$$T = A_o B'_m \quad (2.20)$$

2.1.3 Algoritmo MDPP

O algoritmo de alocação de polos de grau mínimo (MDPP) descrito em Åström e Wittenmark (2013) tem como resultado os polinômios R , S e T do controlador autoajustável. Esses polinômios possuem grau mínimo e são calculados por meio de alocação de polos. A seguir, tem-se o algoritmo MDPP:

Dados: Polinômios A , B .

Especificações: Polinômios A_m , B_m , e A_o .

Condições de Compatibilidade:

$$\begin{aligned} \text{grau } A_m &= \text{grau } A \\ \text{grau } B_m &= \text{grau } B \\ \text{grau } A_o &= \text{grau } A - \text{grau } B^+ - 1 \\ B_m &= B^- B'_m \end{aligned}$$

Passo 1: Fatorar B como $B = B^+ B^-$, em que B^+ é mônico

Passo 2: Encontrar a solução R' e S com $\text{deg } S < \text{deg } A$ de

$$AR' + B^- S = A_o A_m$$

Passo 3: Formar $R = R' B^+$ e $T = A_o B'_m$, e calcular o sinal de controle da lei de controle

$$Ru = Tu_c - Sy$$

Sem cancelamento de zeros: $B^+ = 1$, $B^- = B$, e $B_m = \beta B$, em que $\beta = A_m(1)/B(1)$. Além disso, $\text{deg } A_o = \text{deg } A - \text{deg } B - 1$ e $T = \beta A_o$. O polinômio característico em malha fechada é $A_c = A_o A_m$, e a equação de Diophantine no **Passo 2** resulta em

$$AR + BS = A_c = A_o A_m$$

O método descrito nas Seções 2.1.2 e 2.1.3 foi implementado em um sistema de acompanhamento da trajetória do sol descrito em Machado, Santos e Neto (2016) como fruto desta pesquisa.

2.1.4 *Timeline* e Estado da Arte em Controle Adaptativo

No início dos anos 1950, as pesquisas em controle adaptativo começaram os primeiros desenvolvimentos significativos teóricos e experimentais. Por exemplo: o projeto de pilotos automáticos para aeronaves de alto desempenho. Nesse tipo de sistema a dinâmica muda drasticamente quando a aeronave passa de um ponto de operação para outro. Em Whitaker, Yamron e Kezer (1958) é sugerido um controle adaptativo por modelo de referência (MRAC, *model reference adaptive control*) para resolver o problema do controle automático de aeronaves. Foram usadas leis adaptativas de vários esquemas adaptativos propostos. A regra MIT foi usada para projetar as leis adaptativas dos vários esquemas de controle adaptativos propostos. Em Kalman (1958), foi proposto um esquema adaptativo por alocação de polos baseado no problema do controle linear quadrático ótimo.

Nos anos de 1960 foi o período mais importante para o desenvolvimento da teoria de controle adaptativo. Técnicas de espaço de estados e teoria de estabilidade baseada em

Lyapunov foram introduzidas. Desenvolvimentos em programação dinâmica (BELLMAN, 1957), controle dual (FEL'DBAUM, 1965), controle estocástico de modo geral e em identificação de sistemas e estimação paramétrica (ÅSTRÖM; EYKHOFF, 1971), (TSYPKIN; NIKOLIC, 1971) desempenharam um papel primordial na formulação do controle adaptativo. Em 1966, as leis adaptativas baseadas em regras do MIT usadas nos esquemas de MRAC dos anos 1950, foram reprojatadas aplicando a abordagem de projeto de Lyapunov.

Os avanços na teoria da estabilidade e o progresso na teoria de controle nos anos 1960, melhoraram o entendimento do controle adaptativo e contribuíram para um forte e renovado interesse neste campo de pesquisa nos anos 1970. Por outro lado, o desenvolvimento simultâneo e progressivo em computadores e componentes eletrônicos possibilitaram a implementação de controladores complexos e contribuíram para o aumento do interesse em aplicações de controle adaptativo. Nos anos 1970 foram obtidos vários avanços no projeto de controladores adaptativos. Esquemas MRAC usando a abordagem de projeto de Lyapunov foram projetados e analisados em (EGARDT, 1979), (MORSE, 1980), (NARENDRA; LIN; VALAVANI, 1980). Os conceitos de positividade e hiper-estabilidade foram usadas em (LANDAU, 1979) para desenvolver uma ampla classe de esquemas MRAC com propriedades de estabilidade bem estabelecidas. Ao mesmo tempo, esforços paralelos para plantas de tempo discreto em um ambiente determinístico e estocástico produziram várias classes de esquemas de controle adaptativo com rigorosas provas de estabilidade (GOODWIN; RAMADGE; CAINES, 1980), (GOODWIN, 1984). As pesquisas da década de 1970 e o desenvolvimento de uma ampla classe de esquemas de controle adaptativo com propriedades de estabilidade bem estabelecidas foram acompanhadas por várias aplicações bem-sucedidas (HARRIS; BILLINGS, 1981), (NARENDRA; MONOPOLI, 1980).

A abordagem de controle adaptativo indireto foi desenvolvida significativamente a partir de Åström e Wittenmark (1973) onde o termo “autoajuste” foi idealizado. O esquema resultante correspondeu a uma versão adaptativa do controle discreto de variância mínima. Um outro desenvolvimento aparece em Clarke e Gawthrop (1975). De fato, o controlador de variância mínima autoajustável e suas extensões são esquemas de controle adaptativo direto, uma vez que se estima diretamente os parâmetros do controlador. Foram necessários vários anos para entender que os sistemas de controle adaptativo por modelo de referência de tempo discreto e os reguladores estocásticos de autoajuste baseados na minimização da variância de erro pertencem a mesma família (EGARDT, 1979), (LANDAU, 1982).

Entretanto, os sucessos da década de 1970, foram seguidos por controvérsias sobre a viabilidade do controle adaptativo. No ano de 1979, foi apontado que os esquemas adaptativos da década de 1970 poderiam facilmente se tornar instáveis na presença de pequenos distúrbios (EGARDT, 1979). O comportamento não-robusto do controle adaptativo tornou-se muito controverso no início dos anos 80, quando mais exemplos de instabilidades foram publicados demonstrando falta de robustez na presença de dinâmica não modelada

ou distúrbios limitados (IOANNOU, 1983), (ROHRS et al., 1985). Isso estimulou muitos pesquisadores, cujo objetivo era compreender os mecanismos de instabilidade e encontrar formas de contorná-los. Em meados da década de 1980, várias novas modificações foram propostas e analisadas, levando a um conjunto de trabalhos conhecido como controle adaptativo robusto. O trabalho sobre o controle adaptativo robusto continuou ao longo dos anos 80 e envolveu a compreensão das várias modificações de robustez e sua unificação sob um âmbito mais geral (EGARDT, 1979), (IOANNOU; SUN, 1988), (IOANNOU; DATTA, 1991).

A solução do problema de robustez no controle adaptativo levou à solução do problema de controle de uma planta linear cujos parâmetros são desconhecidos e mudam com o tempo. Até o final da década de 1980 e começo da década subsequente vários resultados inovadores foram publicados na área de controle adaptativo para plantas lineares variantes no tempo (MIDDLETON; GOODWIN, 1988), (GU; SHAO, 1993), (TSAKALIS; LIMANOND, 1994).

O foco da pesquisa de controle adaptativo no final dos anos 1980, ao início dos anos 1990, foi sobre as propriedades de desempenho e sobre a extensão dos resultados da década de 1980 a certas classes de plantas não-lineares com parâmetros desconhecidos. Esses esforços levaram a novas classes de esquemas adaptativos, motivados pela teoria de sistemas não-lineares (KANELLAKOPOULOS, 1992), (KANELLAKOPOULOS; KOKOTOVIC; MORSE, 1991) bem como a esquemas de controle adaptativo com melhor desempenho transitório e em regime permanente (DATTA; IOANNOU, 1991), (SUN, 1993).

Atualmente o controle adaptativo ainda têm sido bastante estudado. O foco principal está em plantas que apresentam não-linearidades, variação paramétrica, múltiplas entradas e múltiplas saídas e queira-se obter ou manter um certo nível de desempenho do sistema de controle em malha fechada quando os parâmetros dinâmicos do processo a ser controlado são desconhecidos e/ou variáveis no tempo.

Em Moreno-Valenzuela et al. (2016) é empregado o controle adaptativo, juntamente com redes neurais, para o objetivo de rastreamento de trajetória de um pêndulo Furuta, que é um sistema subatuado de dois graus de liberdade. Em Zhang et al. (2016a) desenvolve-se um controlador de rastreamento adaptativo para pontes rolantes de pêndulo duplo sujeito a incertezas paramétricas e perturbações externas. Em uma aplicação prática, as pontes rolantes são normalmente sujeitas a incertezas de parâmetros do sistema, como massas de carga útil incertas, comprimentos de cabo, fricções e distúrbios externos, como resistência ao ar. A maioria dos métodos de controle de guindaste existentes tratam o balanço de carga útil como a de um único pêndulo. No entanto, certos tipos de carga útil e mecanismos de içamento resultam em dinâmica de pêndulo duplo. Os efeitos de pêndulo duplo fazem com que a maioria dos métodos de controle de guias existentes não funcionem normalmente. O método de controle de rastreamento adaptativo proposto garante que o

erro de rastreamento do carrinho está sempre dentro de um conjunto anterior de condições de contorno e converge para zero rapidamente. A estabilidade assintótica do ponto de equilíbrio do sistema em malha fechada é assegurada pelas técnicas de Lyapunov e Lemma de Barbalat.

2.2 Arquitetura RNA do tipo MLP

As redes neurais RBF bem como as redes MLP (*multilayer perceptron*) são classificadas como RNA do tipo *feedforward*. Em virtude das redes RBF e MLP terem uma estrutura *feedforward*, nesta seção apresentam-se os principais conceitos deste tipo de arquitetura. No Apêndice A, apresentam-se os conceitos básicos de RNA necessários para uma melhor compreensão do modelo proposto, bem como os conceitos de redes neurais e o neurônio não-linear com as principais funções de ativação aplicadas em redes neurais artificiais.

2.2.1 Redes MLP

As redes de múltiplas camadas alimentadas adiante são uma importante classe de redes neurais. Dentre as camadas que as compõem tem-se a camada de entrada que é composta por um conjunto de unidades sensoriais (nós de fonte), uma ou mais camadas ocultas de nós computacionais e uma camada de saída de nós computacionais. Esse tipo de arquitetura é conhecida como *feedforward* em que o fluxo de sinal é o seguinte: o sinal de entrada se propaga, camada por camada, da camada de entrada para a camada de saída. Essas redes são chamadas de perceptrons de múltiplas camadas.

As redes MLP têm sido aplicadas para resolver diversos problemas. O seu treinamento é realizado de forma supervisionada através de um algoritmo muito popular conhecido como algoritmo de retropropagação de erro (*error backpropagation*). Este algoritmo é alicerçado na regra de aprendizagem por correção de erro (SILVA et al., 2017).

O algoritmo *backpropagation* consiste de duas etapas através das diferentes camadas da rede: um passo para frente, a propagação, e um passo para trás, a retropropagação. Na etapa de propagação, um vetor de entrada é aplicado às unidades sensoriais da rede e o seu resultado se propaga ao longo da rede, camada por camada. Com isso, um conjunto de saídas é produzida como a resposta da rede. Durante a etapa de propagação os parâmetros livres da rede são todos fixos. Ao longo do passo de retropropagação, os parâmetros livres são todos ajustados de acordo com uma regra de correção de erro. O sinal de erro é obtido como a saída da rede subtraída da resposta desejada. Esse sinal de erro é então retropropagado, camada por camada, da camada de saída para a camada de entrada. O ajuste dos parâmetros livres é feito com o objetivo de que a resposta da rede fique mais próxima da resposta desejada.

Os três pontos a seguir destacam as características básicas dos perceptrons multi-camadas:

1. O modelo de cada neurônio da rede inclui uma função de ativação não linear que é diferenciável. Uma função de ativação normalmente utilizada é a função logística

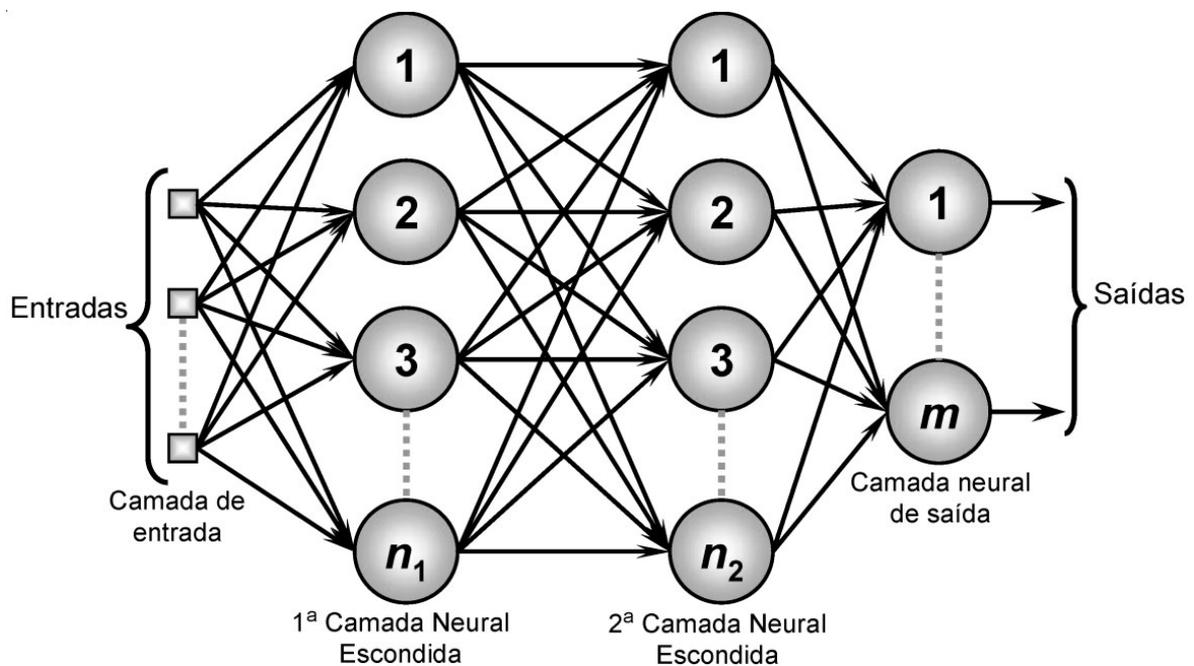
$$y_i = \frac{1}{1 + \exp(-v_j)} \quad (2.21)$$

em que v_j é o campo local induzido (soma ponderada de todas as entradas sinápticas acrescidas do bias) do neurônio j , e y_i é a saída do neurônio.

2. A rede abrange uma ou mais camadas ocultas de neurônios que não fazem parte da entrada e nem da saída da rede.
3. A rede apresenta um alto grau de conectividade, determinado pelas sinapses da rede.

É por meio da combinação dessas características, concomitantemente com a habilidade de aprender da experiência através de treinamento, que advém o poder computacional dos perceptron de múltiplas camadas (HAYKIN et al., 2009).

Figura 4 – Arquitetura de uma rede perceptron de múltiplas camadas com duas camadas ocultas



Fonte: Silva et al. (2017)

Na Figura 4, apresenta-se o gráfico arquitetural de um perceptron de múltiplas camadas com duas camadas ocultas e uma camada de saída. Por meio da Figura 4,

percebe-se que a rede é completamente conectada. Ou seja, um neurônio de cada camada da rede está conectada a um neurônio da camada anterior.

Neste tipo de rede existem dois tipos de sinais: os sinais funcionais e os sinais de erro. Um sinal funcional é um sinal de entrada (estímulo) que incide nas unidades sensoriais da rede, propaga-se para frente (neurônio por neurônio) através da rede e surge na camada de saída da rede como um sinal de saída.

A seguir, descreve-se o algoritmo *backpropagation*. O sinal de erro na saída do neurônio j , na iteração n , é expresso por (HAYKIN et al., 2009)

$$e_j(n) = d_j(n) - y_j(n) \quad (2.22)$$

O erro total é obtido somando todos os termos $\frac{1}{2}e_j^2(n)$ de todos os neurônios da camada de saída, de acordo com

$$\mathcal{E}(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n) \quad (2.23)$$

em que o conjunto C inclui todos os neurônios da camada de saída da rede.

O campo local induzido, $v_j(n)$, produzido na entrada da função de ativação associada ao neurônio j , é dado por

$$v_j = \sum_{i=0}^m w_{ji}(n)y_i(n) \quad (2.24)$$

em que m o número total de entradas; $w_{ji}(n)$ é a conexão entre dois neurônios da rede.

Usando a regra de aprendizagem de Widrow-Hoff, a mesma pode ser escrita como

$$\Delta w_{ji}(n) = \alpha \Delta w_{ji}(n-1) + \eta e_j(n)y_i(n) \quad (2.25)$$

em que α um número positivo chamado constante de momento e η é uma constante positiva chamada de constante de aprendizagem. A taxa de aprendizagem, η , está diretamente relacionada com a taxa de variação dos parâmetros livres da rede, de uma iteração para a outra. Uma taxa de aprendizado grande pode levar a rede à instabilidade, por isso, o termo momento, α , é incluído, para evitar o perigo de instabilidade. No Capítulo 3, apresenta-se a RNA do tipo RBF, bem como realiza-se uma comparação entre as redes do tipo MLP e RBF.

2.3 *Timeline* e Estado da Arte em Redes Neurais

As primeiras pesquisas sobre redes neurais biologicamente inspiradas começaram no ano de 1943, quando McCulloch e Pitts (1943) publicaram o primeiro trabalho relacionado à neurocomputação. Este trabalho resultou na primeira idealização do neurônio artificial,

pois nele os autores realizaram a primeira modelagem matemática inspirada no neurônio biológico.

Já o primeiro método de treinamento de redes neurais foi proposto por Hebb (1949), que em sua homenagem ficou conhecida como regra de aprendizado de Hebb. Esta regra é baseada em suposição e observações de natureza neurofisiológica.

Dentre os inúmeros trabalhos que surgiram, desde então, se destaca o trabalho de Frank Rosenblatt, que durante o biênio de 1957 e 1958, desenvolveu o primeiro neurocomputador, concebendo o modelo básico do Perceptron (ROSENBLATT, 1958).

O modelo desenvolvido por Rosenblatt (1958) estimulou o interesse da comunidade científica devido à sua habilidade em reconhecer padrões simples. Em virtude disso, Widrow, Hoff et al. (1960) desenvolveram um tipo de rede chamada de Adaline – *adaptive linear*. Posteriormente, eles propuseram a Madaline, a Adaline múltipla. Essas redes têm o seu aprendizado alicerçado na regra Delta, que é conhecida como algoritmo de aprendizado do gradiente estocástico (LMS, *least mean square*).

Porém, em Minsky e Papert (1969), os autores demonstraram de forma acentuada a limitação das redes neurais artificiais, formadas por uma camada, em aprender a relação entre entradas e saídas de funções lógicas simples como a XOR. Nesta publicação, os autores demonstraram a impossibilidade das RNAs efetuarem a classificação correta de padrões para classes não linearmente separáveis.

Desde então, as publicações sobre redes neurais foram em pequenas quantidades. Mas os trabalhos foram retomados com Hopfield (1982). Neste trabalho o autor escreve acerca de redes recorrentes fundamentadas em funções de energia.

A retomada definitiva dos estudos sobre redes neurais se deu por vários fatores: computadores com maior capacidade de processamento, novas descobertas sobre o sistema nervoso, algoritmos de otimização mais eficientes. Porém, destaque especial deve ser dado com a publicação de Rumelhart et al. (1986), em que os autores desenvolveram o algoritmo denominado *backpropagation*. Esse algoritmo permite o ajuste dos parâmetros livres de uma rede neural com mais de uma camada possibilitando, inclusive, solucionar o problema de aprendizado de padrões da função lógica XOR.

Com a retomada das pesquisas sobre redes neurais muitos trabalhos foram publicados com diferentes aplicações e diversas propostas de algoritmos. Uma descrição mais detalhada sobre o histórico das redes neurais, desde a origem, com a evolução das mesmas pode ser encontrada em (HAYKIN et al., 2009).

No trabalho descrito em Zhang et al. (2016b), os autores têm o objetivo de resolver o problema de controle de trajetória de manipuladores robóticos coordenados. Para lidar com as incertezas e instabilidade de manipuladores robóticos coordenados e melhorar o desempenho do sistema com restrição de saída, eles projetaram um controlador usando

a rede neural de função de base radial que tem a capacidade de aproximar quaisquer funções limitadas e contínuas efetivamente. Uma barreira de função de Lyapunov também é introduzida para evitar a violação da restrição de saída.

Em pesquisas recentes os algoritmos inteligentes tem sido aplicados em várias áreas de conhecimento. Como destaque tem-se algoritmos genéticos (ASADI et al., 2016), otimização de enxames de partículas (ISHAQUE; SALAM, 2013), colônia de formigas (WANG; GAO; QIU, 2015) e algoritmo do embaralhamento dos sapos saltitantes (BHATTACHARJEE; SARMAH, 2014), redes neurais bioinspiradas (ZHOU et al., 2013), mapa auto-organizado (LEE et al., 2016) e lógica *fuzzy* (HO; CHEN, 2016). Neste trabalho é utilizado redes neurais artificiais como algoritmo inteligente.

O potencial das RNAs permite a solução de vários tipos de problemas provindos das mais diversas áreas de conhecimento possíveis. É o caso do setor automotivo, em que se utiliza redes neurais artificiais para realizar a detecção e classificação de falhas em motores de combustão interna utilizando dados de vibração do ângulo do virabrequim (AHMED et al., 2015). Outra aplicação apresentada em Guo et al. (2014) apresenta um estudo sobre o controle lateral no rastreamento de trajetórias de veículos inteligentes para sistemas rodoviários automatizados. Neste artigo o controlador para o rastreo da trajetória é projetado em função do modelo dinâmico lateral do veículo. O modelo dinâmico é produzido por uma rede neural de função de base radial baseada em uma estrutura variável de modos deslizantes.

Na área de energias renováveis é possível aplicar técnicas baseadas em redes neurais para diagnóstico de faltas em sistemas fotovoltaicos (CHINE et al., 2016). Outra abordagem descrita em Punitha, Devaraj e Sakthivel (2013) é no rastreamento do ponto de máxima potência (MPP, *maximum power point*) quando o sistema está sob condição de sombreamento parcial. Isso gera a presença de múltiplos picos nas características de tensão de alimentação que devem ser considerados pelo algoritmo proposto.

Outra área importante é a de economia e finanças. Essa área apresenta problemas de difícil tratamento devido ao comportamento não-linear desses sistemas. A criação de um sistema inteligente que pode prever com precisão o preço das ações de forma robusta sempre foi um tema de grande interesse para muitos investidores e analistas financeiros. As redes neurais têm ajudado na criação desse sistema por meio da predição do preço das ações (HAFEZI; SHAHRABI; HADAVANDI, 2015) e na previsão de séries temporais de preços financeiros (NIU; WANG, 2014).

Aplicações de redes neurais também podem ser encontradas na área de robótica para realizar o controle de um manipulador robótico de n -elos com restrições nos estados (HE; CHEN; YIN, 2016). Além disso, em He et al. (2016) é realizado o controle de um manipulador robótico com zona morta na entrada e restrições na saída. A rede neural é usada para aproximar a função de zona morta e o modelo desconhecido do manipulador

robótico. Os trabalhos com manipuladores robóticos são desafiadores, pois esses sistemas apresentam dinâmicas não-lineares. Adicionalmente, eles apresentam incertezas nos parâmetros, são sistemas com múltiplas entradas múltiplas saídas (MIMO, *multiple-input multiple-output*) e apresentam vibrações, quando se trata de manipuladores flexíveis, que podem ser suprimidas por meio de um controle neural (SUN; HE; HONG, 2016).

2.3.1 RNA e Controle Adaptativo

O estudo realizado em Zhao et al. (2016) investiga o controle de realimentação de estado da rede neural adaptativa e observação robusta para um sistema de suspensão ativa que considera incertezas paramétricas, distúrbios rodoviários e saturação do atuador. Uma vez que é difícil obter estados precisos na prática, um observador neural é desenvolvido para fornecer estimativa de estado utilizando os dados de entrada e saída medidos do sistema. Sun, Gao e Kaynak (2013) propõem uma estratégia de controle adaptativo para suspensões ativas de veículos com restrições rígidas. Um controlador adaptativo é projetado para estabilizar a posição do veículo e, entretanto, melhorar o conforto de condução na presença de incertezas de parâmetros, onde os espaços de suspensão, as cargas dinâmicas dos pneus e as saturações do atuador são consideradas restrições no domínio do tempo. Além disso, está prevista uma trajetória de referência para manter os movimentos verticais e de inclinação da carroceria para estabilizar em um tempo predeterminado, o que ajuda a ajustar as acelerações de acordo com níveis elevados ou baixos para melhorar o conforto de condução.

Em Castañeda et al. (2016) é apresentado um esquema de controle para a regulação e rastreamento de trajetórias, apesar das perturbações externas em um helicóptero de três graus de liberdade. O esquema permite reduzir o esforço de controle, redução de vibração e rastreamento preciso. A abordagem proposta combina um diferenciador contínuo com um controlador adaptativo. O diferenciador fornece estimativa de estado, enquanto que o algoritmo adaptativo é escolhido de tal forma que os ganhos do controlador são adaptados para reduzir o esforço de controle e não requerem o conhecimento dos limites das incertezas e perturbações.

Muitas outras aplicações de controle adaptativo podem ser encontradas em diversos locais. E, por meio das aplicações citadas, verifica-se que existe uma relação estreita entre o controle adaptativo e as redes neurais artificiais de forma que em muitos dos trabalhos mencionados as duas áreas estão presentes.

2.4 Comentários

Neste capítulo foram apresentadas diversas abordagens em controle adaptativo, tais como escalonamento de ganho e o regulador autoajustável na versão determinística.

Em seguida foi mostrado o algoritmo de estimação paramétrica e o projeto por alocação de polos, bem como o algoritmo MDPP. Ainda, foram exibidos um breve histórico sobre redes neurais artificiais e outro sobre controle adaptativo. O capítulo é finalizado com o estado da arte sobre redes neurais artificiais e controle adaptativo.

Sistemas de Controle Bioinspirados

Os sistemas de controle bioinspirados em redes neurais artificiais são abordados, sob o ponto de vista de controle adaptativo e RNA do tipo função de base radial, com o objetivo de desenvolver modelos e algoritmos para o projeto *online* de controladores neuro-adaptativos.

As redes neurais artificiais são técnicas computacionais motivadas ou inspiradas no funcionamento do cérebro humano. No começo, o trabalho sobre redes neurais foi motivado pela observação de que o cérebro humano processa informações de uma forma completamente diferente de um computador digital tradicional (HAYKIN et al., 2009). Baseado nessa motivação, o desenvolvimento de RNAs proporcionaram diversas orientações para a elaboração de algoritmos de aprendizagem e adaptabilidade (SILVA et al., 2017).

Enquanto os computadores convencionais funcionam de forma sequencial, proporcionando maior eficiência na resolução de tarefas nas quais devem ser obedecidas etapas, o cérebro humano funciona de maneira paralela, e sendo intensamente conectado é mais eficiente na resolução de incumbências que exigem diversas variáveis (KAR, 2016).

O fato que difere as máquinas fundamentadas em redes neurais dos computadores atuais é que estes baseiam seu processamento explicitamente em modelos matemáticos. Já a forma neural, contudo, não é baseada em modelos, mas utiliza cálculos matemáticos para efetuar suas operações e dessa forma podem coordenar diversos graus de liberdade durante a execução de tarefas e frente a diversos ambientes (SILVA et al., 2017).

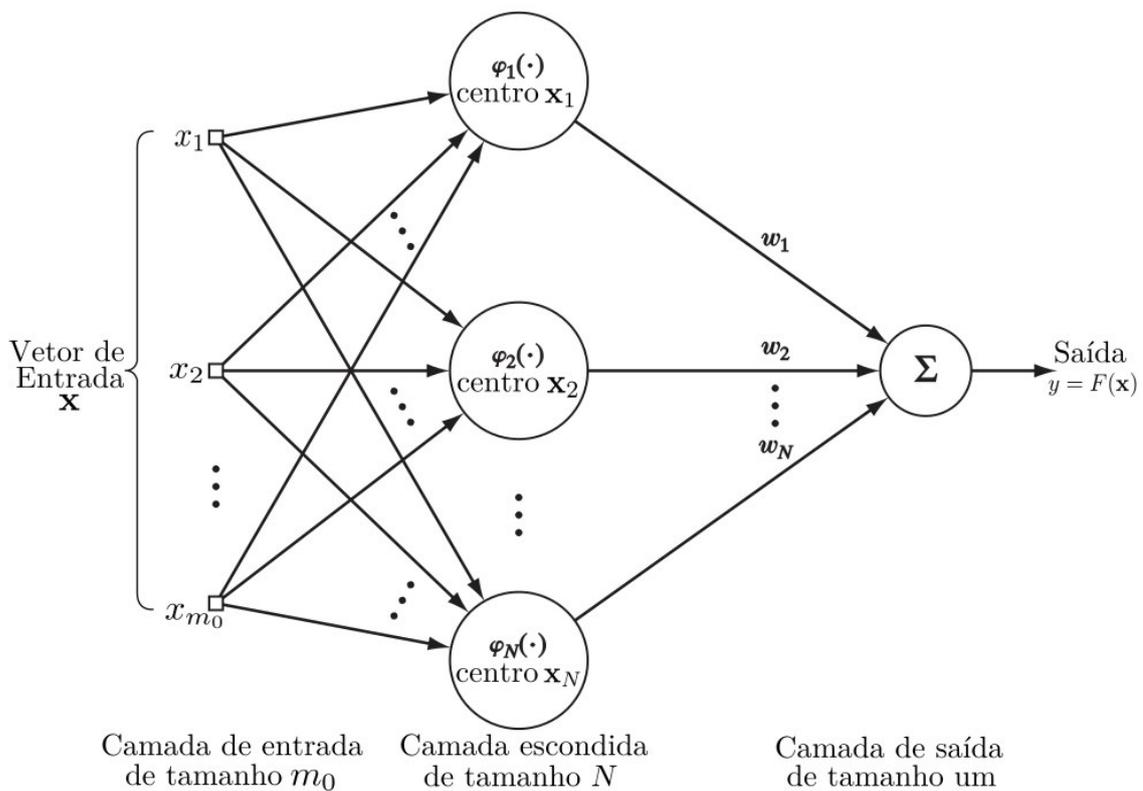
Dentre os muitos tipos de redes neurais artificiais existentes, duas se destacam em função da quantidade de trabalhos em que são usadas, as redes perceptron de múltiplas camadas e as redes de função de base radial. A seguir, abordam-se teorias, métodos e algoritmos em redes neurais artificiais e controle adaptativo para o projeto *online* de controladores. Os tópicos abordados constituem o fundamento para o desenvolvimento desta pesquisa: arquiteturas de RNA (redes MLP e RBF); sistema adaptativo por modelo de referência; controle de modos deslizantes e controle adaptativo baseado em RBF. Estes

tópicos são agrupados com o propósito de formar os modelos para a solução dos problemas de alvos móveis.

3.1 Redes Neurais do tipo RBF

As redes de função de base radial podem ser empregadas em quase todos os tipos de problemas tratados pelo MLP, inclusive aqueles que envolvem aproximação de funções e classificação de padrões. A diferença básica entre elas é que a rede MLP pode ser composta por mais de uma camada oculta, enquanto que a estrutura básica da rede RBF é composta por apenas uma camada oculta, como pode ser visto na Figura 5.

Figura 5 – Estrutura de uma rede de funções de base radial



Fonte: Haykin et al. (2009)

As redes RBF, assim como as MLP, possuem uma arquitetura *feedforward* de camadas múltiplas, cujo treinamento é efetivado de forma supervisionada. A informação é apresentada à entrada da rede, passa pela camada oculta e é finalizada na camada neural de saída.

A função de ativação das redes RBF, geralmente, são as Gaussianas. Para esse tipo de função de base radial quanto mais próximo do centro estiver o estímulo, mais próximo

da unidade será a resposta da função de ativação.

Uma rede de função de base radial é formada por três camadas que possuem funções distintas:

1. A camada de entrada é formada por unidades sensoriais que conectam a rede ao ambiente em que ela está inserida;
2. A segunda camada, única camada oculta existente na rede, executa uma transformação não linear do espaço de entrada para um espaço oculto;
3. A camada de saída é linear, projetada para fornecer a resposta da rede a um padrão de ativação aplicado à camada de entrada

Na Figura 5, foi apresentada a estrutura básica de uma rede neural de função de base radial em que m_0 é a dimensão do vetor de entrada \mathbf{x} ; N é o tamanho da amostra de treinamento que corresponde à quantidade de neurônios na camada oculta. Cada neurônio é matematicamente descrito por uma função de base radial que é dada por

$$\varphi_j(\mathbf{x}) = \varphi(\|\mathbf{x} - \mathbf{x}_j\|), \quad j = 1, 2, \dots, N$$

Considerando as funções de base radial como sendo uma função Gaussiana, cada unidade computacional da camada oculta da rede é definida por

$$\begin{aligned} \varphi_j(\mathbf{x}) &= \varphi(\mathbf{x} - \mathbf{x}_j) \\ &= \exp\left(-\frac{1}{2\sigma_j^2} \|\mathbf{x} - \mathbf{x}_j\|^2\right), \quad j = 1, 2, \dots, N \end{aligned} \quad (3.1)$$

Para o vetor $\boldsymbol{\sigma} = [\sigma_1, \dots, \sigma_N]^T$, σ_j representa o comprimento da j -ésima Gaussiana com centro \mathbf{x}_j para o neurônio j . Os centros das funções Gaussianas são dadas por

$$\mathbf{x}_{ij} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1N} \\ x_{21} & x_{22} & \cdots & x_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m_0 1} & x_{m_0 2} & \cdots & x_{m_0 N} \end{bmatrix} \quad (3.2)$$

em que \mathbf{x}_{ij} representa o valor da coordenada do centro da função gaussiana do neurônio j para a i -ésima entrada, $i = 1, 2, \dots, m_0$.

A saída da rede é definida por

$$y = F(\mathbf{x}) = \sum_{j=1}^N w_j \varphi(\|\mathbf{x} - \mathbf{x}_j\|) \quad (3.3)$$

em que w_j é o i -ésimo elemento do vetor de pesos sinápticos $\mathbf{w} = [w_1, \dots, w_{m_0}]^T$; e, \mathbf{x}_j são os pontos de dados conhecidos considerados como centros das funções de base radial.

O índice de desempenho é definido como uma função do quadrado do erro entre a saída da RBF e a saída desejada e é expresso por

$$\mathcal{E}(n) = \frac{1}{2} (y(n) - d(n))^2 \quad (3.4)$$

Os parâmetros livres podem ser atualizados de acordo com o método do gradiente descendente. A atualização dos pesos conectados ao neurônio j é dada por

$$\Delta w_j(n) = -\eta \frac{\partial \mathcal{E}}{\partial w_j} = \eta (y(n) - d(n)) \mathbf{x}_j$$

De acordo com a regra de Widrow-Hoff e com a inserção do termo momento a atualização dos pesos resulta em

$$w_j(n) = w_j(n-1) + \Delta w_j(n) + \alpha(w_j(n-1) - w_j(n-2)) \quad (3.5)$$

A variação da distância entre as funções gaussianas e os seus respectivos centros é dada por

$$\Delta \sigma_j = -\eta \frac{\partial \mathcal{E}}{\partial \sigma_j} = \eta (y(n) - d(n)) w_j(n) \mathbf{x}_j \frac{\|\mathbf{x} - \mathbf{x}_j\|^2}{\sigma_j^3}$$

Assim como aconteceu com os pesos sinápticos, a atualização das distâncias com a inserção do termo momento assume a seguinte forma

$$\sigma_j(n) = \sigma_j(n-1) + \Delta \sigma_j(n) + \alpha(\sigma_j(n-1) - \sigma_j(n-2)) \quad (3.6)$$

Os centros das gaussianas são expressos por

$$\Delta \mathbf{x}_j = -\eta \frac{\partial \mathcal{E}}{\partial \mathbf{x}_j} = \eta (y(n) - d(n)) \mathbf{x}_j \frac{\mathbf{x} - \mathbf{x}_j^2}{\sigma_j^2} \quad (3.7)$$

Com a inserção do termo momento a atualização dos centros das gaussianas é expressa por

$$\mathbf{x}_j(n) = \mathbf{x}_j(n-1) + \Delta \mathbf{x}_j + \alpha(\mathbf{x}_j(n-1) - \mathbf{x}_j(n-2)) \quad (3.8)$$

Se os valores de \mathbf{x}_j e σ_j forem escolhidos de acordo com a faixa de valores de entrada pode-se fixar os valores de \mathbf{x}_j e σ_j e trabalhar somente com a Equação (3.5).

3.1.1 Comparação Entre Redes do Tipo RBF e MLP

Devido a semelhança entre as topologias das redes RBF e MLP, muitas vezes considera-se que as redes RBF fazem parte das redes MLP. Mas com exceção da topologia, as redes RBF e MLP possuem propriedades diferentes. Em primeiro lugar, as redes RBF são mais simples do que as redes MLP, pois as redes MLP normalmente têm arquiteturas mais

complexas. Em segundo lugar, as redes RBF são mais fáceis de serem treinadas do que as redes MLP por causa da sua arquitetura simples e fixa de três camadas. Em terceiro lugar, as redes RBF atuam como redes de aproximação local e as saídas de rede são determinadas por unidades ocultas especificadas em certos campos receptivos locais, enquanto as redes MLP trabalham globalmente e as saídas de rede são decididas por todos os neurônios (XIE; YU; WILAMOWSKI, 2011)(BISHOP, 2005). Em quarto lugar, é essencial definir estados iniciais corretos para redes RBF, enquanto as redes MLP usam inicialmente parâmetros gerados aleatoriamente. Por último e mais importante, os mecanismos de classificação para redes RBF e redes MLP são diferentes: os agrupamentos RBF são separados por hiper-esferas, enquanto que em redes MLP, hiper-superfícies arbitrariamente formatadas são usadas para a separação (XIE; YU; WILAMOWSKI, 2011).

As redes MLP e RBF têm basicamente as mesmas aplicações, mas estruturas de cálculo internas diferentes. A principal vantagem das redes RBF é a sua estrutura mais simples do que a MLP, uma vez que aquela tem apenas três camadas e o processo de treinamento é muito mais rápido, visto que no processo de treinamento da rede MLP existe uma etapa de propagação e outra de retropropagação do erro (HAYKIN et al., 2009)(BISHOP, 2005). Além disso, as redes RBF são capazes de boa generalização, alta tolerância aos ruídos de entrada e capacidade de aprendizagem *online* (SANTOS et al., 2013). Do ponto de vista da generalização, as RBF podem responder muito bem a padrões que não foram usados para treinamento (YU et al., 2011).

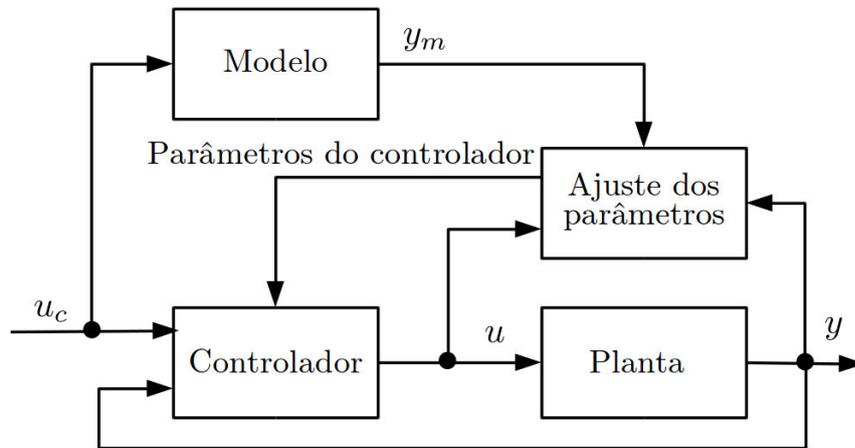
3.1.2 Sistema Adaptativo por Modelo de Referência

O sistema adaptativo por modelo de referência (MRAS, *model-reference adaptive system*) foi originalmente proposto para resolver o problema em que as especificações de desempenho são dadas em termos de um modelo de referência. Ele pode ser considerado como um servo-sistema adaptativo em que o desempenho desejado é expresso em termos de um modelo de referência, o qual fornece a resposta desejada a um determinado sinal de comando. Um diagrama de blocos desse sistema é mostrado na Figura 6. O sistema tem uma malha de realimentação comum composta do processo e do controlador e outra malha de realimentação que modifica os parâmetros do controlador. Os parâmetros mudam com base na realimentação do erro, que é a diferença entre a saída do sistema e a saída do modelo de referência. Os mecanismos de ajuste dos parâmetros no sistema adaptativo por modelo de referência podem ser obtidos de duas formas: usando o método do gradiente (regra MIT) ou aplicando a teoria de estabilidade.

3.1.2.1 A regra MIT

A regra MIT é a abordagem original do controle adaptativo por modelo de referência. Para apresentar a regra MIT deve-se considerar um sistema em malha fechada em que

Figura 6 – Diagrama de blocos de um sistema adaptativo por modelo de referência



Fonte: Adaptado de Åström e Wittenmark (2013)

o controlador tem um parâmetro ajustável, θ . A resposta desejada em malha fechada é especificada por um modelo cuja saída é y_m . Considerando e o erro entre a saída y e a saída do sistema em malha fechada do modelo y_m . Uma possibilidade é ajustar os parâmetros de tal forma que a função de custo

$$J(\theta) = \frac{1}{2}e^2 \quad (3.9)$$

é minimizada. Para minimizar J é necessário mudar os parâmetros na direção do gradiente negativo de J que é dado por

$$\frac{d\theta}{dt} = -\gamma \frac{\partial J}{\partial \theta} = -\gamma e \frac{\partial e}{\partial \theta} \quad (3.10)$$

Em que $e = y - y_m$ determina o erro do modelo e θ é um parâmetro do controlador. O parâmetro γ determina a taxa de adaptação (ÅSTRÖM; WITTENMARK, 2013)(LANDAU et al., 2017). A Equação (3.10) também se aplica quando há muitos parâmetros para serem ajustados. O símbolo θ pode ser interpretado como um vetor e $\partial e / \partial \theta$ como o gradiente do erro com relação aos parâmetros θ .

3.1.2.2 Teoria de Estabilidade

Por não existir garantia que um controlador adaptativo baseado na regra MIT resultará em um sistema estável em malha fechada, é desejável projetar um controlador adaptativo que garanta a estabilidade do sistema. Uma noção intuitiva de estabilidade é que sistemas estáveis são sistemas bem comportados em torno de um ponto de operação desejado. Conceitualmente existem duas classes de estabilidade: a estabilidade de Lyapunov e a estabilidade entrada-saída.

Para a estabilidade de Lyapunov, considere o sistema autônomo

$$\dot{x}(t) = g[x(t)] \quad (3.11)$$

em que $x \in \mathfrak{R}^n$ e $g(x)$ é uma função vetorial $n \times 1$. Assumindo que $g(0) = 0$, assim $x = 0$ é um ponto de equilíbrio do sistema (WANG, 1999).

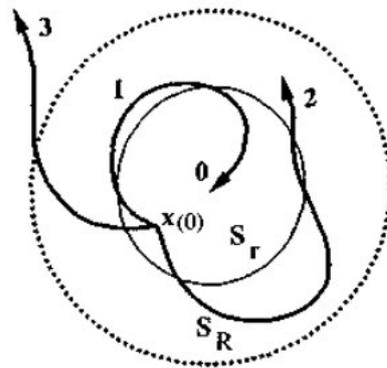
Definição 3.1: (WANG, 1999) (Estabilidade de Lyapunov) O ponto de equilíbrio $x = 0$ é dito estável se, para qualquer $\epsilon > 0$, então existe um $\delta > 0$ tal que $\|x(0)\| < \delta$ implica que $\|x(t)\| < \epsilon$ para todo $t \geq 0$. Um sistema é dito assintoticamente estável se ele é estável e adicionalmente existe $\delta' > 0$ tal que $\|x(0)\| < \delta'$ implica $x(t) \rightarrow 0$ quando $t \rightarrow \infty$. Finalmente, o ponto de equilíbrio $x = 0$ é dito estável se existe números positivos α , λ e r tal que

$$\|x(t)\| < \alpha \|x(0)\| e^{-\lambda t} \quad (3.12)$$

para todo $t \geq 0$ e $\|x(0)\| \leq r$. Se a estabilidade assintótica ou exponencial se mantiver para qualquer estado inicial $x(0)$, então o ponto de equilíbrio $x = 0$ é dito ser globalmente assintótico ou exponencialmente estável.

Na Figura 7, tem-se a ilustração dos sistemas assintoticamente estável (curva 1); sistema marginalmente estável (curva 2) e sistema instável (curva 3).

Figura 7 – Conceitos de estabilidade



Fonte: Slotine, Li et al. (1991)

Na curva 1, da Figura 7, verifica-se que existe uma superfície de raio $r < R$ tal que se $\|x(0)\| < r$, então $\|x(t)\| < R$ para todo $t \geq 0$. Isto significa que a trajetória do estado do sistema pode ser mantida arbitrariamente próxima da origem iniciando suficientemente perto dela. Ainda da Figura 7, percebe-se que quando a trajetória do estado ultrapassa a superfície de raio R o sistema é dito instável. Porém, quando a trajetória dos estados permanece dentro da região $r < \|x(t)\| < R$ o sistema é dito marginalmente estável. A superfície de raio r é chamada de região de atração.

Para a estabilidade entrada-saída, considera-se qualquer sistema que mapeia a entrada $u(t) \in \mathfrak{R}^r$ à saída $y(t) \in \mathfrak{R}^m$.

Definição 3.2: (WANG, 1999) (Estabilidade entrada-saída) Fazendo L_p^n ser o conjunto de todas as funções vetoriais $g(t) = (g_1(t), \dots, g_n(t))^T : [0, \infty) \rightarrow \mathfrak{R}^n$, tal que $\|g\|_p = (\sum_{i=1}^n \|g_i\|_p^2)^{1/2} < \infty$, em que $\|g_i\|_p = (\int_0^\infty |g_i(t)|^p dt)^{1/p}$, $p \in [1, \infty]$ e $\|g_i\|_\infty = \sup_{t \in [0, \infty)} |g_i(t)|$. Um sistema com entrada $u(t) \in \mathfrak{R}^r$ e saída $y(t) \in \mathfrak{R}^m$ é dito L_p estável se

$$u(t) \in L_p^r \text{ implica } y(t) \in L_p^m \quad (3.13)$$

em que $p \in [0, \infty]$. Em particular um sistema é L_∞ estável (ou estabilidade entrada limitada saída limitada) se $u(t) \in L_\infty^r$ implicar que $y(t) \in L_\infty^m$.

3.2 Controle de Modos Deslizantes

Frequentemente, há divergências entre a planta real e o modelo matemático desenvolvido para o projeto do controlador. Estas diferenças podem ser devido a vários fatores, pois o modelo é uma simplificação do mundo real ou de alguma forma conveniente de trabalhar com este mundo de modo que o seu comportamento seja muito próximo do sistema modelado. Vários métodos de controle foram desenvolvidos para eliminar qualquer discordância entre o modelo e a planta real.

Uma dessas abordagens para o projeto do controlador é a metodologia chamada de controle de modo deslizante (SMC, *sliding mode control*). Este é um tipo específico de sistema de controle de estrutura variável. O *sliding mode control* tem sido extensivamente implementado em diferentes tipos de sistemas, incluindo sistemas mecânicos, sistemas de potência, etc.

Considere um sistema SISO não linear dado por

$$x^{(n)} = f(\mathbf{x}) + u \quad (3.14)$$

em que $u \in \mathfrak{R}$ é a entrada de controle, $x \in \mathfrak{R}$ é a saída, e $\mathbf{x} = (x, \dot{x}, \dots, x^{(n-1)})^T \in \mathfrak{R}$ é o vetor de estado. Na Equação 3.14, a função $f(\mathbf{x})$ não é totalmente conhecida, mas a incerteza de $f(\mathbf{x})$ é limitada por uma função conhecida de \mathbf{x} . Entretanto, a função $f(\mathbf{x})$ é expressa por

$$f(\mathbf{x}) = \hat{f}(\mathbf{x}) + \Delta f(\mathbf{x}) \quad (3.15)$$

e

$$|\Delta f(\mathbf{x})| \leq F(\mathbf{x}) \quad (3.16)$$

em que $\Delta f(\mathbf{x})$ conhecido, mas $\hat{f}(\mathbf{x})$ e $F(\mathbf{x})$ são desconhecidas. O objetivo é determinar uma lei de controle $u = u(\mathbf{x})$ tal que o estado \mathbf{x} do sistema em malha fechada siga o

estado desejado $\mathbf{x}_d = (x_d, \dot{x}_d, \dots, x_d^{(n-1)})^T$ (WANG, 1999). Ou seja, o erro de rastreamento, definido por

$$\mathbf{e} = \mathbf{x} - \mathbf{x}_d = (e, \dot{e}, \dots, e^{(n-1)})^T \quad (3.17)$$

deve convergir para zero.

A ideia básica do controle deslizante é dada a seguir. Definida uma função escalar

$$\begin{aligned} s(\mathbf{x}, t) &= \left(\frac{d}{dt} + \lambda\right)^{n-1} e \\ &= e^{(n-1)} + C_{n-1}^1 \lambda e^{(n-2)} + C_{n-1}^2 \lambda^2 e^{(n-3)} + \dots + \lambda^{n-1} e \end{aligned} \quad (3.18)$$

em que λ é uma constante positiva (SLOTINE; LI et al., 1991). Então,

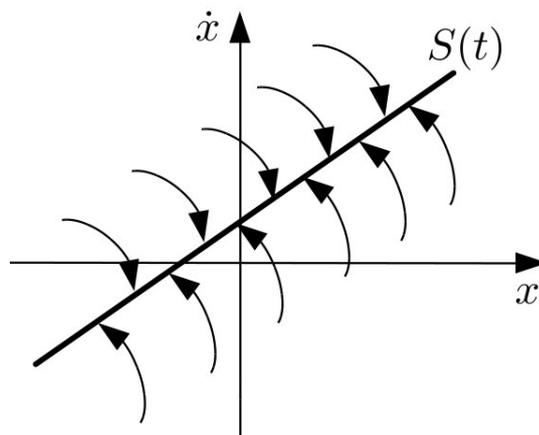
$$s(\mathbf{x}, t) = 0$$

define a superfície variante no tempo $S(t)$ no espaço de estados \mathfrak{R}^n . Por exemplo, se $n = 2$ então a superfície $S(t)$ é dada por

$$s(\mathbf{x}, t) = \dot{e} + \lambda e = \dot{x} + \lambda x - \dot{x}_d - \lambda x_d = 0 \quad (3.19)$$

que é uma linha reta no plano de fase $x - \dot{x}$. Essa superfície de deslizamento é mostrada na Figura 8.

Figura 8 – Superfície de deslizamento



Fonte: Produzido pelo autor

Assim, o problema de controle de rastreamento é equivalente a manter a função escalar $s(\mathbf{x}, t)$ em zero. Isso pode ser feito escolhendo a lei de controle u tal que

$$\frac{1}{2} \frac{d}{dt} s^2 \leq -\eta |s| \quad (3.20)$$

em que η é uma constante positiva. A Equação (3.20), é chamada de condição de deslizamento. Ela garante que $|s(\mathbf{x}, t)|$ diminuirá se \mathbf{x} não está na superfície $S(t)$, ou seja, a trajetória do estado se moverá para a superfície $S(t)$, como ilustrado na Figura 8. A superfície $S(t)$ é chamada de superfície de deslizamento, o sistema na superfície está no modo deslizante e o controle que garante a Equação (3.20), é chamado de controle de modo deslizante ou controle deslizante.

3.3 Comentários

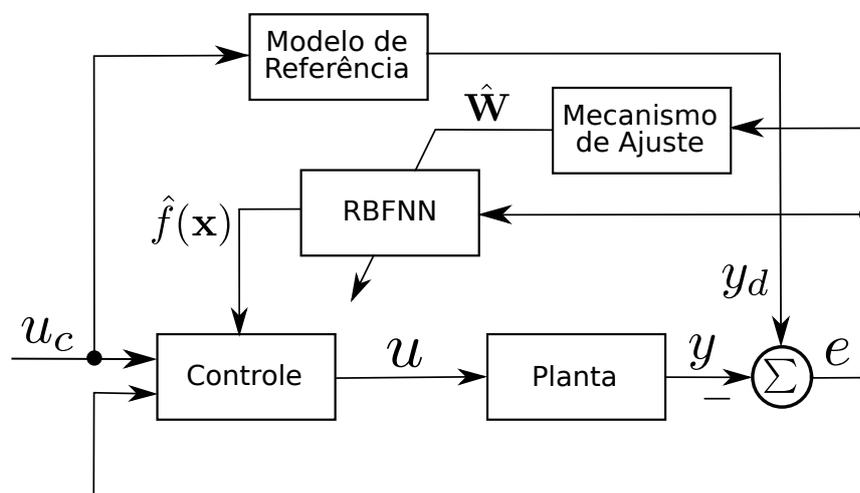
Neste capítulo foi apresentada a arquitetura de uma rede neural artificial do tipo RBF. Em seguida foi feita uma comparação entre ambas as arquiteturas. Para este trabalho foi escolhida a rede do tipo RBF pelo fato de ter uma estrutura mais simples que a MLP e o processo de treinamento ser mais rápido. Posteriormente foi mostrado o sistema adaptativo por modelo de referência com a regra MIT. A regra MIT foi o gênesis do controle adaptativo por modelo de referência, porém a mesma não tem garantia de estabilidade do sistema em malha fechada. Dessa forma, apresentou-se a teoria de estabilidade com foco na estabilidade de Lyapunov. O capítulo é finalizado com o controle por modos deslizantes, que é um tipo de controle com estrutura variável em que as trajetórias dos estados se movem para a superfície de deslizamento.

Modelos Neuro-Adaptativos para Projeto *Online* de SC

Na abordagem apresentada neste trabalho, o sistema é formado por modelo de referência, mecanismo de ajuste, rede neural do tipo RBF e controle por modos deslizantes. Em suma, os blocos que contém a rede neural do tipo RBF e o mecanismo de ajuste são os principais blocos do sistema de controle *online* com propriedades de adaptabilidade e aprendizado, conforme apresentados na Figura 9. Ainda, apresenta-se o desenvolvimento e as características dos modelos baseados em estruturas algébricas e em RNA, abordando-se o desenvolvimento e as características dos modelos e estruturas referidos anteriormente.

Estes modelos estão inseridos no contexto do diagrama de blocos do sistema apresentado na Figura 9, que representa a arquitetura de controle de alto desempenho, no contexto adaptativo, baseado no modelo de referência a ser seguido pela planta, mecanismo de ajuste, rede neural do tipo RBF e estratégia de controle *slide mode control* (SMC). Estes modelos representam os conteúdos que são inseridos nos blocos funcionais do projeto

Figura 9 – Controle adaptativo neural



Fonte: Produzido pelo autor

online de sistema de controle neuro-adaptativo.

Nas seções do restante deste capítulo, apresentam-se os modelos para cada bloco funcional do sistema de controle neuro-adaptativo da Figura 9. A teoria, algoritmos e métodos para a sintonia de parâmetros de sistemas de controle adaptativo constituem a sistematização para a integração dos modelos que constituem as etapas que o projeto *online* de sistema de controle neuro-adaptativos são apresentados de forma integrada e referenciado de acordo com os conceitos e formulações das abordagens de controle adaptativo e redes neurais apresentadas nos Capítulos 2 e 3.

4.1 Modelos Neuro-Adaptativos

Os modelos neuro-adaptativos são constituídos dos modelos que representam as unidades do controlador, sintonia e mecanismo de ajuste das unidades do modelo geral para o projeto *online* de sistemas de controle de alto desempenho.

4.1.1 Controle Adaptativo Baseado em RBF

Considere um sistema de segunda ordem dado da seguinte forma

$$\ddot{y} = f(y, \dot{y}) + g(y, \dot{y})u \quad (4.1)$$

em que f e g são funções desconhecidas, e u e y são a entrada e a saída do sistema, respectivamente.

A Equação (4.1) pode ser escrita como

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(x_1, x_2) + g(x_1, x_2)u \\ y &= x_1 \end{aligned} \quad (4.2)$$

O sinal de posição ideal é y_d e o erro é definido por

$$e = y_d - y = y_d - x_1 \quad (4.3)$$

A lei de controle por modos deslizantes segundo Slotine, Li et al. (1991), é dada por

$$u^* = \frac{1}{g(\mathbf{x})} [-f(\mathbf{x}) + \ddot{y}_d + \beta^T \mathbf{E}] \quad (4.4)$$

em que $\mathbf{x} = [x_1 \ x_2]^T$, $\beta = [\beta_0 \ \beta_1]^T$ e $\mathbf{E} = [e \ \dot{e}]^T$. Substituindo a lei de controle (4.4) e o erro (4.3) em (4.1), obtém-se o seguinte sistema de controle em malha fechada em função do erro

$$\ddot{e} + \beta_1 \dot{e} + \beta_0 e = 0 \quad (4.5)$$

com β_1 e β_0 constantes positivas escolhidas, tais que o polinômio $s^2 + \beta_1 s + \beta_0$ seja estável (WANG, 1999). Com as escolhas corretas de β_1 e β_0 tem-se que quando $t \rightarrow \infty$, $e \rightarrow 0$ e $\dot{e} \rightarrow 0$.

Da Equação (4.4), se $f(\mathbf{x})$ e $g(\mathbf{x})$ não forem conhecidas a lei de controle não é realizável.

Levando em consideração que da Equação (4.4) somente $f(\mathbf{x})$ é desconhecida pode-se projetar uma RBF para aproximar $\hat{f}(\mathbf{x})$ de $f(\mathbf{x})$. O algoritmo da RBF é descrito como (GE et al., 2013)

$$\begin{aligned} h_j &= g(\|\mathbf{E} - c_{ij}\|/b_j^2) \\ f &= \mathbf{W}^T \mathbf{h}(\mathbf{E}) + \varepsilon \end{aligned}$$

em que $g(\cdot)$ é a função gaussiana; c e b são o centro e a largura da gaussiana, respectivamente; \mathbf{E} é o vetor de entradas, i é a i -ésima entrada da rede, j é o j -ésimo neurônio da camada oculta, $\mathbf{h} = [h_1, h_2, \dots, h_n]^T$ é a saída da camada oculta, \mathbf{W} é o vetor de pesos e ε é o erro da aproximação.

A saída da rede RBF que fornece a aproximação de f é dada por

$$\hat{f}(\mathbf{x}) = \hat{\mathbf{W}}^T \mathbf{h}(\mathbf{E}) \quad (4.6)$$

em que $\hat{\mathbf{W}}$ é a estimativa de \mathbf{W} .

Dessa forma, a lei de controle resultante é dado por

$$u = \frac{1}{g(\mathbf{x})} [-\hat{f}(\mathbf{x}) + \ddot{y}_d + \beta^T \mathbf{E}] \quad (4.7)$$

A lei de controle adaptativa escolhida é representada a seguir

$$\dot{\hat{\mathbf{W}}} = -\gamma \mathbf{E}^T \mathbf{P} \mathbf{B} \mathbf{h}(\mathbf{x}) \quad (4.8)$$

4.1.2 Análise da Estabilidade

Substituindo a lei de controle, expressa pela Equação (4.7), na Equação (4.1), tem-se que o sistema em malha fechada é expresso como (WANG, 1999)

$$\ddot{e} = -\beta^T \mathbf{E} + [\hat{f}(\mathbf{x}) - f(\mathbf{x})] \quad (4.9)$$

Fazendo

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\beta_0 & -\beta_1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (4.10)$$

a Equação (4.9) pode ser escrita da seguinte forma

$$\dot{\mathbf{E}} = \mathbf{A} \mathbf{E} + \mathbf{B} [\hat{f}(\mathbf{x}) - f(\mathbf{x})] \quad (4.11)$$

O vetor de peso ótimo, no sentido de aproximar $\hat{f}(\mathbf{x})$ de $f(\mathbf{x})$, é expresso como

$$\mathbf{W}^* = \arg \min_{\mathbf{W} \in \Omega} \left\{ \sup |\hat{f}(\mathbf{x}) - f(\mathbf{x})| \right\} \quad (4.12)$$

O erro de modelagem é caracterizado por

$$\omega = \hat{f}(\mathbf{x}|\mathbf{W}^*) - f(\mathbf{x}) \quad (4.13)$$

Consequentemente, a Equação (4.11) resulta em

$$\dot{\mathbf{E}} = \mathbf{A}\mathbf{E} + \mathbf{B}\{[\hat{f}(\mathbf{x}) - \hat{f}(\mathbf{x}|\mathbf{W}^*)] + \omega\} \quad (4.14)$$

Escolhendo a função de Lyapunov como

$$V = \frac{1}{2}\mathbf{E}^T\mathbf{P}\mathbf{E} + \frac{1}{2\gamma}(\hat{\mathbf{W}} - \mathbf{W}^*)^T(\hat{\mathbf{W}} - \mathbf{W}^*) \quad (4.15)$$

em que γ é uma constante positiva. O erro de estimação do vetor de pesos é representado por $\hat{\mathbf{W}} - \mathbf{W}^*$. A matriz \mathbf{P} é simétrica e definida positiva e satisfaz a seguinte equação de Lyapunov

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} = -\mathbf{Q} \quad (4.16)$$

com $\mathbf{Q} \geq 0$, \mathbf{A} é dada pela Equação (4.10). A derivada de V resulta

$$\begin{aligned} \dot{V} = & -\frac{1}{2}\mathbf{E}^T\mathbf{Q}\mathbf{E} + \mathbf{E}^T\mathbf{P}\mathbf{B}\omega + \\ & + \frac{1}{\gamma}(\hat{\mathbf{W}} - \mathbf{W}^*)^T \left[\dot{\hat{\mathbf{W}}} + \gamma\mathbf{E}^T\mathbf{P}\mathbf{B}h(\mathbf{x}) \right] \end{aligned} \quad (4.17)$$

Substituindo a lei adaptativa (4.8) na Equação (4.17), tem-se que

$$\dot{V} = -\frac{1}{2}\mathbf{E}^T\mathbf{Q}\mathbf{E} + \mathbf{E}^T\mathbf{P}\mathbf{B}\omega \quad (4.18)$$

Desde que o erro de aproximação, ω , pode ser feito muito pequeno por meio da RBF e $-\frac{1}{2}\mathbf{E}^T\mathbf{Q}\mathbf{E} \leq 0$, pode-se obter $\dot{V} \leq 0$.

4.2 Algoritmos *Online* do SC Proposto

Os algoritmos *online* do sistema de controle adaptativo tem por base as arquiteturas de rede neurais do tipo RBF, e o controle adaptativo do tipo *model reference adaptive control*. Estas arquiteturas são integradas ao método ou estratégia de controle do tipo modo deslizante para determinação dos parâmetros do sistema de controle com características de adaptabilidade e aprendizado.

Os algoritmos de estimação e controle estão descritos nas Subseções 4.2.1 e 4.2.2.

4.2.1 Algoritmos do Estimador

O Algoritmo 4.1, representa a implementação das Equações (2.1-2.3) do *recursive least square*, conforme apresentado na Subseção 2.1.2.2 do Capítulo 2.

Algoritmo 4.1: Algoritmo do Estimador RLS

Entrada: \mathbf{P} , y , u , N

Resultado: θ

```

1 inicialização;
2 para  $i = 1$  até  $N$  faça
    /* cálculo do vetor de regressores */
3    $\varphi = [-y(i-1); -y(i-2); u(i-1)u(i-2)];$ 
4    $K = (\mathbf{P} \times \varphi) / (1 + (\varphi^T \times \mathbf{P} \times \varphi))$ ; /* cálculo do ganho */
    /* cálculo da matriz de covariância */
5    $\mathbf{P} = (\mathbf{P} - K \times \varphi^T \times \mathbf{P})$ ;
    /* cálculo do vetor de parâmetros */
6    $\theta(1:N, i) = \theta(1:N, i-1) + K \times (y(i) - \varphi^T \times \theta(1:N, i-1))$ ;
7 fim
```

O estimador do sistema tem por base o treinamento RLS padrão do Algoritmo 4.1. O Algoritmo 4.1, tem como entrada os vetores de saída, y , e de entrada, u , que são amostrados em cada instante T , a matriz \mathbf{P} é inicializada com valores altos.

Regido pelas Equações (2.1-2.3) do RLS, o restante das instruções do algoritmo consiste dos passos básicos que são o cálculo do vetor de regressores na instrução 3, o cálculo do ganho na instrução 4, o cálculo da matriz de covariância na instrução 5 e o cálculo do vetor de parâmetros na instrução 6.

4.2.2 Algoritmo de Controle

O algoritmo de controle ajusta o ganho do controlador em função da saída da rede do tipo RBF e a sintonia dos parâmetros livres da rede neural é realizada conforme descrito na Subseção 4.1.1.

O algoritmo de controle representa o processo de rastreamento do alvo móvel pelo sistema proposto. O acompanhamento do objeto em questão é realizado pela unidade de atuação que posiciona o sistema-rastreador de forma a acompanhar o alvo observado. No Algoritmo 4.2, existe a presença da saturação do sinal de controle. Essa saturação será

abordada na Seção 5.3.

Algoritmo 4.2: Algoritmo de controle

Entrada: \mathbf{B} , \mathbf{C} , b , y_d , y_{d_1} , y_1 , γ , \mathbf{P} , \mathbf{W}_1 , gx , K , N

Resultado: $u(x)$

```

1  inicialização;
2  M=# de colunas de  $\mathbf{C}$ ;
3  para  $k = 1$  até  $N$  faça
4       $time(k) = k \times t_s$ ;                               /*  $t_s$  é o tempo de amostragem */
5       $dy_d = (y_d(k) - y_{d_1})/t_s$ ;
6       $ddy_d = (dy_d(k) - dy_{d_1})/t_s$ ;
7      ler( $y(k)$ );                                           /* saída do sensor de posição */
8       $dy(k) = (y(k) - y_1)/t_s$ ;
9       $e(k) = (y_d(k) - y(k))$ ;                               /* cálculo do erro */
10      $de(k) = (dy_d(k) - dy(k))$ ;
11      $\mathbf{E} = [e(k), de(k)]^T$ ;                               /* vetor de erro */
12     para  $j = 1$  até  $M$  faça
13          $h(j) = exp(-(\mathbf{E} - \mathbf{C}(1 : M, j))^T \times (\mathbf{E} - \mathbf{C}(1 : M, j)) / (2 \times b \times b))$ 
14     fim
15      $\mathbf{W} = \mathbf{W}_1 - t_s \times \gamma \times \mathbf{E}^T \times \mathbf{P} \times \mathbf{B} \times h$ ; /* cálculo do vetor de pesos */
16      $\hat{f}(k) = \mathbf{W}^T \times h$ ;                               /* função  $\hat{f}(x)$  */
17      $u(k) = (-\hat{f}(k) + ddy_d(k) + \mathbf{K}^T \times \mathbf{E})/gx$ ;     /* cálculo de  $u(x)$  */
18     /* saturação de  $u(x)$  */                               */
19     se  $u(k) < -12$  então
20          $u(k) = -12$ ;
21     senão se  $u(k) > 12$  então
22          $u(k) = 12$ ;
23     fim
24     escreve( $u(k)$ ); /* sinal de controle aplicado ao atuador */
25     /* retorno dos parâmetros */                           */
26      $\mathbf{W}_1 = \mathbf{W}$ ;
27      $y_{d_1} = y_d(k)$ ;
28      $dy_{d_1} = dy_d(k)$ ;
29 fim

```

Seguidamente, apresenta-se a descrição de operação do algoritmo e a origem das equações que fazem parte das instruções. Os dados de entrada do Algoritmo 4.2, são: \mathbf{B} , \mathbf{C} , b , y_d , y_{d_1} , y_1 , γ , \mathbf{P} , \mathbf{W}_1 , gx , K , N . Na instrução 4, t_s é o tempo de amostragem para aquisição do sinal em intervalos de tempo T . Nas linhas 5 e 6 são feitas diferenciações numéricas de primeira e segunda ordem, respectivamente, do sinal desejado y_d . Estes cálculos servirão para compor a lei de controle da Equação (4.7). Na instrução 7, é

realizada a leitura da tensão na saída do potenciômetro para fins de comparação entre o sinal de saída do sistema e o sinal desejado. Os cálculos feitos nas linhas 8-10 servem para computar os elementos que constituem o vetor de erro \mathbf{E} . No laço composto pelas linhas 12-14, é realizado o cálculo das camadas ocultas da rede RBF. Na linha 15, é obtido o valor do vetor de pesos \mathbf{W} , conforme Equação (4.8). Nas linhas 16 e 17, são calculadas as Equações (4.6) e (4.7), respectivamente. A função de saturação do sinal de controle está representada entre as linhas 18-22. Na linha 23, o valor obtido de $u(k)$ é aplicado ao atuador do sistema. Nas linhas 24-26, é realizado o retorno dos parâmetros do algoritmo.

4.3 Comentários

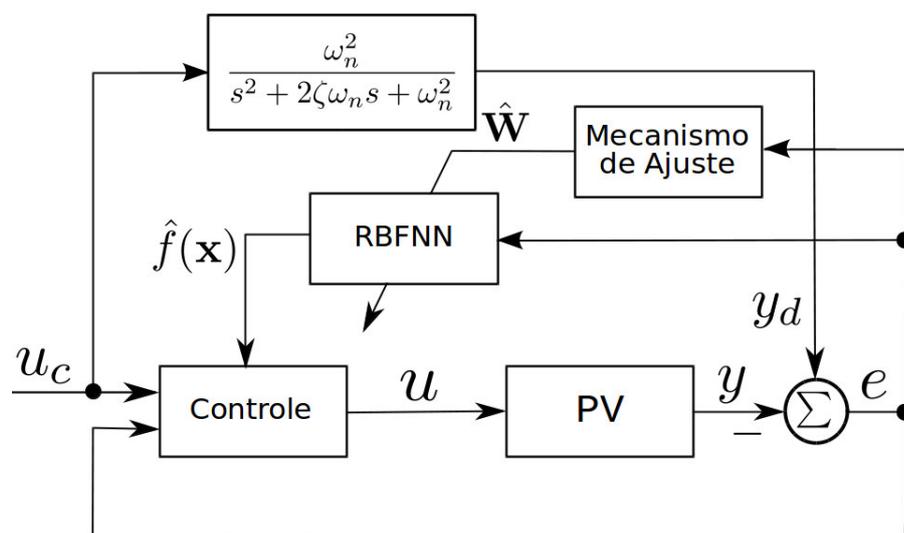
Neste capítulo, foi exibido o controle adaptativo baseado em rede neural artificial do tipo RBF, bem como a análise da estabilidade acerca da lei de controle adaptativa escolhida. A lei de controle escolhida leva em consideração a filosofia do controle por modos deslizantes. Na sequência, é apresentado o algoritmo do estimador dos parâmetros da planta e o algoritmo de controle proposto para realizar o ajuste da sinal de controle de forma que a variável controlada tenha a mesma dinâmica que o modelo de referência pré-determinado. O capítulo foi finalizado com os algoritmos propostos do estimador e de controle que são baseados no treinamento RLS e RBF, respectivamente.

Projeto *Online* do SC-Neuro-Adaptativo

O desenvolvimento dos métodos e algoritmos propostos são apresentados neste capítulo. A base teórica para este desenvolvimento foram expostos no Capítulo 4. Além disso, é apresentada a descrição do problema juntamente com a sua formulação matemática e o seu modelo discretizado. Para a realização do controle do seguidor solar é apresentado o projeto e a implementação do SC-neuro-adaptativo. Posteriormente, são apresentados os testes de validação computacional e em *hardware*.

Na Figura 10, apresenta-se a arquitetura do sistema de controle neuro-adaptativo. Esta estrutura viabiliza a implementação de uma plataforma para o projeto *online* de controladores e estimadores do referido sistema de controle. Por sua vez, o SC-neuro-adaptativo tem por base o controle adaptativo, controle de modos deslizantes e redes neurais artificiais.

Figura 10 – Controle neuro-adaptativo aplicado ao PV



Fonte: Produzido pelo autor

5.1 Problema do Seguidor Solar

A metodologia apresentada é aplicada em um sistema de acompanhamento solar com o propósito de manter a superfície do módulo fotovoltaico (PV) perpendicular a incidência dos raios solares. O sistema dinâmico, representado pelo atuador, PV e sensor de posicionamento pode ser visto na Figura 11. O atuador do sistema é um motor de corrente contínua com a sua armadura constituída de ímã permanente. O elemento responsável por medir a posição angular do PV é um potenciômetro acoplado ao eixo do PV.

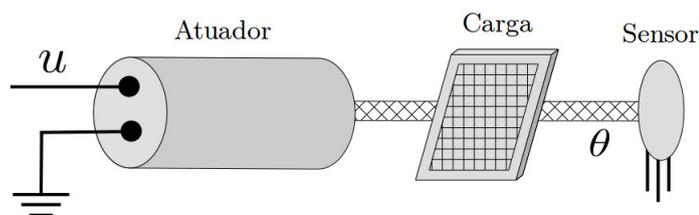
Figura 11 – Visão geral do sistema



Fonte: Produzido pelo autor

Para fins de controle, o PV é modelado como uma carga no eixo do motor. Com base nessa premissa, na Figura 12, mostra-se o acoplamento entre a carga e o atuador.

Figura 12 – PV como carga no eixo do motor



Fonte: Produzido pelo autor

Ainda que na Figura 12, o PV apareça conectado diretamente no eixo do atuador é

relevante salientar que o eixo do atuador, para a referida aplicação, é um parafuso sem fim acoplado em uma engrenagem que, por sua vez, ao seu centro está conectado o eixo que transmite o movimento ao PV. O acoplamento entre o parafuso sem fim e a engrenagem não permite que exista o movimento no sentido do painel para o eixo do motor. Portanto, nesse tipo de montagem, o movimento só acontece no sentido do eixo do motor para o painel fotovoltaico.

A modelagem matemática do conjunto atuador-carga é apresentada no Apêndice B.

5.2 Formulação Matemática do Sistema de Controle

A formulação matemática do problema é apresentada para o conjunto atuador e carga. No Apêndice B, apresenta-se o atuador eletromecânico utilizado e os detalhes das equações diferenciais ordinárias que regem a dinâmica do mesmo.

Se a corrente de campo, i_f , é constante, o torque aplicado na carga é dado por

$$J \frac{d^2\theta(t)}{dt^2} + f \frac{d\theta(t)}{dt} = k_t i_a(t) \quad (5.1)$$

em que $k_t i_a(t)$ o conjugado $T(t)$ e $k_t = k i_f(t)$ uma constante. O coeficiente J é considerado como o momento de inércia total da carga, eixo e rotor do motor; θ , o deslocamento angular da carga; e f , o coeficiente de fricção viscoso dos mancais de deslizamento.

Quando o atuador está movimentando a carga, uma tensão, v_b , força contra eletromotriz (fem), surge no circuito de armadura que, por sua vez, opõe-se à tensão aplicada. A tensão $v_b(t)$ é linearmente proporcional à velocidade angular do eixo do atuador é dada por

$$v_b(t) = k_b \frac{d\theta(t)}{dt} \quad (5.2)$$

Assim o circuito de armadura que pode ser visualizado na Figura 38, do Apêndice B, é descrito por

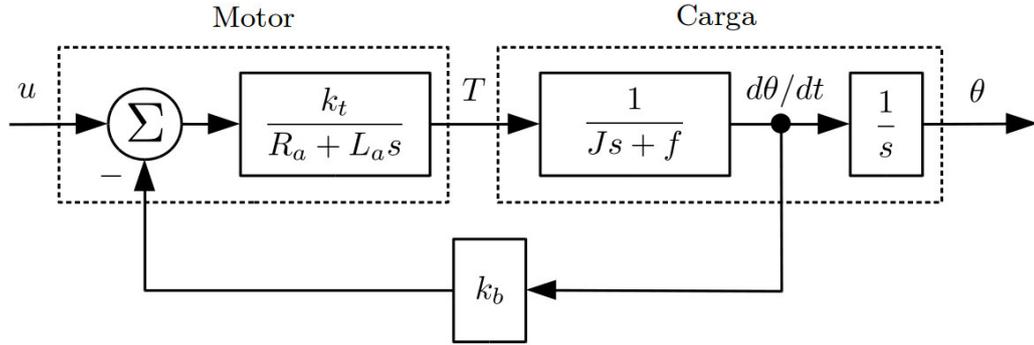
$$R_a i_a(t) + L_a \frac{di_a(t)}{dt} + v_b(t) = v_a(t) \quad (5.3)$$

Aplicando a transformada de Laplace na Equação (5.3) e por meio da eliminação de I_a nessas equações, obtém-se a função de transferência de $v_a = u$ para θ que é dada por

$$\bar{G}(s) = \frac{\Theta(s)}{U(s)} = \frac{k_t}{s[(Js + f)(R_a + L_a s) + k_t k_b]} \quad (5.4)$$

O diagrama de blocos da Figura 13, representa o modelo, em função de transferência, do motor CC controlado pela armadura. Este diagrama é elaborado com base nas Equações (5.1 – 5.3).

Figura 13 – Diagrama de blocos atuador-carga



Fonte: Produzido pelo autor

Em aplicações, a indutância de armadura, L_a , é levada a zero (CHEN, 2015). Neste caso, a Equação (5.4) é reduzida a

$$\bar{G}(s) = \frac{\Theta(s)}{U(s)} = \frac{k_t}{s(JR_a s + k_t k_b + fR_a)} = \frac{k_m}{s(\tau_m s + 1)} \quad (5.5)$$

em que

$$k_m = \frac{k_t}{k_t k_b + fR_a} \quad : \quad \text{Constante de ganho} \quad (5.6)$$

e

$$\tau_m = \frac{JR_a}{k_t k_b + fR_a} \quad : \quad \text{Constante de tempo} \quad (5.7)$$

A constante de tempo, τ_m , na Equação (5.7), depende da carga e do circuito de armadura. As Equações (5.4) e (5.5) são frequentemente chamadas de função de transferência do motor, embora elas sejam efetivamente a função de transferência do motor e carga.

5.2.1 Modelo Discretizado

Em seguida, deve-se encontrar o equivalente digital do conjunto eixo-PV, pois os compensadores digitais geram uma sequência de números que deve, posteriormente, ser transformada em um sinal analógico para ser aplicado a planta analógica. Essa conversão é chamada de segurador de ordem zero (ZOH).

A função de transferência do segurador de ordem zero é dada por

$$G_h(s) = \frac{1 - e^{-Ts}}{s} \quad (5.8)$$

Em controle digital, uma planta é geralmente conectada a um segurador de ordem zero (conversão D/A).

A função de transferência do modelo discretizado, considerando a dinâmica do segurador de ordem zero, é dada por

$$G(z) = \frac{z-1}{z} \mathcal{Z} \left[\frac{\bar{G}(s)}{s} \right] \quad (5.9)$$

Multiplicando o numerador e o denominador da Equação (5.5), por $1/\tau_m$, tem-se que

$$\bar{G}(s) = k_m \frac{1/\tau_m}{s(s+1/\tau_m)} \quad (5.10)$$

e

$$\frac{\bar{G}(s)}{s} = k_m \frac{1/\tau_m}{s^2(s+1/\tau_m)} \quad (5.11)$$

A função de transferência discreta, correspondente a Equação (5.11), é dada por

$$\mathcal{Z} \left[\frac{\bar{G}(s)}{s} \right] = \frac{k_m}{\tau_m} \frac{z[(\frac{T}{\tau_m} - 1 + e^{-\frac{T}{\tau_m}})z + (1 - e^{-\frac{T}{\tau_m}} - \frac{T}{\tau_m} e^{-\frac{T}{\tau_m}})]}{(z-1)^2(z - e^{-T/\tau_m})} \quad (5.12)$$

e o equivalente digital do conjunto eixo-PV resulta em

$$G(z) = \frac{k_m}{\tau_m} \frac{(\frac{T}{\tau_m} - 1 + e^{-\frac{T}{\tau_m}})z + (1 - e^{-\frac{T}{\tau_m}} - \frac{T}{\tau_m} e^{-\frac{T}{\tau_m}})}{(z-1)(z - e^{-T/\tau_m})} \quad (5.13)$$

Como o sistema é projetado para ficar ao ar livre, ele estará sujeito às mudanças das condições ambientais tais como variação de temperatura e de direção e intensidade do vento. Além disso, a função de transferência do mesmo motor será diferente se ele aciona várias cargas distintas, pois as características dinâmicas do atuador dependem da carga. Caso a carga não esteja balanceada o momento de inércia, J , é alterado em função do deslocamento angular, θ . Essas situações levam a uma variação dos parâmetros do modelo em função do tempo. Outra razão para o uso desta metodologia adaptativa são as incertezas paramétricas. Portanto, a fim de alcançar e manter um nível aceitável de desempenho do sistema de controle em malha fechada na ocorrência de variação dos parâmetros do modelo, uma abordagem de controle adaptativo deve ser considerada (LANDAU et al., 2011).

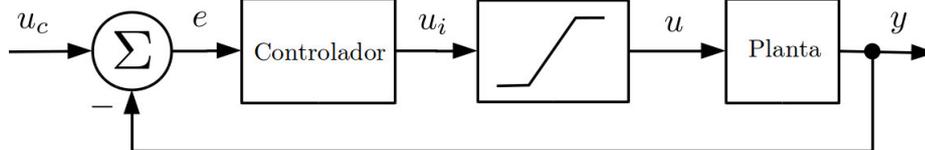
5.3 Projeto do SC-Neuro-Adaptativo

Os modelos para o projeto *online* de sistemas de controle neuro-adaptativo são aplicados, por meio de algoritmos de estimação e controle, em um sistema dinâmico constituído de um painel solar.

5.3.1 Modelagem da Não-Linearidade do Sistema

O sistema em malha fechada é mostrado na Figura 14. Entre a saída do controlador, u_i , e o sinal que efetivamente é aplicado nos terminais do atuador, u , existe a presença de uma função de saturação. A saída da função de saturação é o próprio sinal de controle

Figura 14 – Sistema em malha fechada



Fonte: Produzido pelo autor

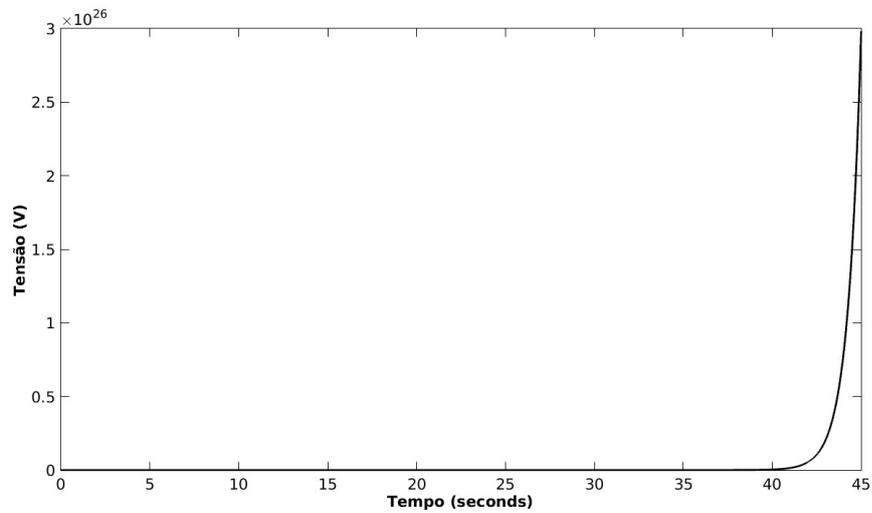
quando o mesmo está entre -12 V e 12 V. A função de saturação representa os limites de tensão impostos ao atuador, pois a fonte que fornece tensão ao atuador é limitada entre 0 V e 12 V. A comutação entre uma tensão positiva e negativa é realizada por meio de um circuito de ponte H. Matematicamente, a função de saturação é expressa por

$$u = \begin{cases} 12, & \text{se } u_i > 12 \\ u, & \text{se } -12 \leq u_i \leq 12 \\ -12, & \text{se } u_i < -12 \end{cases} \quad (5.14)$$

5.3.2 Identificação da Planta

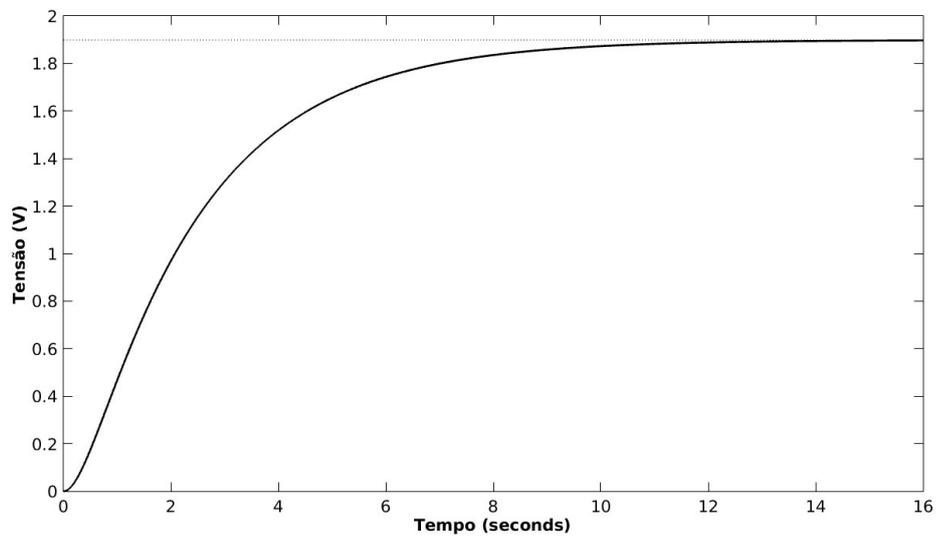
A função de transferência da planta foi obtida aplicando um sinal do tipo degrau no sistema de posicionamento em malha fechada, pois o sistema em malha aberta é instável (ver Figura 15). Com os dados de entrada e saída do sistema de posicionamento foi realizada a identificação da planta, por meio do algoritmo RLS. Na Figura 15, pode ser vista a resposta ao degrau unitário do sistema identificado em malha aberta e na Figura 16, a resposta do sistema identificado ao degrau unitário em malha fechada. Percebe-se das Figuras 15 e 16, que o sistema é instável em malha aberta, mas estável em malha fechada. Isso se dá porque a variável de saída é a posição. Quando é aplicado um nível de tensão fixo a um motor CC, este começa a girar indefinidamente. Já, se o sinal aplicado for a saída do detector de erro, o motor só irá parar quando a saída do sistema chegar próxima do sinal de referência, pois neste caso o erro é próximo de zero.

Figura 15 – Resposta ao degrau do sistema em malha aberta



Fonte: Produzido pelo autor

Figura 16 – Resposta ao degrau do sistema em malha fechada



Fonte: Produzido pelo autor

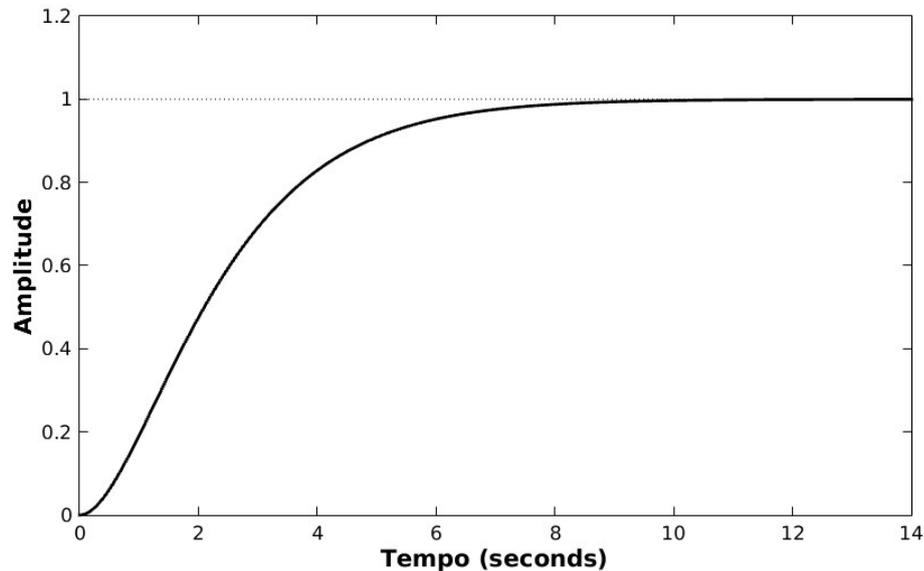
5.3.3 Modelo de Referência

O sinal de posição desejado é dado por $y_d = \omega_n^2 u_c / (s^2 + 2\zeta\omega_n s + \omega_n^2)$ em que $\omega_n = 2$ rad/s e $\zeta = 0,8$. Portanto, o sinal do modelo de referência é definido pela dinâmica dada por

$$y_d = \frac{4}{s^2 + 3,2s + 4} u_c \quad (5.15)$$

em que y_d , o sinal de trajetória desejado e o sinal de comando é u_c . A resposta ao degrau unitário da dinâmica expressa na Equação (5.15), pode ser visualizada na Figura 17.

Figura 17 – Resposta ao degrau unitário do modelo de referência



Fonte: Produzido pelo autor

5.3.4 Modelo do Atuador-Carga

O modelo do conjunto atuador-carga pode ser representado pelo seguinte sistema de equações

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(\mathbf{x}) + g(\mathbf{x})u \\ y &= x_1,\end{aligned}$$

em que x_1 e x_2 são a posição e a velocidade, respectivamente; u é a lei de controle e y é a saída do sistema.

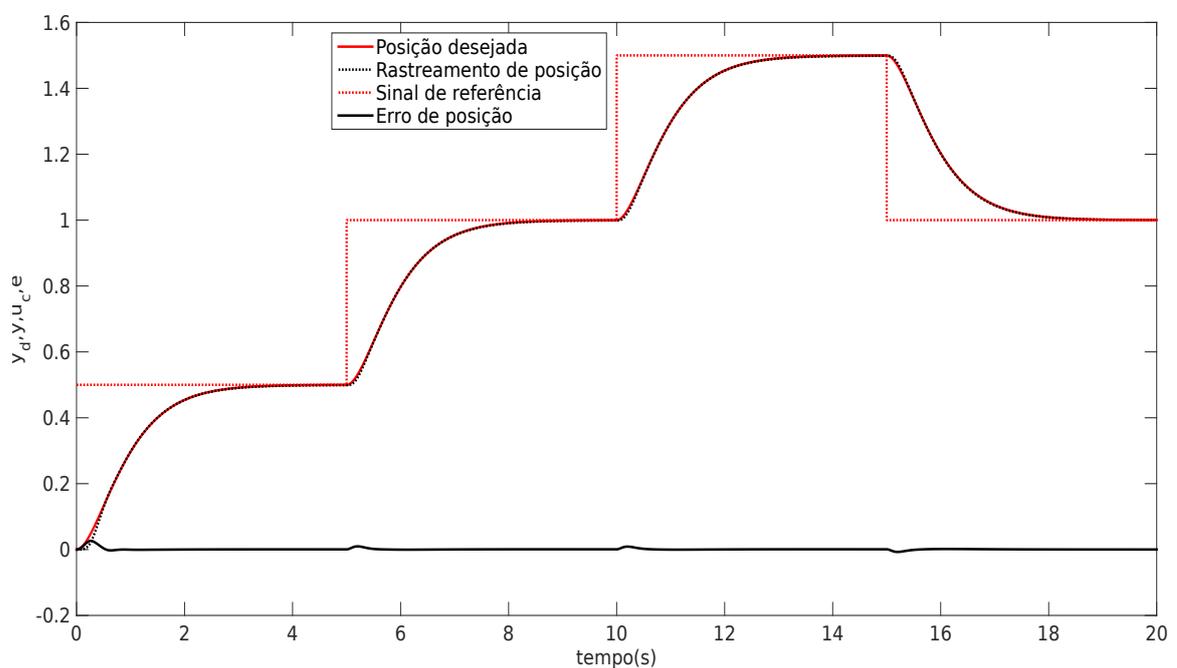
Por meio da identificação dos parâmetros do sistema obteve-se os seguintes valores de $f(\mathbf{x})$ e $g(\mathbf{x})$: $f(\mathbf{x}) = -1,219x_1 - 3,717x_2$ e $g(\mathbf{x}) = 2,315$.

Os estados iniciais da planta são nulos, pois inicialmente o sistema está em repouso. A estrutura da rede RBF é escolhida como 2-7-1, ou seja, duas unidades sensoriais, sete neurônios na camada oculta e um neurônio na camada de saída. Os parâmetros \mathbf{c} e b foram escolhidos como $\mathbf{c} = \frac{1}{10}[-3 \quad -2 \quad -1 \quad 0 \quad 1 \quad 2 \quad 3; -3 \quad -2 \quad -1 \quad 0 \quad 1 \quad 2 \quad 3]$; $b = 0.1$.

5.3.5 Experimentos Computacionais

Para a realização dos experimentos computacionais, foram empregadas a lei de controle (4.7) e a lei adaptativa (4.8). Escolhendo $\mathbf{Q} = 5000I_2$, $\beta_1 = 5$, $\beta_0 = 50$ e $\gamma = 16000$, tem-se os resultados do experimentos computacionais mostrados nas Figuras 18, 19, 20 e 21. Na Figura 18, tem-se que o erro entre o sinal do modelo de referência e o sinal de posição capturado do sensor tende a zero à medida que o tempo passa, sendo que a maior amplitude do sinal de erro foi no início do teste.

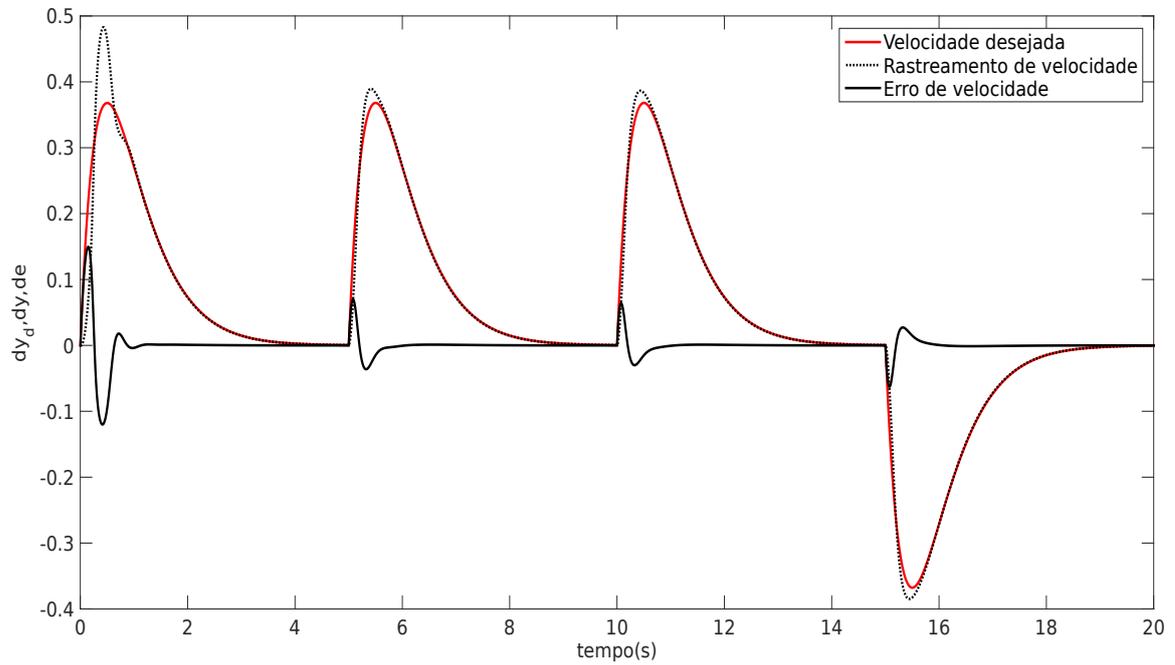
Figura 18 – Gráficos da posição



Fonte: Produzido pelo autor

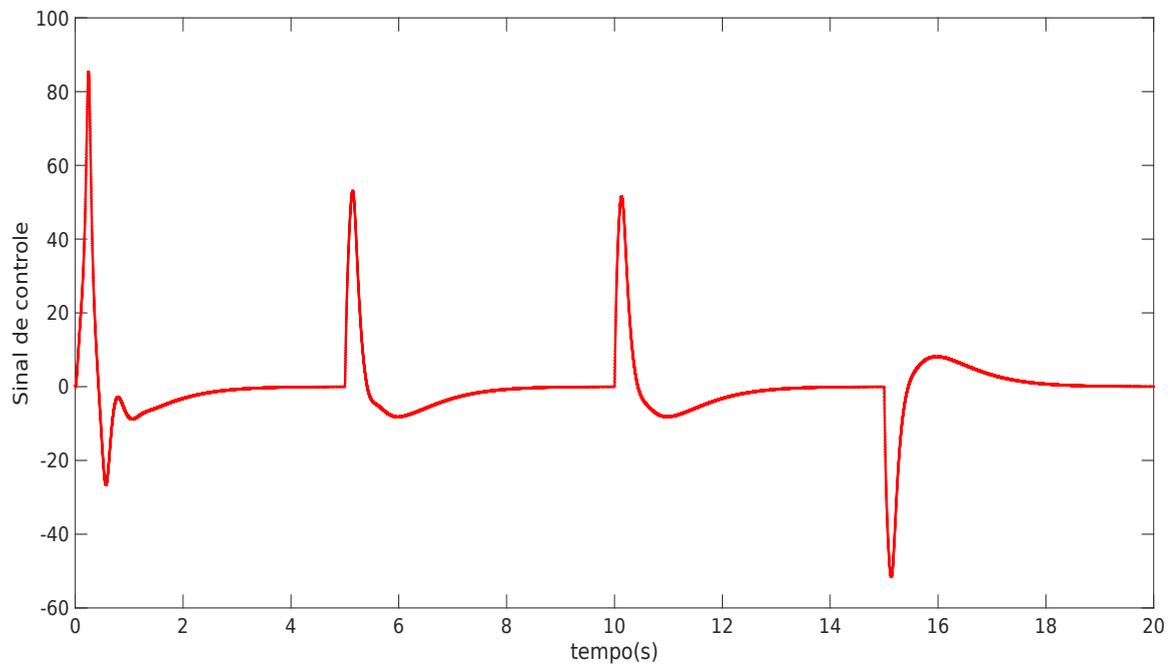
Portanto, a saída da planta consegue acompanhar bem a dinâmica estabelecida pelo modelo de referência. O sinal da velocidade estimada é mostrado na Figura 19, juntamente com a velocidade desejada. Assim como no caso anterior, as maiores amplitudes relacionadas ao sinal de erro são presenciadas no início do teste. O sinal de controle pode ser visto na Figura 20. Esse sinal apresenta picos de amplitude ao longo do teste. Esses picos ocorrem durante o transitório ocasionado pelas variações bruscas do sinal de referência no sentido de realizar a correção do sinal de saída de forma mais rápida. Durante os instantes em que o sinal de saída está próximo do sinal do modelo de referência o sinal de controle é próximo de zero.

Figura 19 – Gráficos da velocidade



Fonte: Produzido pelo autor

Figura 20 – Lei de controle

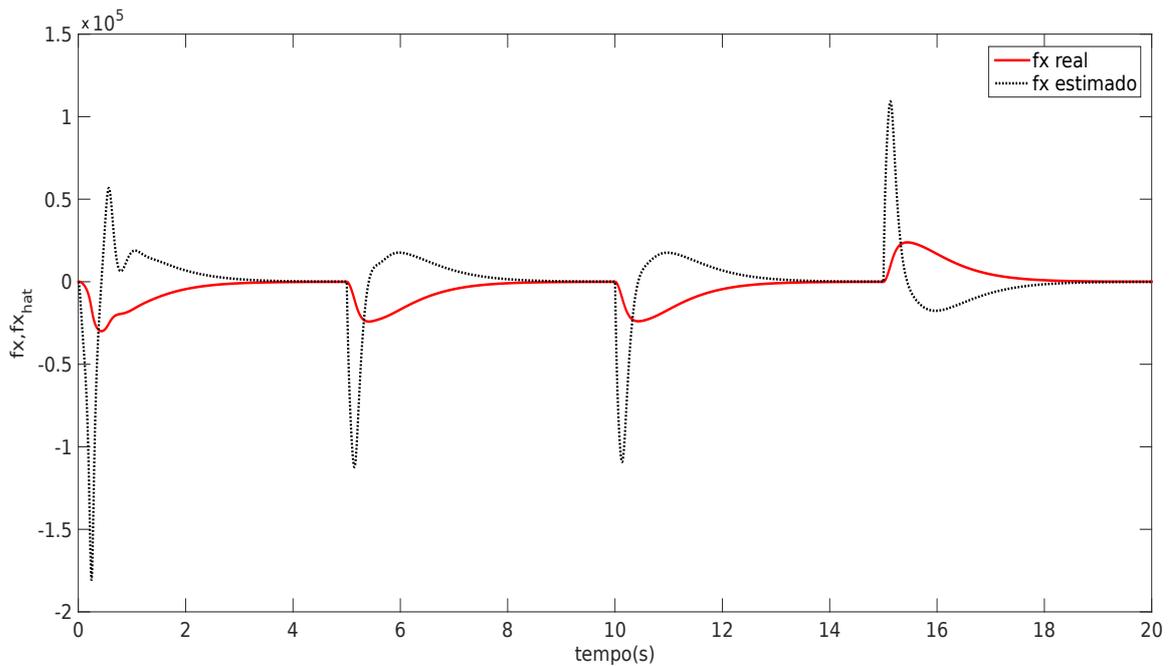


Fonte: Produzido pelo autor

Na Figura 21, são apresentados o sinal $f(x)$ e a sua estimação, $\hat{f}(x)$. Mesmo os

valores real e estimado de $f(x)$ não ficando próximos um do outro, em grande parte do tempo, o sinal de saída consegue acompanhar bem o sinal do modelo. O tempo de acomodação do sistema em malha fechada pode ser reduzido modificando-se os parâmetros ζ e ω_n , que compõem o modelo de referência.

Figura 21 – Gráfico de $\hat{f}(x)$ e $f(x)$



Fonte: Produzido pelo autor

5.4 Desenvolvimento e Montagem do SC-Neuro-Adaptativo

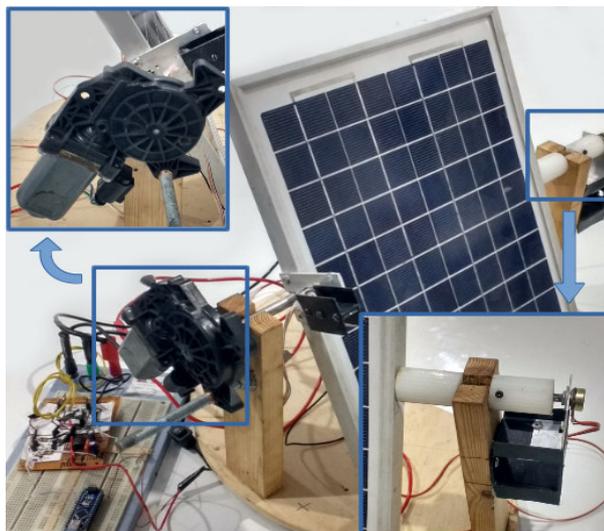
Nesta seção, apresenta-se o desenvolvimento da montagem do sistema de controle neuro-adaptativo. O projeto do referido sistema de controle é abordado em termos da unidade de controle, planta (atuador e processo) e o sensor. Além disso, é apresentada uma visão geral da planta juntamente com a unidade de potência utilizada neste trabalho.

5.4.1 Unidades SC-Neuro-Adaptativo

Os componentes do sistema de controle de malha fechada composto de controlador, sistema de acionamento, atuador, PV e sensor de posição podem ser vistos na Figura 22.

No canto superior esquerdo da Figura 22, destaca-se o atuador do sistema, um motor CC com o campo constituído de ímã permanente com alimentação de armadura de 12 V e velocidade de 75 rpm. No canto inferior direito da Figura 22, destaca-se o sensor de posição utilizado no projeto, um potenciômetro linear com resistência de 20 k Ω . A

Figura 22 – Sistema de controle em malha fechada



Fonte: Produzido pelo autor

tensão de saída do PV é de 12 V. Ao lado do circuito de acionamento, tem-se o controlador utilizado no projeto, um microcontrolador mega328p da Atmel.

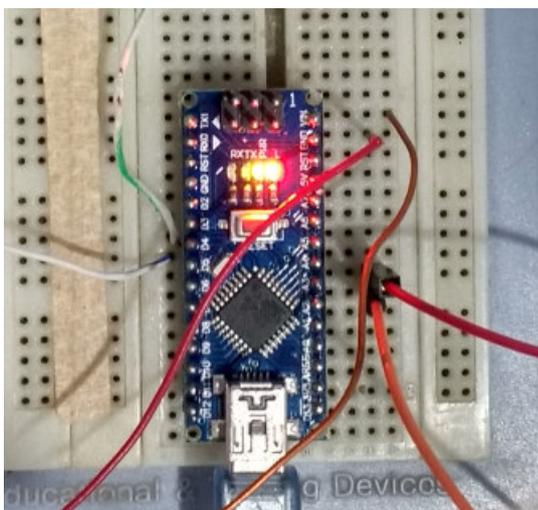
5.4.1.1 Unidade de Sensoriamento

O sensor de posição utilizado no projeto é um potenciômetro linear com resistência de 20 k Ω . A amplitude dos sinais são dados em Volts. O sistema está limitado a um ângulo de varredura menor que 360°, sendo que a tensão obtida na saída do potenciômetro varia entre 0 e 3,57 V. A tensão do sinal de comando pode variar entre 0 V e 5 V, mas está limitada aos mesmos valores da tensão lida pelo potenciômetro.

5.4.1.2 Unidade de Controle

A unidade de controle consiste de um microcontrolador mega328p da Atmel, conforme apresentado na Figura 23. Este microcontrolador contém um conversor A/D de 10 bits de resolução. Ao microcontrolador mega 328p, estão conectados como entrada o sinal de posição lido do potenciômetro, que é a variável controlada, e o sinal de controle, que é a variável manipulada. A variável manipulada é um sinal do tipo PWM (*pulse width modulation*) enviado ao sistema de acionamento juntamente com o sentido de giro do atuador. A frequência do sinal PWM é de aproximadamente 490 Hz, sendo o ciclo de trabalho variável entre 0 % e 100 %. O ciclo de trabalho indica qual o percentual de tensão da bateria é aplicado nos terminais do atuador.

Figura 23 – Microcontrolador do sistema de controle

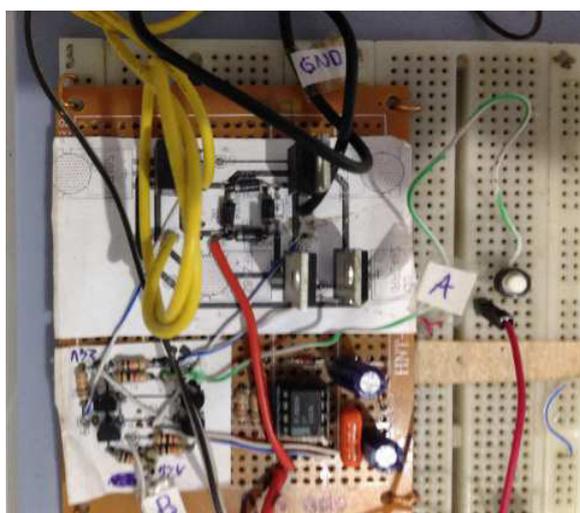


Fonte: Produzido pelo autor

5.4.2 Unidade de Potência

A vista superior da montagem da unidade de potência é apresentada na Figura 24. O sistema de acionamento é composto de um circuito de ponte H, dobrador de tensão e circuito de disparo.

Figura 24 – Montagem do sistema de acionamento



Fonte: Produzido pelo autor

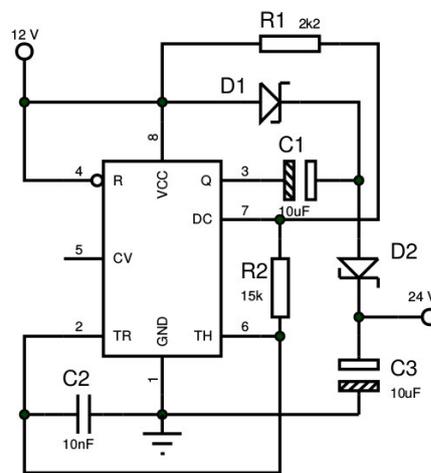
O circuito de disparo recebe o sinal de controle do microcontrolador, na forma de PWM, e envia esse sinal entre os terminais de *gate* e *source* dos MOSFETs que compõem a ponte H. A ponte H é utilizada neste projeto para permitir com que a carga possa girar

no sentido horário e no sentido anti-horário. O circuito dobrador de tensão é usado para aumentar a tensão entre os terminais de *gate* e *source* dos MOSFETs de forma que os MOSFETs funcionem na região de saturação com a menor resistência possível entre os terminais de *drain* e *source*.

5.4.2.1 Circuito Dobrador de Tensão

O circuito dobrador de tensão foi sintetizado com o oscilador 555, resistores, capacitores e diodos zeneres. Na Figura 25, apresenta-se o diagrama esquemático do circuito dobrador de tensão.

Figura 25 – Diagrama esquemático do circuito dobrador de tensão



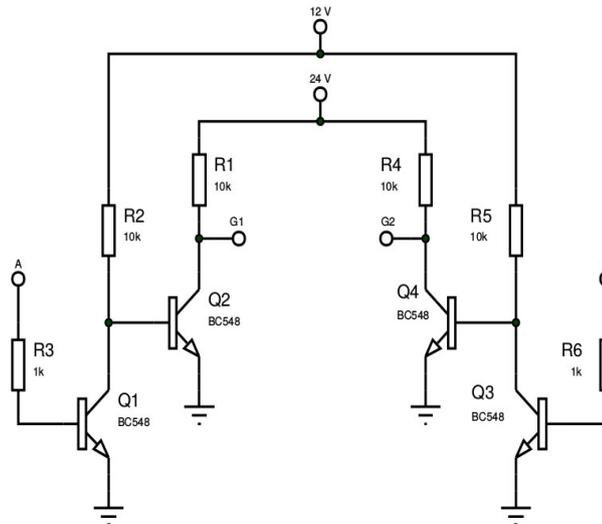
Fonte: Produzido pelo autor

Este circuito recebe uma tensão de entrada de 12 V e entrega uma tensão de saída em torno de 24 V. A tensão que este circuito entrega é utilizada para o disparo dos MOSFETs que compõem a ponte H, de forma que os MOSFETs trabalhem na região de saturação.

5.4.2.2 Circuito de Disparo

O diagrama esquemático do circuito de disparo é apresentado na Figura 26. O circuito de disparo funciona na lógica direta, ou seja, se no terminal “A”, o nível lógico for “HIGH” então no terminal “G1”, o nível lógico será “HIGH”. O mesmo raciocínio é empregado na relação entre os terminais “B” e “G2”. A presença de um nível lógico “HIGH” no terminal “A”, significa que o eixo do atuador se movimenta no sentido horário. A variação da largura do pulso PWM no terminal “A”, faz com a tensão aplicada nos terminais do atuador seja modificada. A presença de um nível lógico “HIGH” no terminal “B”, significa que o eixo do atuador se movimenta no sentido anti-horário. É importante

Figura 26 – Diagrama esquemático circuito de disparo



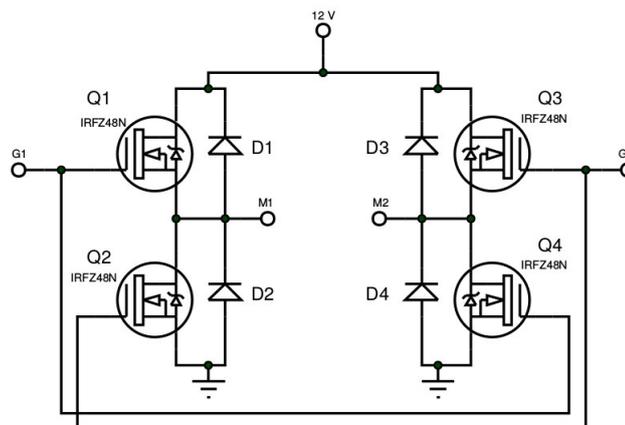
Fonte: Produzido pelo autor

ressaltar que os terminais “A” e “B” não podem ter sinais lógicos “HIGH” ao mesmo tempo, pois isso faz com que todos os MOSFETs da ponte H conduzam ao mesmo tempo provocando, dessa forma, um curto circuito na fonte.

5.4.2.3 Circuito da Ponte H

O circuito esquemático da ponte H (ver Figura 27) é constituído de quatro MOSFETs de canal N e quatro diodos. Os MOSFETs funcionam aos pares. Quando o sentido

Figura 27 – Diagrama esquemático circuito da ponte H



Fonte: Produzido pelo autor

de giro é horário os MOSFETs “Q1” e “Q4” estão funcionando na região de saturação e os MOSFETs “Q2” e “Q3” estão funcionando na região de corte. Quando o sentido

de giro é anti-horário os MOSFETs “Q2” e “Q3” estão funcionando na região de saturação e os MOSFETs “Q1” e “Q4” estão funcionando na região de corte. Os diodos em anti-paralelo são usados para proteger o circuito contra a tensão reversa causada pelos terminais no motor na ausência súbita de corrente entre os seus terminais. Isso ocorre ao retirar, instantaneamente, a tensão entre os terminais do atuador (borda de descida do sinal PWM).

Na Tabela 1, é mostrada a tabela verdade com os níveis dos sinais lógicos nos terminais “A” e “B”, conforme Figura 26, juntamente com o sentido de giro do eixo do atuador. O estado não permitido significa que a fonte de tensão estará em curto-circuito.

Tabela 1 – Tabela verdade do sentido de giro do atuador

A	B	Sentido de giro
LOW	LOW	Parado
LOW	HIGH	Anti-horário
HIGH	LOW	Horário
HIGH	HIGH	Não permitido

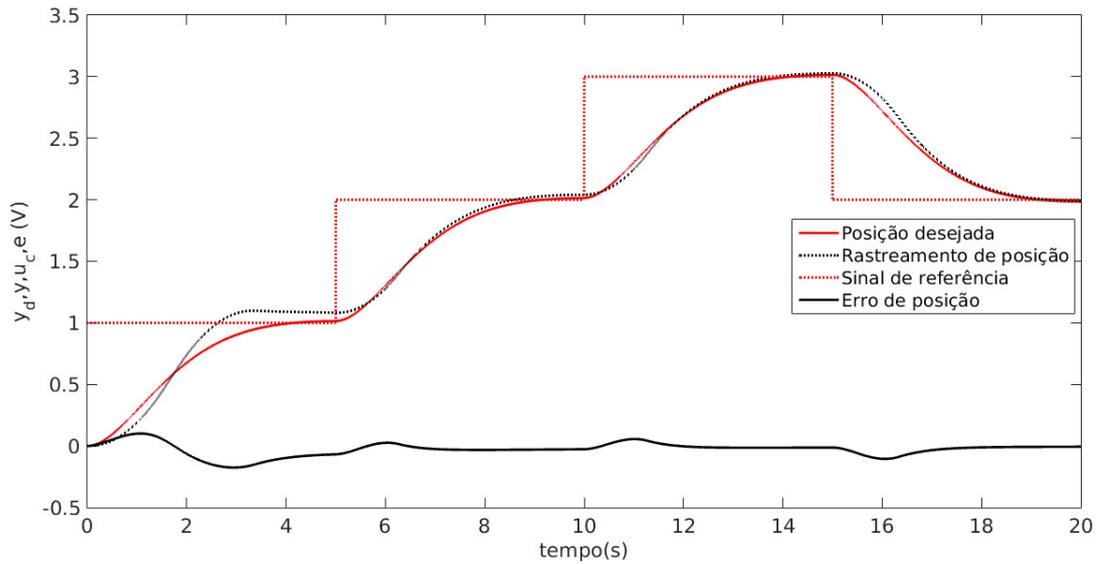
Fonte: Produzido pelo autor

5.5 Experimentos em *Hardware*

Os experimentos em *hardware* consistem dos testes de validação da metodologia proposta aplicada ao rastreamento de um alvo móvel, ou seja, o rastreador para posicionar um painel solar.

O desempenho do sistema é avaliado como seguidor. É importante ressaltar que nos testes de validação o sinal de controle ficou limitado entre -12 V e 12 V, pois a tensão nominal da fonte é de 12 V. Isso faz com que o desempenho geral do sistema seja comprometido. Na Figura 28, é apresentada o sinal de posição do sistema juntamente com os sinais do modelo de referência, o sinal de comando do sistema e a diferença entre eles. O sinal de erro apresenta a maior amplitude no início do teste. A amplitude dos sinais é dada em Volts. O sistema está limitado a um ângulo de varredura menor que 360°sendo que a tensão obtida do potenciômetro varia entre 0 V e 3,57 V e a tensão do sinal de comando podendo ser entre 0 V e 5 V, mas limitada aos mesmos valores da tensão lida pelo potenciômetro.

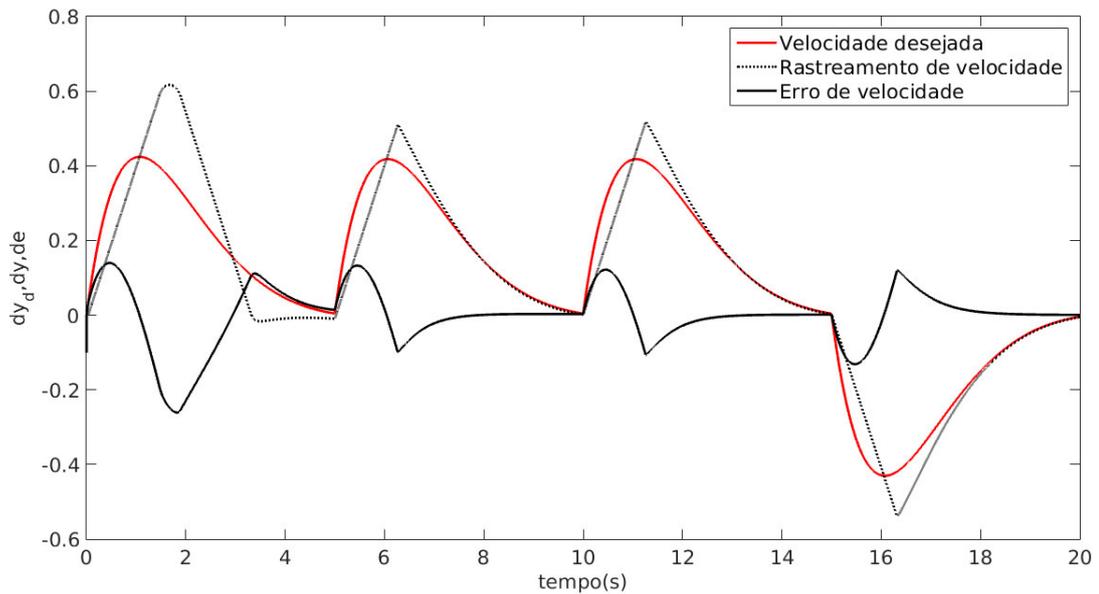
Figura 28 – Posição do seguidor



Fonte: Produzido pelo autor

Na Figura 29, percebe-se que o desempenho da velocidade do seguidor foi comprometida. Essa situação deve-se ao fato do sinal de controle ser limitado entre -12 V e 12 V.

Figura 29 – Velocidade do seguidor

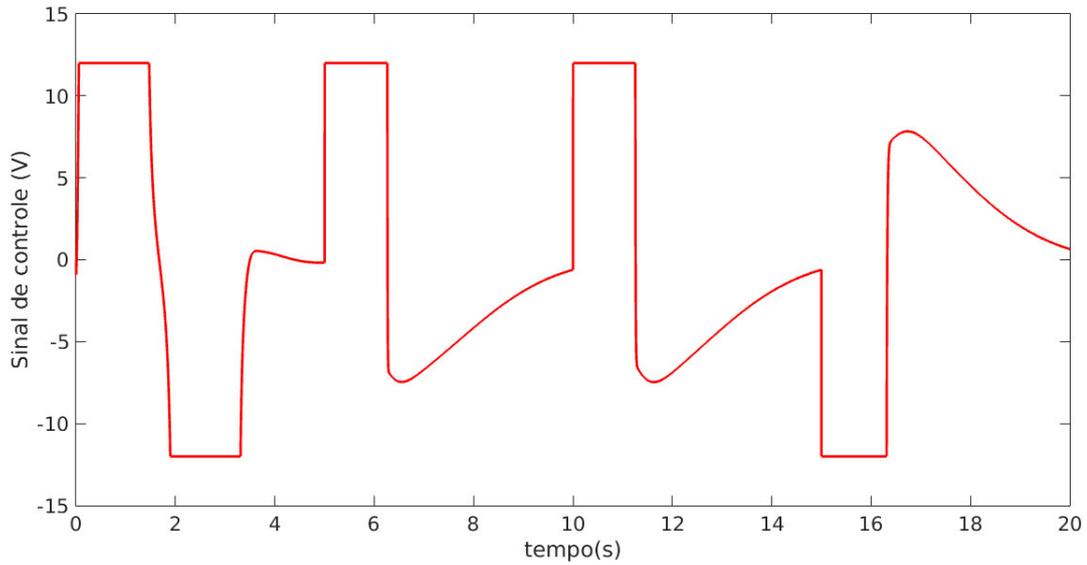


Fonte: Produzido pelo autor

Como pode ser visto na Figura 30, o sinal de controle é limitado entre -12 V e 12 V. A saturação do sinal de controle é obtida várias vezes ao longo do teste. Essa é a causa

do desempenho do sistema ser comprometido.

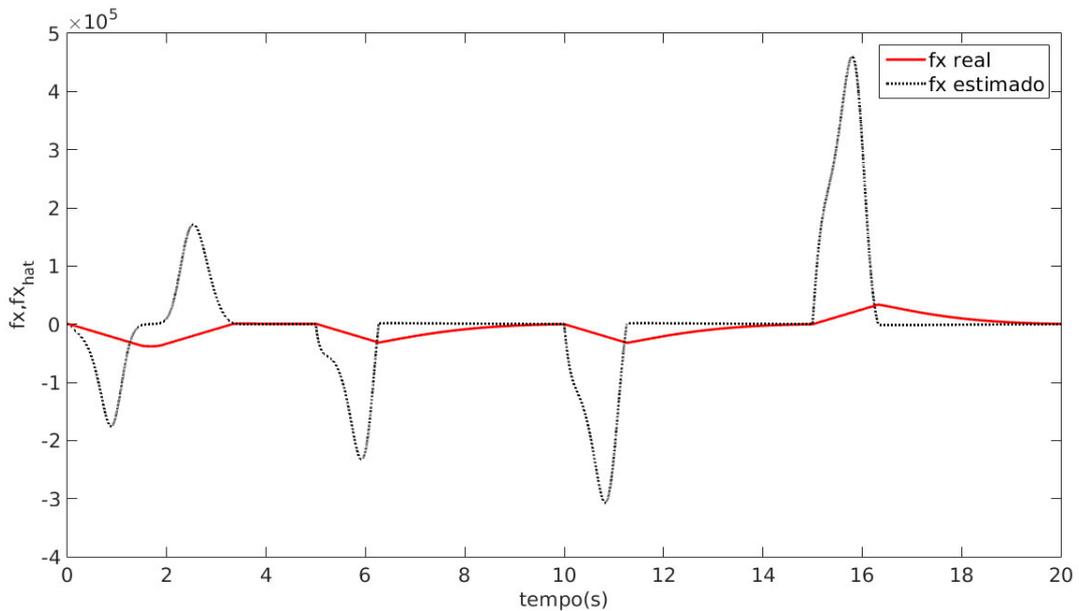
Figura 30 – Lei de controle – seguidor



Fonte: Produzido pelo autor

Na Figura 31, mostra-se o gráfico de $\hat{f}(x)$ e $f(x)$ referentes ao teste de validação. Percebe-se que mesmo os valores real e estimado de $f(x)$ não ficando próximos um do outro em grande parte do tempo o sinal de saída consegue acompanhar o sinal do modelo.

Figura 31 – Gráfico de $\hat{f}(x)$ e $f(x)$ – seguidor



Fonte: Produzido pelo autor

5.6 Comentários

Neste capítulo, foi apresentado o desenvolvimento dos modelos, dos procedimentos e dos algoritmos para o projeto *online* de sistemas de controle neuro-adaptativo. Além disso, foram apresentados os resultados dos experimentos computacionais, que foram obtidos por meio de simulações de modelos matemáticos, e dos experimentos em *hardware*, que foram obtidos em uma planta do mundo real. É importante salientar que nos resultados do teste de validação existe a questão da saturação do sinal de controle e que o mesmo está limitado entre -12 V e 12 V. Em virtude disso, o sinal de saída do teste de validação não fica muito próximo do sinal de saída, conforme os resultados obtidos nos experimentos computacionais.

Conclusão

Nesta dissertação, apresentou-se uma metodologia de controle adaptativo, por modelo de referência, baseada em redes neurais de função de base radial e controle por modos deslizantes, para o controle de sistemas dinâmicos. Inicialmente, apresentou-se um estudo acerca de controle adaptativo e as redes neurais artificiais. Neste estudo, aborda-se o controle adaptativo por escalonamento de ganhos e o regulador autoajustável. O escalonamento de ganhos é bem sucedido em muitas aplicações, porém a carga computacional é elevada devido à necessidade de calcular muitos controladores lineares. Já no regulador autoajustável, o ajuste dos parâmetros do controlador é realizado de forma a obter as propriedades desejadas do sistema em malha fechada. Esse ajuste pode ser feito de forma direta ou indireta. Em ambos os casos, o algoritmo de estimação paramétrica empregado é o RLS. Além disso, é abordado o algoritmo de alocação de polos de grau mínimo para o projeto de sistemas de controle em malha fechada. Em relação às redes neurais artificiais, foi apresentada a arquitetura das redes perceptron de múltiplas camadas, juntamente com o algoritmo *backpropagation*. A regra de aprendizagem utilizada é a regra de aprendizagem de Widrow-Hoff. O algoritmo *backpropagation* é frequentemente empregado em redes do tipo MLP.

Em seguida, apresentou-se as redes neurais de funções de base radial e realizou-se um estudo comparativo entre as redes MLP e RBF, no que concerne às suas arquiteturas, funções de ativação e treinamento. Salientou-se, ainda que o algoritmo *backpropagation* é frequentemente empregado em redes do tipo MLP, porém ele não é utilizado em redes do tipo RBF, porque neste tipo de rede não existe a etapa de retropropagação. Apresentou-se o sistema adaptativo por modelo de referência. A proposta do MRAS é resolver o problema em que as especificações de desempenho são dadas em termos de um modelo de referência. Mostrou-se que o método do gradiente (regra MIT) como mecanismo de ajuste dos parâmetros no sistema não possui garantia de estabilidade do sistema em malha fechada. Porém, pode-se fazer uso da teoria de estabilidade de Lyapunov para resolver esta questão. Outrossim, foi apresentada a filosofia do controle de modos deslizantes, que

é um tipo de controle de estrutura variável.

Para o projeto *online* do sistema de controle neuro-adaptativo, apresentou-se o controle adaptativo baseado em RBF, juntamente com a análise de estabilidade. Também, foram apresentados os algoritmos de estimação e de controle. Apresentou-se o problema que se quer resolver, juntamente com a sua solução. Fez-se referência aos circuitos eletrônicos que constituem a unidade de potência do sistema, bem como o microcontrolador utilizado e o sensor de posicionamento.

Por fim, foram apresentados os resultados, experimentais e em *hardware*, obtidos com este trabalho. A metodologia apresentada, foi aplicada em um sistema de rastreamento de trajetória solar. Por meio dos resultados obtidos, percebeu-se que a resposta do sistema dinâmico seguiu, de forma satisfatória, a trajetória esperada, que é especificada pela dinâmica desejada do modelo de referência.

6.1 Trabalhos Futuros

A seguir listam-se alguns trabalhos futuros que podem ser desenvolvidos a partir da proposta apresentada. Os temas ou itens para os trabalhos futuros são:

1. Validação em outros sistemas dinâmicos, tal como um aeroestabilizador;
2. Avaliação da complexidade computacional do sistema proposto;
3. Comparação com outros tipos de controladores, tal como o LQR.

Referências

- AHMED, R. et al. Automotive internal-combustion-engine fault detection and classification using artificial neural network techniques. *IEEE Transactions on vehicular technology*, IEEE, v. 64, n. 1, p. 21–33, 2015. Citado na página 33.
- ASADI, H. et al. Robust optimal motion cueing algorithm based on the linear quadratic regulator method and a genetic algorithm. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, IEEE, 2016. Citado na página 33.
- ÅSTRÖM, K. J.; EYKHOFF, P. System identification—a survey. *Automatica*, Elsevier, v. 7, n. 2, p. 123–162, 1971. Citado na página 27.
- ÅSTRÖM, K. J.; WITTENMARK, B. On self tuning regulators. *Automatica*, Elsevier, v. 9, n. 2, p. 185–199, 1973. Citado na página 27.
- ÅSTRÖM, K. J.; WITTENMARK, B. *Adaptive control*. [S.l.]: Courier Corporation, 2013. Citado 6 vezes nas páginas 16, 20, 21, 22, 25 e 41.
- BELLMAN, R. *Dynamic Programming*. [S.l.]: Princeton University Press, 1957. Citado na página 27.
- BHATTACHARJEE, K. K.; SARMAH, S. P. Shuffled frog leaping algorithm and its application to 0/1 knapsack problem. *Applied Soft Computing*, Elsevier, v. 19, p. 252–263, 2014. Citado na página 33.
- BISHOP, C. *Neural Networks for Pattern Recognition*. [S.l.]: Clarendon Press, 2005. (Advanced Texts in Econometrics). ISBN 9780198538646. Citado na página 40.
- CASTAÑEDA, H. et al. Continuous differentiator based on adaptive second-order sliding-mode control for a 3-dof helicopter. *IEEE Transactions on Industrial Electronics*, IEEE, v. 63, n. 9, p. 5786–5793, 2016. Citado na página 34.
- CHEN, C.-T. Analog and digital control system design. 2015. Citado 2 vezes nas páginas 56 e 87.
- CHINE, W. et al. A novel fault diagnosis technique for photovoltaic systems based on artificial neural networks. *Renewable Energy*, Elsevier, v. 90, p. 501–512, 2016. Citado na página 33.

- CLARKE, D. W.; GAWTHROP, P. J. Self-tuning controller. In: IET. *Proceedings of the Institution of Electrical Engineers*. [S.l.], 1975. v. 122, n. 9, p. 929–934. Citado na página 27.
- DATTA, A.; IOANNOU, P. A. Performance improvement versus robust stability in model reference adaptive control. In: IEEE. *Decision and Control, 1991., Proceedings of the 30th IEEE Conference on*. [S.l.], 1991. p. 748–753. Citado na página 28.
- DUAN, H.; LI, P. *Bio-inspired computation in unmanned aerial vehicles*. [S.l.]: Springer, 2014. Citado na página 15.
- EGARDT, B. *Stability of adaptive controllers*. [S.l.]: Springer, 1979. v. 20. Citado 2 vezes nas páginas 27 e 28.
- FEL'DBAUM, A. A. *Optimal Control Systems*. [S.l.]: Elsevier, 1965. v. 22. Citado na página 27.
- GE, S. S. et al. *Stable adaptive neural network control*. [S.l.]: Springer Science & Business Media, 2013. v. 13. Citado na página 48.
- GOODWIN, G. *Adaptive filtering prediction and control*. [S.l.]: Prentice Hall, 1984. Citado na página 27.
- GOODWIN, G.; RAMADGE, P.; CAINES, P. Discrete-time multivariable adaptive control. *IEEE Transactions on Automatic Control*, IEEE, v. 25, n. 3, p. 449–456, 1980. Citado na página 27.
- GU, X.; SHAO, C. Robust adaptive control of time-varying linear plants using polynomial approximation. In: IET. *IEE Proceedings D-Control Theory and Applications*. [S.l.], 1993. v. 140, n. 2, p. 111–118. Citado na página 28.
- GUO, L. et al. Intelligent vehicle trajectory tracking based on neural networks sliding mode control. In: IEEE. *Informative and Cybernetics for Computational Social Systems (ICCSS), 2014 International Conference on*. [S.l.], 2014. p. 57–62. Citado na página 33.
- HAFEZI, R.; SHAHRABI, J.; HADAVANDI, E. A bat-neural network multi-agent system (bnnmas) for stock price prediction: Case study of dax stock price. *Applied Soft Computing*, Elsevier, v. 29, p. 196–210, 2015. Citado na página 33.
- HARRIS, C. J.; BILLINGS, S. *Self-tuning and adaptive control: theory and applications*. [S.l.]: Peter Peregrinus London, 1981. Citado na página 27.
- HAYKIN, S. S. et al. *Neural networks and learning machines*. [S.l.]: Pearson Upper Saddle River, NJ, USA:, 2009. v. 3. Citado 8 vezes nas páginas 15, 30, 31, 32, 36, 37, 40 e 82.
- HE, W.; CHEN, Y.; YIN, Z. Adaptive neural network control of an uncertain robot with full-state constraints. *IEEE Transactions on Cybernetics*, IEEE, v. 46, n. 3, p. 620–629, 2016. Citado na página 33.
- HE, W. et al. Neural network control of a robotic manipulator with input deadzone and output constraint. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, IEEE, v. 46, n. 6, p. 759–770, 2016. Citado na página 33.

- HEBB, D. O. *The organization of behavior: A neuropsychological approach*. [S.l.]: John Wiley & Sons, 1949. Citado na página 32.
- HO, S.-J.; CHEN, B.-S. Low design-cost fuzzy controllers for robust stabilization of nonlinear partial differential systems. *IEEE Transactions on Fuzzy Systems*, IEEE, v. 24, n. 6, p. 1378–1394, 2016. Citado na página 33.
- HOPFIELD, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, National Acad Sciences, v. 79, n. 8, p. 2554–2558, 1982. Citado na página 32.
- IOANNOU, P.; DATTA, A. Robust adaptive control: Design, analysis and robustness bounds. In: *Foundations of adaptive control*. [S.l.]: Springer, 1991. p. 71–152. Citado na página 28.
- IOANNOU, P.; SUN, J. Theory and design of robust direct and indirect adaptive-control schemes. *International Journal of Control*, Taylor & Francis, v. 47, n. 3, p. 775–813, 1988. Citado na página 28.
- IOANNOU, P. A. Adaptive systems with reduced models. *Lecture notes in control and information sciences*, Springer-Verlag, 1983. Citado na página 28.
- ISHAQUE, K.; SALAM, Z. A deterministic particle swarm optimization maximum power point tracker for photovoltaic system under partial shading condition. *IEEE transactions on industrial electronics*, IEEE, v. 60, n. 8, p. 3195–3206, 2013. Citado na página 33.
- KALMAN, R. E. Design of a self-optimising control system. *Trans. ASME*, v. 80, p. 468–478, 1958. Citado na página 26.
- KANELLAKOPOULOS, I. *Adaptive control of nonlinear systems*. Tese (Doutorado) — University of Illinois at Urbana-Champaign, 1992. Citado na página 28.
- KANELLAKOPOULOS, I.; KOKOTOVIC, P. V.; MORSE, A. S. Systematic design of adaptive controllers for feedback linearizable systems. *IEEE Transactions on Automatic control*, IEEE, v. 36, n. 11, p. 1241–1253, 1991. Citado na página 28.
- KAR, A. K. Bio inspired computing – a review of algorithms and scope of applications. *Expert Systems with Applications*, Elsevier, v. 59, p. 20–32, 2016. Citado 2 vezes nas páginas 15 e 36.
- LANDAU, I. *Adaptive control: the model reference approach*. [S.l.]: Marcel Dekker, Inc., New York, 1979. Citado na página 27.
- LANDAU, I. Combining model reference adaptive controllers and stochastic self-tuning regulators. *Automatica*, Elsevier, v. 18, n. 1, p. 77–84, 1982. Citado na página 27.
- LANDAU, I. D. et al. *Adaptive and Robust Active Vibration Control: Methodology and Tests*. 1. ed. [S.l.]: Springer International Publishing, 2017. (Advances in Industrial Control). ISBN 978-3-319-41450-8,978-3-319-41449-2. Citado 2 vezes nas páginas 21 e 41.
- LANDAU, I. D. et al. *Adaptive control: algorithms, analysis and applications*. [S.l.]: Springer Science & Business Media, 2011. Citado 3 vezes nas páginas 16, 21 e 57.

- LEE, J. H. et al. A novel memetic algorithm using modified particle swarm optimization and mesh adaptive direct search for pmsm design. *IEEE Transactions on Magnetics*, IEEE, v. 52, n. 3, p. 1–4, 2016. Citado na página 33.
- MACHADO, M. C.; SANTOS, H. S. S.; NETO, J. V. d. F. Online project of adaptive control systems applied to the problem of solar tracking. *International Journal of Modern Engineering Research*, v. 6, n. 12, p. 50–58, December 2016. Citado na página 26.
- MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, Springer, v. 5, n. 4, p. 115–133, 1943. Citado na página 31.
- MIDDLETON, R. H.; GOODWIN, G. C. Adaptive control of time-varying linear systems. *IEEE Transactions on Automatic Control*, IEEE, v. 33, n. 2, p. 150–155, 1988. Citado na página 28.
- MINSKY, M.; PAPERT, S. Perceptrons. MIT press, 1969. Citado na página 32.
- MORENO-VALENZUELA, J. et al. Adaptive neural network control for the trajectory tracking of the furuta pendulum. *IEEE Transactions on Cybernetics*, IEEE, v. 46, n. 12, p. 3439–3452, 2016. Citado na página 28.
- MORSE, A. Global stability of parameter-adaptive control systems. *IEEE Transactions on Automatic Control*, IEEE, v. 25, n. 3, p. 433–439, 1980. Citado na página 27.
- NARENDRA, K.; LIN, Y.-H.; VALAVANI, L. Stable adaptive controller design, part ii: Proof of stability. *IEEE Transactions on Automatic Control*, IEEE, v. 25, n. 3, p. 440–448, 1980. Citado na página 27.
- NARENDRA, K. S.; MONOPOLI, R. *Applications of adaptive control*. [S.l.]: Academic Press, New York, 1980. Citado na página 27.
- NIU, H.; WANG, J. Financial time series prediction by a random data-time effective rbf neural network. *Soft Computing*, Springer, v. 18, n. 3, p. 497–508, 2014. Citado na página 33.
- PUNITHA, K.; DEVARAJ, D.; SAKTHIVEL, S. Artificial neural network based modified incremental conductance algorithm for maximum power point tracking in photovoltaic system under partial shading conditions. *Energy*, Elsevier, v. 62, p. 330–340, 2013. Citado na página 33.
- ROHRS, C. et al. Robustness of continuous-time adaptive control algorithms in the presence of unmodeled dynamics. *IEEE Transactions on Automatic Control*, IEEE, v. 30, n. 9, p. 881–889, 1985. Citado na página 28.
- ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, American Psychological Association, v. 65, n. 6, p. 386, 1958. Citado na página 32.
- RUMELHART, D. E. et al. Parallel distributed processing, vol. 1&2. *Cambridge, MA: The MIT Press*, 1986. Citado na página 32.

- SANTOS, R. B. et al. Comparison between multilayer feedforward neural networks and a radial basis function network to detect and locate leaks in pipelines transporting gas. *Chem. Eng. Trans.*, v. 32, n. 1375, p. e1380, 2013. Citado na página 40.
- SILVA, I. N. d. et al. *Artificial Neural Networks : A Practical Course*. 1. ed. [S.l.]: Springer International Publishing, 2017. ISBN 978-3-319-43162-8,978-3-319-43161-1. Citado 3 vezes nas páginas 29, 30 e 36.
- SLOTINE, J.-J. E.; LI, W. et al. *Applied nonlinear control*. [S.l.]: Prentice-Hall Englewood Cliffs, NJ, 1991. v. 199. Citado 3 vezes nas páginas 42, 44 e 47.
- SUN, C.; HE, W.; HONG, J. Neural network control of a flexible robotic manipulator using the lumped spring-mass model. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, IEEE, 2016. Citado na página 34.
- SUN, J. A modified model reference adaptive control scheme for improved transient performance. *IEEE Transactions on Automatic Control*, IEEE, v. 38, n. 8, p. 1255–1259, 1993. Citado na página 28.
- SUN, W.; GAO, H.; KAYNAK, O. Adaptive backstepping control for active suspension systems with hard constraints. *IEEE/ASME transactions on mechatronics*, IEEE, v. 18, n. 3, p. 1072–1079, 2013. Citado na página 34.
- TAY, T.; MAREELS, I.; MOORE, J. *High Performance Control*. 1. ed. [S.l.]: Birkhäuser Boston, 1998. (Systems & Control: Foundations & Applications). Citado na página 15.
- TSAKALIS, K. S.; LIMANOND, S. Asymptotic performance guarantees in adaptive control. *International journal of adaptive control and signal processing*, Wiley Online Library, v. 8, n. 2, p. 173–199, 1994. Citado na página 28.
- TSYPKIN, Y. Z.; NIKOLIC, Z. J. *Adaptation and learning in automatic systems*. [S.l.]: Academic Press New York, 1971. v. 73. Citado na página 27.
- WANG, L.-X. *A course in fuzzy systems*. [S.l.]: Prentice-Hall press, USA, 1999. Citado 4 vezes nas páginas 42, 43, 44 e 48.
- WANG, T.; GAO, H.; QIU, J. A combined adaptive neural network and nonlinear model predictive control for multirate networked industrial process control. IEEE, 2015. Citado na página 33.
- WHITAKER, H. P.; YAMRON, J.; KEZER, A. *Design of model-reference adaptive control systems for aircraft*. [S.l.]: Massachusetts Institute of Technology, Instrumentation Laboratory, 1958. Citado na página 26.
- WIDROW, B.; HOFF, M. E. et al. Adaptive switching circuits. In: NEW YORK. *IRE WESCON convention record*. [S.l.], 1960. v. 4, n. 1, p. 96–104. Citado na página 32.
- XIE, T.; YU, H.; WILAMOWSKI, B. Comparison between traditional neural networks and radial basis function networks. In: IEEE. *Industrial Electronics (ISIE), 2011 IEEE International Symposium on*. [S.l.], 2011. p. 1194–1199. Citado na página 40.
- YU, H. et al. Advantages of radial basis function networks for dynamic system design. *IEEE Transactions on Industrial Electronics*, IEEE, v. 58, n. 12, p. 5438–5450, 2011. Citado na página 40.

ZHANG, M. et al. Adaptive tracking control for double-pendulum overhead cranes subject to tracking error limitation, parametric uncertainties and external disturbances. *Mechanical Systems and Signal Processing*, Elsevier, v. 76, p. 15–32, 2016. Citado na página 28.

ZHANG, S. et al. Adaptive neural network control of coordinated robotic manipulators with output constraint. *IET Control Theory & Applications*, IET, v. 10, n. 17, p. 2271–2278, 2016. Citado na página 32.

ZHAO, F. et al. Adaptive neural network control for active suspension system with actuator saturation. *IET Control Theory and Applications*, IET, v. 10, n. 14, p. 1696–1705, 2016. Citado na página 34.

ZHOU, Q. et al. Observer-based adaptive neural network control for nonlinear stochastic systems with time delay. *IEEE Transactions on Neural Networks and Learning Systems*, IEEE, v. 24, n. 1, p. 71–80, 2013. Citado na página 33.

Apêndices



Redes Neurais Artificiais e Sintonia

O desenvolvimento dos paradigmas de redes neurais artificiais é apresentado no contexto de conceitos básicos, arquiteturas e métodos de treinamento.

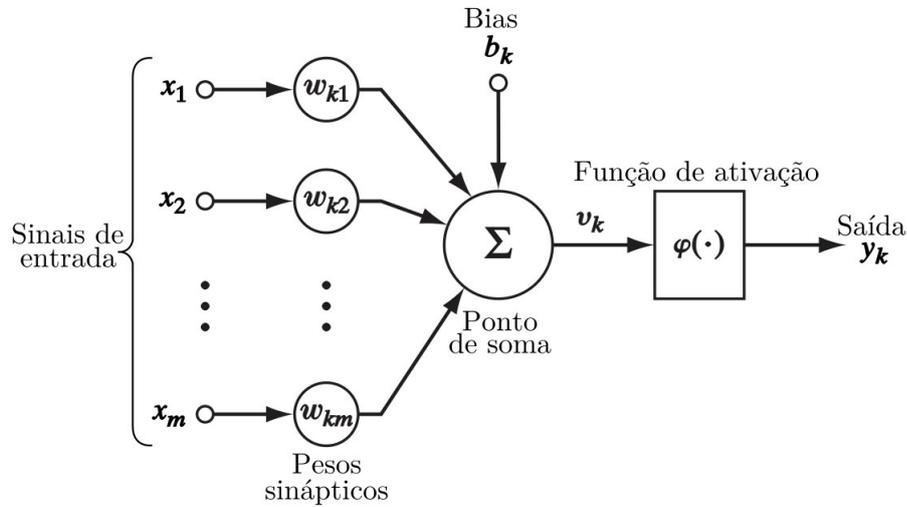
A.1 Modelo do Neurônio Não Linear

Inicialmente, na Figura 32, é apresentado o modelo da unidade básica dessas redes, o neurônio não linear. Esse neurônio pode ser identificado por três elementos básicos:

1. Um conjunto de sinapses ou pesos sinápticos. Essas ponderações são representadas por w_{kj} . Isso significa que o sinal de entrada, x_j , é conectado ao neurônio k através do peso sináptico w_{kj} . Ao contrário da ponderação de uma sinapse no cérebro, o peso sináptico de um neurônio artificial pode situar-se em um intervalo que inclui tanto valores negativos como valores positivos.
2. Um combinador linear, responsável por realizar a soma dos sinais de entrada ponderados pelas respectivas sinapses ao neurônio.
3. Uma função de ativação para limitar a amplitude da saída do neurônio. A função de ativação é também chamada de função de esmagamento, em que a amplitude permissível do sinal de saída é limitada para algum valor finito. Geralmente a saída de um neurônio é normalizada entre uma faixa de 0 a 1 ou alternativamente de -1 a 1.

No modelo do neurônio da Figura 32, tem um fator externo aplicado ao neurônio chamado de bias, b_k . O bias, tem como consequência aplicar uma transformação afim ao combinador linear para aumentar ou diminuir o valor de v_k , dependendo se o seu sinal é positivo ou negativo.

Figura 32 – Modelo de um neurônio não linear com múltiplas entradas



Fonte: Haykin et al. (2009)

Para o neurônio k da Figura 32, o campo local induzido, v_k , é definido por

$$v_k = \sum_{j=1}^m w_{kj} x_j + b_k \quad (\text{A.1})$$

e a saída, y_k , do neurônio k é dada por

$$y_k = \varphi(v_k) \quad (\text{A.2})$$

em que x_1, x_2, \dots, x_m são os sinais de entrada; $w_{k1}, w_{k2}, \dots, w_{km}$ são os pesos sinápticos do neurônio k ; b_k é o bias e $\varphi(\cdot)$ é a função de ativação.

Dentre as funções de ativação empregadas nas redes neurais artificiais do tipo MLP, as mais utilizadas são a função logística e a tangente hiperbólica. Já quando se trata de RNAs do tipo função de base radial a mais utilizada é a função gaussiana.

As funções de ativação são divididas em dois grandes grupos: funções parcialmente diferenciáveis e as funções totalmente diferenciáveis.

Funções de ativação parcialmente diferenciáveis: As funções de ativação parcialmente diferenciáveis são funções de ativação que possuem pontos cuja primeira derivada não existe. Nessa categoria as mais utilizadas são:

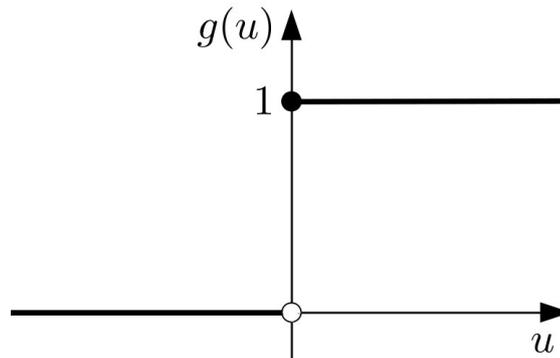
a) Função de limiar ou função de Heaviside

A função de Heaviside pode ser vista na Figura 33. A saída dessa função de ativação produz zero se o seu argumento é menor que zero e produz 1 se o seu argumento é maior ou igual a 0. Essa função de ativação é utilizada para classificar entradas em

dois grupos distintos. Assim, tem-se que

$$g(u) = \begin{cases} 1, & \text{se } u \geq 0 \\ 0, & \text{se } u < 0 \end{cases} \quad (\text{A.3})$$

Figura 33 – Função de Heaviside



Fonte: Produzido pelo autor

b) Função linear por partes

Os resultados desta função de ativação são iguais aos valores do campo local induzido quando definidos dentro do intervalo $[-a, a]$, e limitados aos valores inferiores e superiores, caso contrário. A função linear por partes pode ser vista na Figura 34, e matematicamente é definida por

$$g(u) = \begin{cases} -a, & \text{se } u < -a \\ u, & \text{se } -a \leq u \leq a \\ a, & \text{se } u > a \end{cases} \quad (\text{A.4})$$

Funções de ativação totalmente diferenciáveis: As funções de ativação totalmente diferenciáveis são funções de ativação cuja primeira derivada existe para todos os pontos do seu domínio de definição. Nessa categoria, as mais utilizadas são:

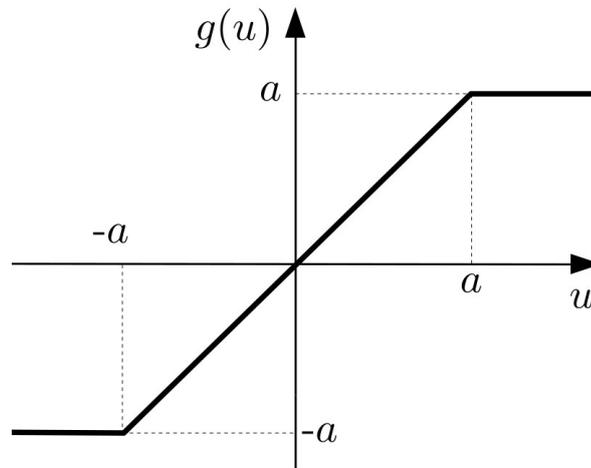
a) Função Logística

O resultado da função logística sempre assumirá valores reais entre zero e um. Sua expressão matemática é dada por

$$g(u) = \frac{1}{1 + \exp(-a \cdot u)} \quad (\text{A.5})$$

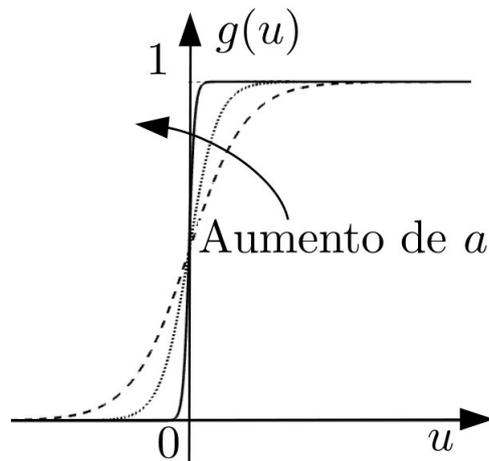
em que a a constante real que representa o parâmetro de inclinação da função logística. Variando o valor de a é possível obter funções logísticas com diferentes

Figura 34 – Função linear por partes



Fonte: Produzido pelo autor

Figura 35 – Função logística



Fonte: Produzido pelo autor

inclinações. Com o aumento do valor de a ($a \rightarrow \infty$) o formato da função logística vai se aproximando do formato da função de Heaviside. Esse comportamento é ilustrado na Figura 35.

Em contraposição à função de Heaviside, a função logística é totalmente diferenciável para todos os seus valores de entrada.

b) Função tangente hiperbólica

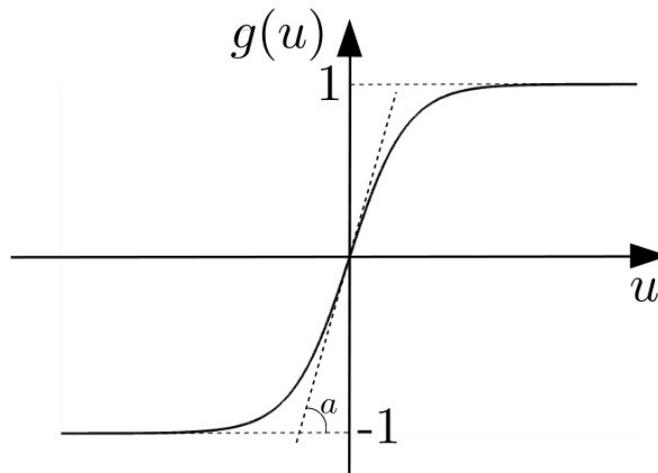
A saída da função tangente hiperbólica, diferente da logística, pode assumir qualquer

valor entre -1 e 1 . Matematicamente essa função é definida por

$$g(u) = \frac{1 - \exp(-a \cdot u)}{1 + \exp(-a \cdot u)} \quad (\text{A.6})$$

A constante a está associada ao nível de inclinação da função tangente hiperbólica em relação a origem. A ilustração dessa função pode ser vista na Figura 36.

Figura 36 – Função tangente hiperbólica



Fonte: Produzido pelo autor

Assim como acontece para a função logística, quanto maior for o valor do parâmetro a , mais inclinado será o gráfico da função tangente hiperbólica.

As funções logística e tangente hiperbólica fazem parte da família de funções chamadas de sigmoidais.

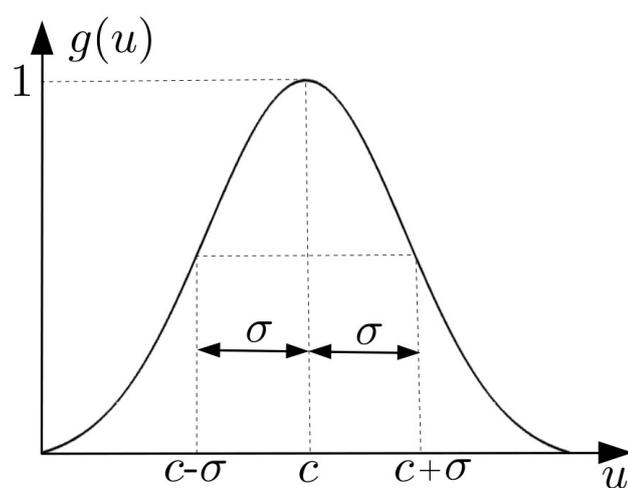
c) Função gaussiana

A função de ativação gaussiana é a função de ativação mais comumente utilizada em redes de função de base radial. A saída do neurônio, $g(u)$, produz as mesmas quantidades para valores do campo local induzido, u , que estejam a uma mesma distância do seu centro, c . Dessa forma, a curva é simétrica em relação ao centro. A função gaussiana é definida por

$$g(u) = \exp\left(-\frac{1}{2\sigma^2} \|u - c\|^2\right) \quad (\text{A.7})$$

em que u é o valor do campo local induzido, o parâmetro c define o centro da função gaussiana e σ indica o quão dispersa está a curva em relação ao seu centro. Graficamente esta função é ilustrada na Figura 37.

Figura 37 – Função gaussiana



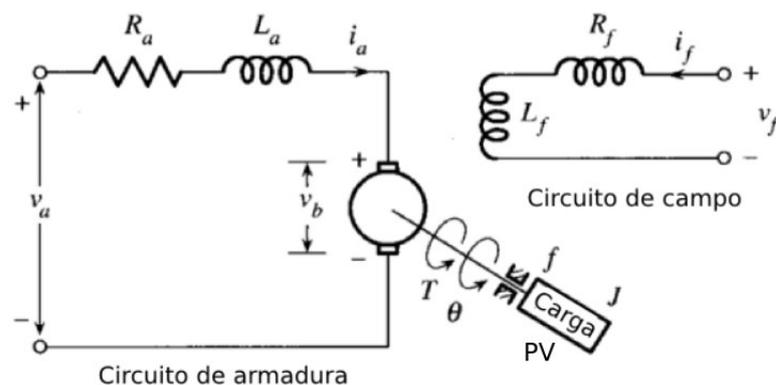
Fonte: Produzido pelo autor

Atuadores Eletromecânicos

Dentro de uma malha de controle, existem elementos que produzem movimento atendendo a comandos que podem ser do tipo elétrico. Um exemplo de atuador muito utilizado em sistemas de controle é o motor de corrente contínua. Este mecanismo transforma a energia de entrada (energia elétrica) em energia cinética.

Na Seção 5.1, foi apresentado o conjunto atuador-carga que é um motor de corrente contínua com a armadura constituída de íma permanente. Neste apêndice, é apresentada a modelagem matemática do conjunto atuador-carga, representado na Figura 38, no sentido de se obter a função de transferência que relacione a saída e a entrada do sistema.

Figura 38 – Modelo do conjunto atuador-carga



Fonte: Adaptado de Chen (2015)

Como a armadura do atuador é constituída de ímã permanente o circuito de campo é considerado constante e a tensão de entrada é aplicada no circuito de armadura.

Se a corrente de campo, i_f , é constante, o torque aplicado na carga é dado por

$$T(t) = k_t i_a(t) \quad (\text{B.1})$$

em que $k_t = ki_f(t)$ uma constante. Considerando J , como o momento de inércia total da carga, eixo e o rotor do motor; θ , o deslocamento angular da carga; e f , o coeficiente de fricção viscoso do rolamento. Então, tem-se que o torque é expresso por

$$T(t) = J \frac{d^2\theta(t)}{dt^2} + f \frac{d\theta(t)}{dt} \quad (\text{B.2})$$

que descreve a relação entre o torque do motor e o deslocamento angular da carga.

Quando o atuador está movimentando a carga, uma tensão, v_b , contra eletromotriz (fcem) surge no circuito de armadura que, por sua vez, opõe-se à tensão aplicada. A tensão $v_b(t)$, proporcionalmente linear à velocidade angular do eixo do atuador é definida por

$$v_b(t) = k_b \frac{d\theta(t)}{dt} \quad (\text{B.3})$$

Assim, o circuito de armadura da Figura 38, é descrito por

$$R_a i_a(t) + L_a \frac{di_a(t)}{dt} + v_b(t) = v_a(t) \quad (\text{B.4})$$

ou

$$R_a i_a(t) + L_a \frac{di_a(t)}{dt} + k_b \frac{d\theta(t)}{dt} = u(t) \quad (\text{B.5})$$

Usando as Equações (B.1), (B.2) e (B.5), podemos encontrar a função de transferência do atuador. A substituição da Equação (B.1) na Equação (B.2), e a aplicação da transformada de Laplace na Equação (B.2), assumindo as condições iniciais nulas, resulta em

$$k_t I_a(s) = J s^2 \Theta(s) + f s \Theta(s) \quad (\text{B.6})$$

e

$$U(s) = R_a I_a(s) + L_a s I_a(s) + k_b s \Theta(s) \quad (\text{B.7})$$

A eliminação de I_a , das Equações (B.6) e (B.7) resulta em

$$\bar{G}(s) = \frac{\Theta(s)}{U(s)} = \frac{k_t}{s[(Js + f)(R_a + L_a s) + k_t k_b]} \quad (\text{B.8})$$

A Equação (B.8) é a função de transferência de $v_a = u$ para θ .