

Universidade Federal de Minas Gerais

Escola de Engenharia

Programa de Pós-Graduação em Engenharia Elétrica

**CONTROLADORES BASEADOS EM TÉCNICAS
DE INTELIGÊNCIA COMPUTACIONAL:
ANÁLISE, PROJETO E APLICAÇÕES**

Anderson Vasconcelos Pires

Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais, como requisito parcial à obtenção de título de Mestre em Engenharia Elétrica.

Orientadores: **Walmir Matos Caminhas**
Benjamim Rodrigues de Menezes

Maio de 2007

Agradecimentos

Agradeço primeiramente a Deus, fonte de vida e sabedoria, por ter sempre iluminado o meu caminho.

Aos meus pais, pelo amor, incentivo, dedicação e exemplo de vida. Meu irmão, pelo apoio e amizade. Aos familiares e amigos, por compreenderem a minha ausência nos momentos de descontração.

Aos meus orientadores, professor Walmir Matos Caminhas e professor Benjamim Rodrigues Menezes, os quais sou imensamente grato pelo que me ensinaram, além da confiança, paciência e amizade.

A todos os professores, funcionários e colegas do CPDEE que contribuíram direta ou indiretamente para a realização deste trabalho. Em especial a Marlon Rosa de Gouvêa, que contribuiu muito para o meu crescimento técnico, pessoal e profissional.

Ao meu antigo chefe ainda pela ATAN, Marcelo Szuster, por me conceder tempo para poder dedicar ao mestrado. Meu atual chefe na Gerdau Açominas, Mário Hermes de Resende, por apoiar o meu estudo, não só no Mestrado, mas também na graduação.

A todas as pessoas cujos nomes não foram citados, mas que jamais serão esquecidos pelas contribuições.

Resumo

Este trabalho analisa a aplicação de técnicas de inteligência computacional no controle de processos, evidenciando as suas vantagens de facilidade de implementação e melhoria dos índices de desempenho. O On-Line Neurofuzzy Controller (ONFC) é uma das técnicas de controle utilizada neste trabalho, que se apresenta como uma técnica alternativa ao controlador PI. Por ser um controlador recentemente proposto (2005) e não haver ainda estudos analíticos mais detalhados realizados sobre o mesmo, propõe-se apresentar uma contribuição na análise matemática de suas equações. Ainda sobre o ONFC, é apresentado o seu análogo neural e a versão que utiliza a técnica de Controle por Modos Deslizantes para realizar o treinamento deste controlador. Uma outra técnica analisada neste trabalho é o controle dinâmico inverso. São apresentados alguns pontos que dificultam obter o modelo artificial inverso do sistema a ser controlado e o que se pode fazer para facilitar a obtenção de tal modelo. Em virtude de algumas limitações desse controlador e por ser um controlador que não necessita ter um conhecimento profundo do sistema a ser controlado, é proposto um controlador dinâmico inverso aproximado. Nessa proposta, utiliza-se o modelo artificial do sistema para se obter a função inversa a cada iteração. Por utilizar o modelo artificial do sistema, que é muito mais fácil de obter do que o modelo inverso, esse controlador apresenta resultados bem melhores do que o controle dinâmico inverso, principalmente no controle com treinamentos on-line.

Abstract

This work introduces a study about On-Line Neurofuzzy Controller (ONFC) through the mathematical analysis of its equations that allow understand it more deeply. Still on the ONFC, is introduced the neural and sliding modes versions of controller. The inverse dynamic control is boarded also where are presented some points that complicate to get the inverse artificial model and what can be done to facilitate the attainment of such model. Because of some limitations of inverse dynamic control and to be a controller that does not need to have a so great knowledge of system to be controlled, is propose the approached inverse dynamic control. In this proposal, the artificial model of the system is used to get the inverse function. The artificial model of the system is easier to get than inverse model. This allows obtaining better results in approached inverse dynamic control, especially in control using on-line training.

Sumário

Agradecimentos	iii
Resumo	v
Abstract	vii
Sumário	ix
Lista de Figuras	xiii
Lista de Tabelas	xvii
Lista de Símbolos	xix
1 Introdução Geral	1
1.1 Introdução	1
1.2 Estrutura do texto	3
2 Inteligência Computacional Aplicada no Projeto de Controladores e o NFN	5
2.1 Introdução	5
2.2 Inteligência computacional aplicada no projeto de controladores	6
2.2.1 Controladores nebulosos.....	6
2.2.2 Controladores neurais	6
2.2.3 Controladores neurofuzzy.....	7
2.3 Estruturas de controle.....	8
2.3.1 Ajuste de parâmetros de controladores convencionais.....	8
2.3.2 Controle nebuloso	8
2.3.3 Controle preditivo.....	10
2.3.4 Controle por modelo de referência	11
2.3.5 Controle inverso.....	12
2.4 Neo Fuzzy Neuron.....	13
2.4.1 Processo de aprendizagem	16
2.5 Conclusões	17
3 Controladores Baseados no ONFC	19
3.1 Introdução	19
3.1.1 Análise matemática	22
3.1.2 Resultados de simulação	26
3.2 On-Line Neuro Controller.....	28
3.2.1 Resultados de simulação	31
3.3 ONFC em modos deslizantes	33
3.3.1 Convergência	34
3.3.2 Resultados de simulação	37

3.4	Conclusões.....	38
4	Controle Neurofuzzy Baseado na Dinâmica Inversa	41
4.1	Introdução	41
4.2	Controle de velocidade do motor CC	44
4.2.1	Simulações com treinamento off-line.....	47
4.2.1.1	Inserção de ruído na medição de velocidade.....	49
4.2.1.2	Inversão do torque de carga	51
4.2.1.3	Variação da ação de controle limitada em 70 volts com ruído na medição da velocidade	53
4.2.1.4	Variação da ação de controle limitada em 70 volts com inversão do torque de carga.....	53
4.2.1.5	Variação da ação de controle limitada em 17,5 volts com ruído na medição da velocidade	54
4.2.1.6	Variação da ação de controle limitada em 17,5 volts com inversão do torque de carga.....	55
4.2.1.7	Comparativo entre o PI e o controlador utilizando a dinâmica inversa	56
4.2.2	Simulações com treinamento on-line	58
4.3	Conclusões.....	63
5	Controle Dinâmico Inverso Aproximado	65
5.1	Introdução	65
5.2	Controle de velocidade do motor CC	68
5.2.1	Simulações com treinamento off-line.....	69
5.2.1.1	Inserção de ruído na medição de velocidade.....	71
5.2.1.2	Inversão do torque de carga	72
5.2.1.3	Variação da ação de controle limitada em 70 volts com ruído na medição da velocidade	73
5.2.1.4	Variação da ação de controle limitada em 70 volts com inversão do torque de carga.....	74
5.2.1.5	Variação da ação de controle limitada em 17,5 volts com ruído na medição da velocidade	75
5.2.1.6	Variação da ação de controle limitada em 17,5 volts com inversão do torque de carga.....	76
5.2.2	Simulações com treinamento on-line	77
5.2.2.1	Inserção de ruído na medição de velocidade.....	79
5.2.2.2	Inversão do torque de carga	80
5.2.2.3	Variação da ação de controle limitada em 70 volts com ruído na medição da velocidade	81
5.2.2.4	Variação da ação de controle limitada em 70 volts com inversão do torque de carga.....	82
5.2.2.5	Variação da ação de controle limitada em 17,5 volts com ruído na medição da velocidade	83
5.2.2.6	Variação da ação de controle limitada em 17,5 volts com inversão do torque de carga.....	84
5.2.2.7	Interrupção do treinamento da rede com ruído na medição de velocidade.....	85

5.2.2.8	Interrupção do treinamento da rede com inversão do torque de carga.....	86
5.2.2.9	Controle por um longo período com inserção de ruído na medição de velocidade.....	89
5.2.2.10	Controle por um longo período com inversão do torque de carga	93
5.3	Redes lineares e não lineares	98
5.4	Controle de sistemas não lineares	101
5.4.1	1º Sistema não linear.....	101
5.4.2	2º Sistema não linear.....	104
5.5	Conclusões	107
6	Conclusões e Propostas de Continuidade	109
6.1	Conclusões	109
6.2	Propostas de continuidade	112
	Referências Bibliográficas	113
	A Parâmetros do motor CC	119

Lista de Figuras

2.1 – Diagrama de blocos da estrutura básica de um controlador nebuloso.....	9
2.2 – Diagrama de blocos da modelagem de uma planta $G(s)$ utilizando técnicas de inteligência computacional.....	10
2.3 – Diagrama de blocos do controle preditivo utilizando técnicas de inteligência computacional.....	11
2.4 – Diagrama de blocos do controle pelo modelo de referência.....	12
2.5 – Diagrama de blocos do controle inverso.....	12
2.6 – Estrutura do NFN [<i>Caminhas et al. (1998)</i>].....	14
2.7 – Funções de pertinência [<i>Caminhas et al. (1998)</i>].....	15
2.8 – Mecanismo de inferência do NFN [<i>Caminhas et al. (1998)</i>].....	16
3.1 – Diagrama de blocos do controle com o ONFC.....	19
3.2 – Estrutura da rede NFN utilizada pelo ONFC.....	20
3.3 – Diagrama de blocos da estrutura interna do ONFC.....	21
3.4 – Controle da velocidade do motor CC com aplicação e retirada de carga utilizando o ONFC.....	27
3.5 – Comportamento dos pesos do ONFC durante o controle de velocidade do motor.....	27
3.6 – Controle de velocidade do motor CC utilizando o ONFC.....	28
3.7 – Neurônio representativo do ONFC.....	29
3.8 – Controle de velocidade do motor CC utilizando o controlador neural análogo ao ONFC.....	32
3.9 – Comportamento dos pesos da rede neural no controle de velocidade do motor CC.....	32
3.10 – Controle de velocidade do motor CC utilizando o ONFC em modos deslizantes.....	37
4.1 – Diagrama de blocos de controle utilizando dinâmica inversa.....	42
4.2 – Reposta ao degrau para a planta $G(s) = 1/(s+1)$	43
4.3 – Entradas utilizadas na rede NFN com duas partições para modelar a dinâmica inversa do motor CC.....	45
4.4 – Validação da rede NFN com treinamento através dos mínimos quadráticos.....	46
4.5 – Validação da rede NFN com treinamento através do gradiente.....	46
4.6 – Controle de velocidade do motor CC utilizando a dinâmica inversa com aplicação e retirada de carga.....	47
4.7 – Controle de velocidade do motor CC utilizando a dinâmica inversa.....	48
4.8 – Controle de velocidade do motor CC utilizando um PI.....	49
4.9 – Controle de velocidade do motor CC utilizando dinâmica inversa com ruído na medição de velocidade.....	50

4.10 – Controle de velocidade do motor CC com PI com ruído na medição de velocidade.	51
4.11 – Controle de velocidade do motor CC utilizando dinâmica inversa com inversão de torque a cada dois segundos.	52
4.12 – Controle de velocidade do motor CC utilizando PI com inversão de torque a cada dois segundos.	52
4.13 – Controle da velocidade do motor CC com ruído na medição de velocidade e com a variação da ação de controle limitada em 70 volts.	53
4.14 – Controle da velocidade do motor CC com inversão de torque e com a variação da ação de controle limitada em 70 volts.	54
4.15 – Controle da velocidade do motor CC com ruído na medição da velocidade e com a variação da ação de controle limitada em 17,5 volts.	55
4.16 – Controle da velocidade do motor CC com inversão de torque e com a variação da ação de controle limitada em 17,5 volts.	56
4.17 – Controle da velocidade do motor CC com aplicação e retirada de carga.	60
4.18 – Controle da velocidade do motor CC com treinamento on-line.	60
4.19 – Função do ganho de treinamento em função do erro médio com o $A = 5$ e $B = 2$	62
4.20 – Controle da velocidade do motor CC utilizando treinamento híbrido.	63
5.1 – Diagrama de blocos do controlador aproximando a função inversa a cada iteração.	66
5.2 – Gráfico demonstrando a escolha da ação de controle.	67
5.3 – Entradas utilizadas na rede NFN para modelar a velocidade do motor CC.	68
5.4 – Controle da velocidade do motor CC com aplicação e retirada de carga utilizando aproximação da inversa.	70
5.5 – Controle de velocidade do motor CC utilizando a aproximação da dinâmica inversa.	71
5.6 – Controle de velocidade do motor CC utilizando dinâmica inversa com ruído na medição de velocidade.	72
5.7 – Controle de velocidade do motor CC utilizando aproximação da dinâmica inversa com inversão de torque a cada dois segundos.	73
5.8 – Controle da velocidade do motor CC com ruído na medição de velocidade e com a variação da ação de controle limitada em 70 volts.	74
5.9 – Controle da velocidade do motor CC com inversão de torque e com a variação da ação de controle limitada em 70 volts.	75
5.10 – Controle da velocidade do motor CC com ruído na medição da velocidade e com a variação da ação de controle limitada em 17,5 volts.	76
5.11 – Controle da velocidade do motor CC com inversão de torque e com a variação da ação de controle limitada em 17,5 volts.	77
5.12 – Controle da velocidade do motor CC com aplicação e retirada de carga utilizando aproximação da inversa.	78
5.13 – Controle da velocidade do motor CC utilizando aproximação da inversa.	79
5.14 – Controle da velocidade do motor CC utilizando aproximação da inversa com ruído na medição de velocidade.	80
5.15 – Controle da velocidade do motor CC utilizando aproximação da inversa com inversão do torque de dois em dois segundos.	81

5.16 – Controle da velocidade do motor CC com ruído na medição de velocidade e com a variação da ação de controle limitada em 70 volts.....	82
5.17 – Controle da velocidade do motor CC com inversão do torque de carga e com a variação da ação de controle limitada em 70 volts.....	83
5.18 – Controle da velocidade do motor CC com ruído na medição de velocidade e com a variação da ação de controle limitada em 17,5 volts.	84
5.19 – Controle da velocidade do motor CC com inversão do torque de carga e com a variação da ação de controle limitada em 17,5 volts.....	85
5.20 – Controle da velocidade do motor CC com treinamento on-line interrompido em $t = 60$ segundos.	86
5.21 – Controle da velocidade do motor CC com inversão do torque de carga onde o treinamento on-line foi interrompido em $t = 60$ segundos.	87
5.22 – Controle da velocidade do motor CC com inversão do torque de carga onde o treinamento on-line foi interrompido em $t = 62$ segundos.	88
5.23 – Controle da velocidade do motor CC com inversão do torque de carga onde o treinamento on-line foi interrompido em $t = 100$ segundos.	89
5.24 – Controle da velocidade do motor CC com ruído na medição de velocidade onde o controle é feito por um longo período de tempo.....	90
5.25 – Zoom do controle da velocidade do motor CC com ruído na medição de velocidade onde o controle é feito por um longo período de tempo.	91
5.26 – Zoom do controle da velocidade do motor CC com ruído na medição de velocidade onde o controle é feito por um longo período de tempo.	92
5.27 – Controle da velocidade do motor CC com ruído na medição de velocidade onde o controle é feito por um longo período de tempo com limitação máxima de 70 volts.....	93
5.28 – Controle da velocidade do motor CC com inversão do torque de carga onde o controle é feito por um longo período de tempo com limitação máxima de 70 volts.....	94
5.29 – Controle da velocidade do motor CC com inversão do torque de carga onde o controle é feito por um longo período de tempo com limitação máxima de 70 volts e ajuste automático do ganho de treinamento.	95
5.30 – Controle da velocidade do motor com inversão do torque de carga onde o controle é feito por um longo período de tempo e com limitação da referência.	97
5.31 – Controle da velocidade do motor CC com inversão do torque de carga onde o controle é feito por um longo período de tempo com limitação máxima de 17,5 volts.....	98
5.32 – Controle da velocidade do motor CC com inversão do torque de carga onde o controle é feito por um longo período de tempo com limitação máxima de 70 volts.....	100
5.33 – Validação do treinamento do 1º sistema não linear utilizando a rede NFN.	102
5.34 – Controle do sistema não linear utilizando aproximação da inversa com treinamento off-line da rede NFN.....	103
5.35 – Controle do sistema não linear utilizando aproximação da inversa com treinamento on-line da rede NFN.	103
5.36 – Validação do treinamento off-line com a rede NFN.....	105
5.37 – Controle do sistema não linear utilizando aproximação da inversa com a rede NFN treinada off-line.	105

5.38 – Controle do sistema não linear utilizando aproximação da inversa com a rede MLP treinada off-line.	106
--	-----

Lista de Tabelas

3.1 – Parâmetros da rede do controlador ONFC no controle da velocidade do motor CC.	27
3.2 – Parâmetros do controlador neural análogo ao ONFC.	33
3.3 – Parâmetros da rede do controlador ONFC em modos deslizantes utilizado no controle da velocidade do motor CC.	38
3.4 – Comparação entre o ONFC e o ONFC em modos deslizantes no controle da velocidade do motor CC.	38
4.1 – Comparação entre o PI e o controle baseado na Dinâmica Inversa (DI) sem inserção de ruído na medição de velocidade e inversão do torque de carga.	57
4.2 – Comparação entre o PI e o controle baseado na Dinâmica Inversa na condição de inserção de ruído na medição de velocidade.	57
4.3 – Comparação entre o PI e o controle baseado na dinâmica inversa na condição de inversão do torque de carga.	58

Lista de Símbolos

ω	Velocidade do motor
V	Tensão de armadura
I	Corrente do motor
Tl	Conjugado de carga
W	Peso de redes artificiais
$\mu(x_i)$	Grau de pertinência da variável x_i
u	Ação de controle
e	Erro entre a referência e o valor de saída da planta
α	Taxa de aprendizado ou ganho de treinamento

Siglas:

ANFIS	Adaptative Neuro-Fuzzy Inference System
CC	Corrente Contínua
CMD	Controle em Modos Deslizantes
DI	Dinâmica Inversa
EM	Erro Máximo
MLP	Multilayer Perceptrom
NFN	Neo Fuzzy Neuron
ONC	On-Line Neuro Controller
ONFC	On-Line Neurofuzzy Controller
PI	Proporcional Integral
PID	Proporcional Integral Derivativo
SEV	Sistemas de Estrutura Variáveis

Capítulo 1

Introdução Geral

Neste capítulo será feita a introdução geral deste trabalho apresentando as propostas e a estrutura do texto.

1.1 Introdução

Os métodos convencionais de controle automático baseados em controladores do tipo PID [Ogata (2003)] são simplesmente os mais utilizados em malhas de controle, pois são simples, eficientes e têm uma base teórica bem desenvolvida e consolidada. Entretanto, dificilmente encontra-se esse tipo de controlador bem sintonizado nos processos em que são empregados. Um dos motivos é que a sintonia dos parâmetros do controlador não é uma tarefa das mais fáceis. Em muitos casos são sintonizados através de métodos heurísticos, o que geralmente resulta em controladores mal sintonizados.

Outra desvantagem desses controladores é que possuem parâmetros fixos enquanto os sistemas dinâmicos reais apresentam-se, em sua quase totalidade, como sistemas não lineares e com parâmetros variáveis no tempo. Este fato pode implicar na variação do comportamento dinâmico do sistema controlado, podendo inclusive levar o sistema à instabilidade. Estes problemas decorrem da metodologia de projeto baseada em técnicas lineares, que estão associadas a modelos linearizados, válidos apenas para pequenas perturbações em torno de um ponto de operação.

Com o objetivo de superar tais limitações dos controladores com parâmetros fixos, surgiram os controladores do tipo adaptativo [Harris et al. (1981)]. Em uma de suas formas mais

simples, tais controladores são constituídos por compensadores convencionais cujos parâmetros são ajustados a partir da identificação on-line dos parâmetros do sistema que se deseja controlar. No entanto, o processo de identificação on-line dos parâmetros do sistema implica em maior complexidade do sistema de controle.

Dentre as alternativas ao controlador PI com ganhos fixos, destacam-se os controladores baseados em inteligência computacional como, por exemplo, os controladores nebuloso, neural e neurofuzzy [Jang *et al.* (1997), Justino (2004) e Gouvêa (2005)]. Uma das vantagens em se utilizar controladores baseados em inteligência computacional é a maior facilidade de sintonia dos mesmos. Além disso, tem-se maior robustez de desempenho em uma ampla faixa de operação e capacidade de adaptação ao sistema controlado, tanto por perturbações quanto por variações paramétricas.

O controlador proposto por Gouvêa (2005), o On-Line Neurofuzzy Controller (ONFC), é uma alternativa interessante ao PI por ser um controlador simples, de baixo custo computacional, se adapta à dinâmica do processo numa ampla faixa de operação e de fácil sintonia de seus parâmetros. Como os resultados apresentados com a utilização deste controlador são muito promissores [Gouvêa (2005)], este trabalho se propõe a contribuir na análise matemática das equações envolvidas no controle de um processo e no desenvolvimento de novas estruturas de controladores.

Uma das propostas apresentada é um análogo neural do ONFC, o On-Line Neuro Controller (ONC), que é ainda mais simples que o ONFC e apresenta um custo computacional ainda mais baixo. É apresentado também uma outra versão do ONFC que utiliza a teoria de controle conhecida como Sistemas de Estruturas Variáveis (SEV) [Itkis (1976), Utkin (1977)], mais especificamente, Controle por Modos Deslizantes (CMD) [Utkin (1978)] para treinar a rede Neo Fuzzy Neuron (NFN) [Yamakawa *et al.* (1992)] do controlador. Com a utilização da teoria de CMD para realizar o treinamento da rede é possível obter resultados melhores devido à superfície de deslizamento.

Na seqüência do trabalho também é apresentado um estudo sobre o controle baseado na dinâmica inversa [Jang *et al.* (1997)]. O uso deste controlador é interessante, pois não necessita ter um conhecimento profundo do sistema a ser controlado, quando se utiliza a técnica de

inteligência computacional para obter o modelo da dinâmica inversa. Basta, a princípio, utilizar dados históricos do processo para que se obtenha tal modelo e utilizá-lo para controlar um processo. Mas, nem todo processo possui uma função inversa e, quando existe, não é fácil obtê-la, mesmo utilizando técnicas de inteligência computacional. Serão apresentados alguns pontos que dificultam a obtenção do modelo da dinâmica inversa do sistema a ser controlado, o que se pode ser feito a fim de facilitar a obtenção de tal modelo e as características deste controlador.

Em virtude de algumas limitações do controle baseado na dinâmica inversa e por ser um controlador que não necessita ter um conhecimento profundo do sistema a ser controlado, a outra proposta deste trabalho é um controlador baseado no controlador dinâmico inverso. Neste controlador proposto, a dinâmica inversa da planta é aproximada a cada iteração através do próprio modelo da planta. Essa proposta traz como benefício o poder de se adaptar ao processo mais facilmente do que na dinâmica inversa direta, já que o modelo artificial do processo é mais fácil de obter do que o modelo artificial inverso. Ele diminui o esforço de sintonia do controlador, principalmente quando comparado com o PI, já que o mesmo pode ser feito usando treinamentos off-line com dados históricos do processo ou até mesmo utilizando em alguns casos, somente, treinamentos on-line.

Os testes e validações dos estudos realizados neste trabalho serão feitos através de simulações digitais no Matlab® controlando a velocidade de um motor de Corrente Contínua e de dois sistemas não lineares.

1.2 Estrutura do texto

Este trabalho é dividido em seis partes:

Primeiro Capítulo: É feita uma abordagem genérica dos objetivos deste trabalho, mostrando sua relevância e contribuições do assunto tratado;

Segundo Capítulo: É apresentado um breve histórico sobre o uso da inteligência computacional aplicada no projeto de controladores mostrando algumas estruturas de

controle bem difundidas. É apresentada também a rede NFN que é a principal rede artificial explorada neste trabalho no projeto dos controladores;

Terceiro Capítulo: É apresentado o estudo sobre o ONFC através da análise matemática das equações envolvidas no controle de um processo, o análogo neural deste controlador e o ONFC que utiliza a teoria de Controle por Modos Deslizantes para ajustar os pesos da rede NFN do controlador.

Quarto Capítulo: É feita uma introdução sobre o controle neurofuzzy baseado na dinâmica inversa do processo. O controle baseado na dinâmica inversa será aplicado no controle de velocidade de um motor CC. São mostradas as dificuldades em se obter a função inversa da planta em um sistema dinâmico e as características desse controlador. A introdução do controle neurofuzzy baseado na dinâmica inversa é uma base para o desenvolvimento do controlador dinâmico inverso aproximado proposto neste trabalho.

Quinto Capítulo: É apresentada a proposta do controle dinâmico inverso aproximado. Mostram-se os benefícios em se utilizar essa abordagem e pontos que dificultam o controle de um processo. Esse controlador proposto será utilizado para controlar a velocidade de um motor CC e dois sistemas não lineares.

Sexto Capítulo: As principais conclusões deste trabalho são apresentadas, assim como as propostas para trabalhos futuros.

Capítulo 2

Inteligência Computacional Aplicada no Projeto de Controladores e o NFN

Neste capítulo será apresentado um breve histórico sobre o uso da inteligência computacional aplicada no projeto de controladores mostrando algumas estruturas de controle bem difundidas. Será apresentada também a rede NFN, estrutura utilizada neste trabalho nas implementações dos controladores.

2.1 Introdução

A teoria de inteligência computacional vem sendo amplamente aplicada ao longo dos anos no projeto de controladores. Um dos motivos é porque abstrai o projetista, em muitos casos, de saber as técnicas de sintonia de controladores convencionais devido à capacidade de aprendizado e de adaptação dos mesmos. Assim, o esforço humano no desenvolvimento do controlador é diminuído consideravelmente e em consequência o tempo gasto também é bem menor. As metodologias de sintonia de controladores convencionais apresentam, normalmente, resultados pouco satisfatórios, já que, além de serem trabalhosas, os parâmetros do controlador obtidos no projeto, quando utilizados na prática, apresentam resultados insatisfatórios e isso favorece o emprego de métodos empíricos para a sintonia de seus parâmetros.

Outra vantagem em se utilizar controladores baseados em inteligência computacional está ligada ao fato dos mesmos trabalharem naturalmente com sistemas não lineares. Além disso, permitirão o tratamento de incertezas e informações vagas com mais facilidade. Nos

controladores convencionais é necessário utilizar, dependendo do processo, ganhos escalonados em função dos diferentes pontos de operação para garantir um bom desempenho global do processo ou, até mesmo, para evitar problemas de instabilidade. Isso implica em uma maior complexidade na sintonia dos ganhos escalonados do sistema.

2.2 Inteligência computacional aplicada no projeto de controladores

Os controladores nebuloso, neural e neurofuzzy têm-se destacado ao longo dos anos como alternativas interessantes aos controladores convencionais. A seguir será feita uma breve descrição sobre esses controladores.

2.2.1 Controladores nebulosos

A lógica nebulosa tem sido vastamente aplicada em controle desde que a teoria dos conjuntos nebulosos foi introduzida por *Zadeh (1965)*. *Mamdani (1976)* foi um dos pioneiros em utilizar lógica nebulosa em controle e demonstrou as vantagens em utilizar a linguagem natural para desenvolver controladores. Desde então, a lógica nebulosa vem sendo aplicada com bastante sucesso em uma vasta gama de aplicações [*Takagi & Sugeno (1983)*, *Sugeno (1985)*, *Lee (1990)*].

A estrutura básica desse tipo de controle será apresentada na seção 2.3.2.

2.2.2 Controladores neurais

Com o advento do algoritmo de treinamento *backpropagation* [*Rumelhart et al. (1986)*] começou-se definitivamente a utilização das RNA em controle de processos. Algumas características atraem a utilização dessas redes na área de controle, tais como: aproximador universal [*Hornik et al. (1989)*], estrutura interna altamente paralela, atuar como uma “caixa preta” e permite trabalhar com múltiplas entradas e múltiplas saídas. A característica da RNA de atuar como

uma caixa preta pode não ser interessante onde se deseja interpretar a sua estrutura física em termos de significado lingüístico.

Ao longo dos anos, diversos trabalhos foram desenvolvidos utilizando redes neurais. *Nguyen et al. (1990)* mostram a aplicação de redes neurais para controlar sistemas não lineares. *Liguni et al. (1991)* utilizam o controlador neural para auxiliar no controle dos controladores convencionais para compensar as não linearidades do sistema.

2.2.3 Controladores neurofuzzy

As redes neurofuzzy vêm atraindo bastante atenção dos pesquisadores ao longo dos últimos anos porque a estrutura dessa rede funde as melhores características da lógica nebulosa e da rede neural em uma única rede. A fusão dessas duas estruturas permite entender as expressões de conhecimento humano usado em termos da lógica nebulosa através das regras *Se-Então*. Isso reduz a dificuldade em descrever o treinamento da rede neural na qual é tratada como uma caixa preta. O outro benefício da fusão dessas duas estruturas é permitir utilizar a capacidade de aprendizagem da rede neural.

Entre as estruturas de redes neurofuzzy destacam-se a ANFIS (Adaptive Neuro-Fuzzy Inference System) [*Jang et al. (1997)*] e o NFN (Neo Fuzzy Neuron) [*Yamakawa et al. (1992)*]. A rede ANFIS usa o sistema de inferência de Sugeno. O aprendizado dessa rede se dá através do método do gradiente descendente (*backpropagation*) e/ou dos mínimos quadráticos. Essa rede apresenta uma estrutura bem complexa quando se utiliza todas as combinações possíveis de suas regras. Isso faz com que o treinamento dessa rede cresça exponencialmente com o número de entradas da rede. A rede NFN, ao contrário da rede ANFIS, é uma rede bem mais simples, de menor custo computacional, que faz com que ela seja atraente para aplicações onde é necessário que o esforço computacional seja pequeno. Ela também pode utilizar em seu aprendizado o algoritmo do gradiente descende ou dos mínimos quadráticos.

Recentemente, *Gomêa (2005)* desenvolveu um controlador baseado na rede NFN, identificado como ONFC (On-line Neurofuzzy Controller), que será explorado nesse trabalho. Esse controlador apresenta uma estrutura bem simples, de baixo custo

computacional, se adapta à dinâmica do processo numa ampla faixa de operação e de fácil sintonia de seus parâmetros. Ele foi aplicado no controle de um motor de indução, uma planta nada simples de se controlar. Este controlador foi aplicado com bastante sucesso no controle de velocidade um motor de indução em uma bancada experimental, no controle de pH e temperatura de torres de resfriamento de uma planta Carboquímica na Gerdau Açominas [Gouvêa et al. (2005 a), Carvalho et al. (2006)].

2.3 Estruturas de controle

A inteligência computacional, através das redes artificiais, está presente em diversas estruturas de controle, seja atuando diretamente como um controlador ou indiretamente como, por exemplo, no ajuste dos parâmetros de controladores convencionais. Serão apresentadas, a seguir, algumas das estruturas mais difundidas.

2.3.1 Ajuste de parâmetros de controladores convencionais

Nesse tipo de estrutura, basicamente a inteligência computacional é utilizada para modelar a planta em tempo real e esse modelo é usado para se ajustar os parâmetros dos controladores convencionais. *Ligumi et al. (1991)* utiliza controlador neural para auxiliar controladores convencionais para compensar não linearidades do sistema.

2.3.2 Controle nebuloso

O controle nebuloso [Lee (1990)] é baseado em sistema de inferência nebulosa. A Figura 2.1 exibe o diagrama de blocos da estrutura básica de um controlador nebuloso. Essa estrutura compreende de quatro principais componentes: um módulo de Fuzzificação, um sistema de Conhecimento (base de regras), um sistema de Decisão (mecanismo de inferência) e um módulo de Desfuzzificação.

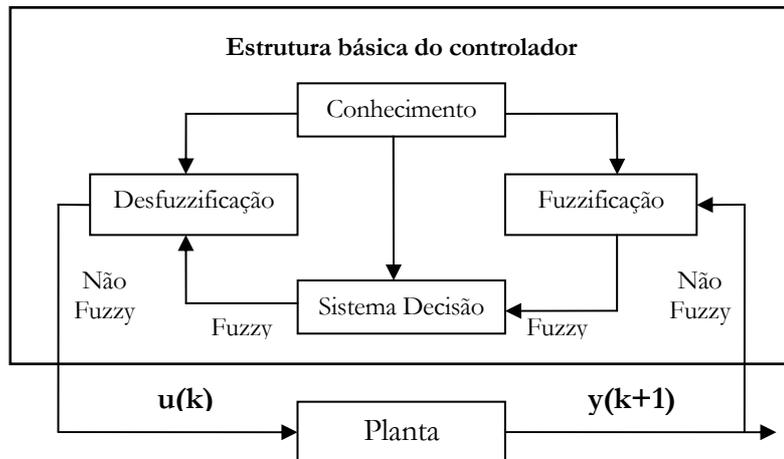


Figura 2.1 – Diagrama de blocos da estrutura básica de um controlador nebuloso.

As regras nebulosas são de fácil entendimento porque utiliza a linguagem natural do pensamento humano. Um exemplo simples de controle de nível de um tanque é mostrado abaixo utilizando regras nebulosas:

1. Se (Nível é OK) então (Abertura da Válvula é Não Alterar).
2. Se (Nível é baixo) então (Abertura da Válvula é Abrir Rápido).
3. Se (Nível é alto) então (Abertura da Válvula é Fechar Rápido).

Através das três regras descritas acima é possível controlar o nível de tanque atuando em sua válvula. Pode-se melhorar o desempenho do processo aplicando as duas regras abaixo. Essas duas regras possuem a característica de um controlador derivativo.

4. Se (Nível é bom) e (taxa é negativa), então (Abertura da Válvula é Fechar Devagar).
5. Se (Nível é bom) e (taxa é positiva), então (Abertura da Válvula é Abrir Devagar).

Através do exemplo acima das regras nebulosas aplicado ao controle, pode ser visto que o controle nebuloso simplifica bastante o projeto do controlador. Em contrapartida, o ajuste dessas regras pode ser uma tarefa mais complicada, mas que pode se tornar mais fácil com o

advento da utilização de redes neurofuzzy, que possuem uma ótima capacidade de aprendizado e adaptação.

2.3.3 Controle preditivo

No controle preditivo [Soloway & Haley (1996)] a rede é treinada de forma a modelar a planta, veja Figura 2.2. Esse modelo é usado pelo controlador que utiliza um método de otimização numérica para determinar a ação de controle, $\mathbf{u}(\mathbf{k})$, de forma a minimizar uma função de custo J .

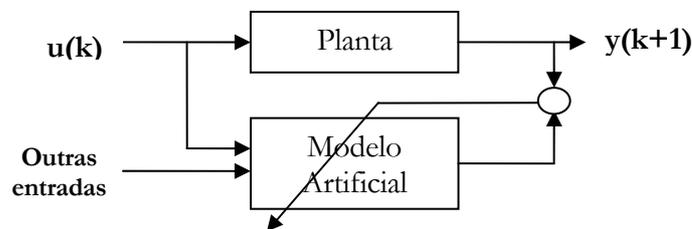


Figura 2.2 – Diagrama de blocos da modelagem de uma planta $G(s)$ utilizando técnicas de inteligência computacional.

A equação (2.1) mostra uma função de custo apropriada para se utilizar no algoritmo de otimização desse controlador. Onde N representa o número de passos futuros, β a contribuição da variação da ação de controle na função de custo e \mathbf{r} a referência.

$$J = \sum_{i=1}^N (r(k+i) - y'(k+i))^2 + \beta \sum_{i=1}^N (u'(k+i-1) - u'(k+i-2))^2 \quad (2.1)$$

A Figura 2.3 mostra o diagrama de blocos do controle preditivo. O controlador consiste de um modelo da planta $G(s)$ e um bloco de otimização. O bloco de otimização determina valores de $\mathbf{u}'(\mathbf{k})$ para encontrar através do modelo $\mathbf{y}'(\mathbf{k}+1)$ que é utilizado na minimização da função de custo J . Após minimizar a função de custo J , o $\mathbf{u}(\mathbf{k})$ ótimo é aplicado na planta.

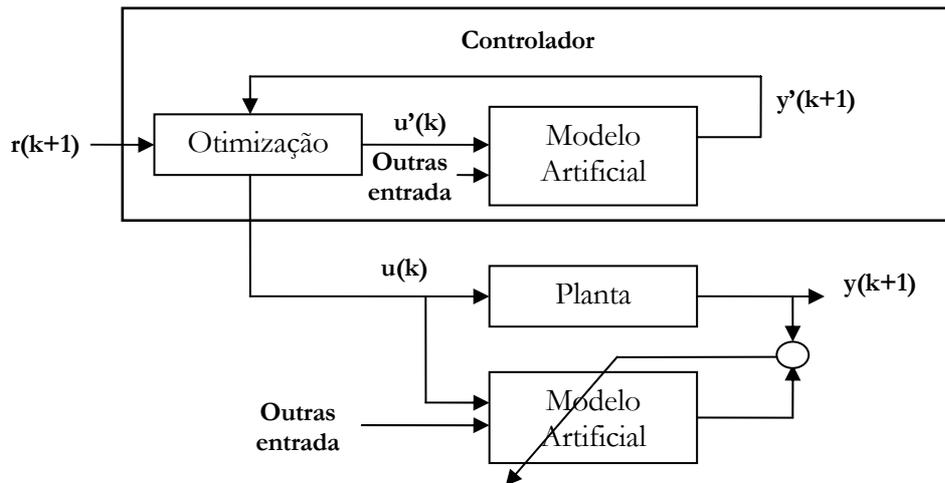


Figura 2.3 – Diagrama de blocos do controle preditivo utilizando técnicas de inteligência computacional.

Uma desvantagem da utilização dessa estrutura de controle é que não existe uma realimentação direta do erro entre a referência e a saída da planta. Para suprir essa deficiência basta que seja realizado o treinamento on-line do modelo da planta. O sucesso desse tipo de controle depende muito do modelo, um modelo ruim pode implicar em um processo mal controlado.

2.3.4 Controle por modelo de referência

A Figura 2.4 mostra o diagrama de blocos do controle por modelo de referência. A arquitetura do controle pelo modelo de referência usa duas redes: uma para o controlador e outra para modelar a planta. O modelo é identificado primeiramente, e então é usado para propagar o erro no treinamento da rede controladora. O treinamento da rede controladora é feito minimizando uma função de custo J .

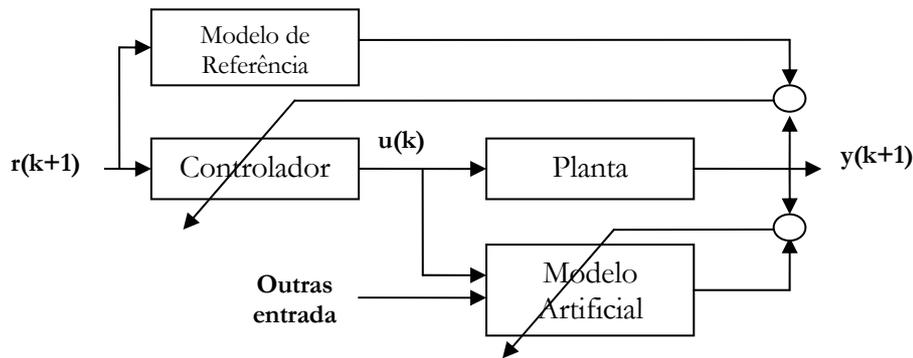


Figura 2.4 – Diagrama de blocos do controle pelo modelo de referência.

O custo de treinar duas redes é uma desvantagem dessa arquitetura. Após completar o treinamento da rede controladora, é difícil mudar o desempenho do mesmo porque é necessário re-treinar a rede controladora. Em geral, o treinamento do controlador é mais difícil do que o identificador que é outra desvantagem.

2.3.5 Controle inverso

A estrutura do controle inverso possui uma rede com o objetivo de mapear a entrada da planta em função da saída, veja o diagrama de blocos do controle inverso mostrado na Figura 2.5. Esse mesmo modelo inverso é utilizado como o controlador do processo colocando-o em série com a planta. Desta forma, a dinâmica do controlador cancela a dinâmica da planta. Assim, teoricamente tudo que se colocar como referência será à saída da planta.

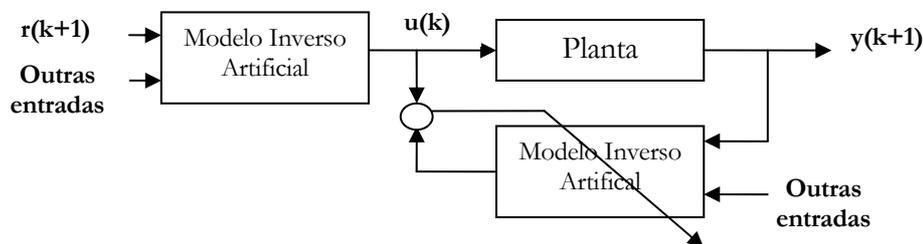


Figura 2.5 – Diagrama de blocos do controle inverso.

Essa estrutura é interessante, pois utilizando dados históricos do processo pode-se realizar facilmente o treinamento do modelo inverso. Isso faz com que o projeto do controlador seja realizado rapidamente sem muito esforço humano. Uma vantagem dessa estrutura é que pode ser adaptativa já que é possível realizar o treinamento on-line do modelo. Em contrapartida, não existe uma realimentação direta do erro entre a referência e a saída da planta.

O problema desse tipo de controle é que a maioria dos sistemas dinâmicos não possui um modelo inverso para todo domínio do processo. Em virtude da dificuldade em se obter a dinâmica inversa para todo domínio do processo, algumas estruturas complementares auxiliam esse controlador. *Lee & Ko (2002)* utilizam a dinâmica inversa juntamente com um sistema de redução do erro do modelo. A dinâmica inversa atua como um controlador *feedforward* e o sistema de redução do erro atua como um controlador fechando a malha. *Chen et al. (2003)* propõem um controle adaptativo dinâmico inverso. A estrutura proposta para esse controlador é bem complexa, pois é composta de um controlador convencional e um sistema nebuloso. O parâmetro do algoritmo de treinamento da rede artificial é ajustado por lógica nebulosa.

O controle inverso será utilizado no Capítulo 4 para controlar a velocidade de um motor CC mostrando as dificuldades em se utilizar essa topologia de controle.

2.4 Neo Fuzzy Neuron

A estrutura do NFN é mostrada na Figura 2.6. As características de cada sinapse são representadas por uma função não-linear $f_i(x_i)$, determinada a partir de inferência nebulosa e defuzzificação. A recodificação dos sinais sinápticos deixa de ser caracterizada por uma função sigmoideal, como nas redes convencionais, passando a ser obtida simplesmente por uma soma algébrica [*Nunes et al. (1996)*]. Isto faz com que a função erro do NFN seja quadrática e convexa com relação aos pesos que são ajustados durante o seu treinamento; portanto, o mínimo local é mínimo global [*Caminhas et al. (1998)*].

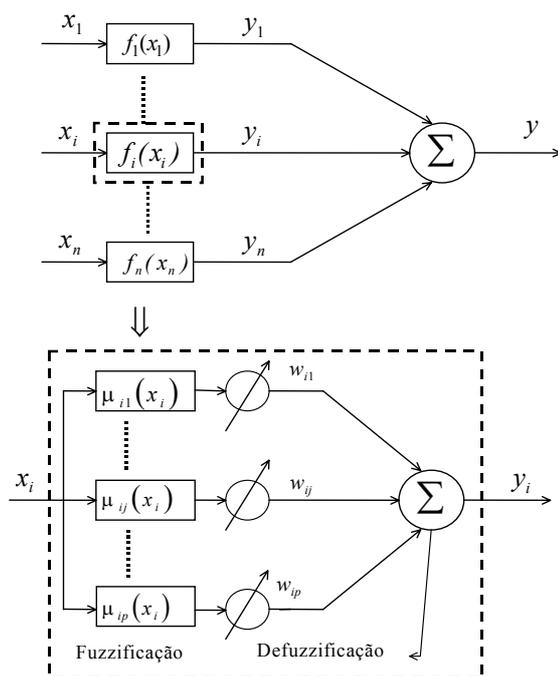


Figura 2.6 – Estrutura do NFN [Caminhas et al. (1998)].

A saída do NFN pode ser representada pela seguinte equação:

$$y = \sum_{i=1}^n y_i = \sum_{i=1}^n f_i(x_i) \quad (2.2)$$

Os valores y_i são determinados a partir de um conjunto de regras nebulosas do tipo Se-Então. Considerando que o universo de discurso da entrada x_i é dividido em p partições nebulosas, descritas por funções de pertinência mostradas na Figura 2.7, tem-se as seguintes p regras:

Se x_i é A_{i1} então y_i é w_{i1}
 Se x_i é A_{i2} então y_i é w_{i2}
 Se x_i é A_{ip} então y_i é w_{ip}

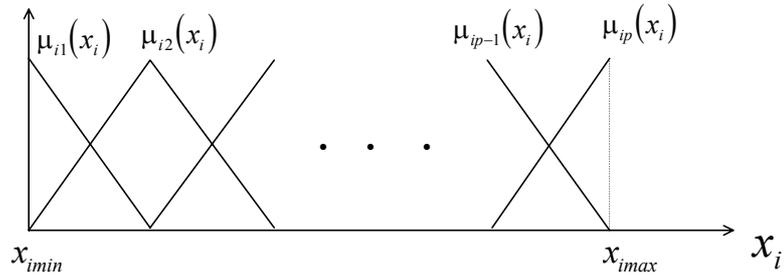


Figura 2.7 – Funções de pertinência [Caminhas et al. (1998)].

Para este conjunto de regras, as variáveis x_i e y_i são determinísticas e Λ_{ij} são conjuntos nebulosos, definidos por $\mu_{ij}(x_i)$. Isto exige as etapas de fuzzificação para x_i e defuzzificação para determinar os valores de y_i . Considerando funções de pertinências complementares, Figura 2.7, os valores das funções de pertinências são diferentes de zero para, no máximo, duas funções vizinhas. Isto equivale a dizer que, para cada valor da variável de entrada x_i , no máximo duas das p regras são ativadas, sendo indexadas por k_i e k_{i+1} . Com isto, utilizando o esquema de inferência max-min, mostrado na Figura 2.8, e o centro de gravidade para efetuar a defuzzificação, os valores de $f_i(x_i)$ podem ser determinados da seguinte forma:

$$f_i(x_i) = \frac{\sum_{j=1}^p \mu_{ij}(x_i) w_{ij}}{\sum_{j=1}^p \mu_{ij}(x_i)} = \frac{\mu_{ik_i}(x_i) w_{ik_i} + \mu_{ik_{i+1}}(x_i) w_{ik_{i+1}}}{\mu_{ik_i}(x_i) + \mu_{ik_{i+1}}(x_i)} \quad (2.3)$$

Utilizando funções de pertinência complementares a soma de duas funções sucessivas é igual a 1. Portanto, tem-se a seguinte expressão:

$$f_i(x_i) = \mu_{ik_i}(x_i) \cdot w_{ik_i} + \mu_{ik_{i+1}}(x_i) w_{ik_{i+1}} \quad (2.4)$$

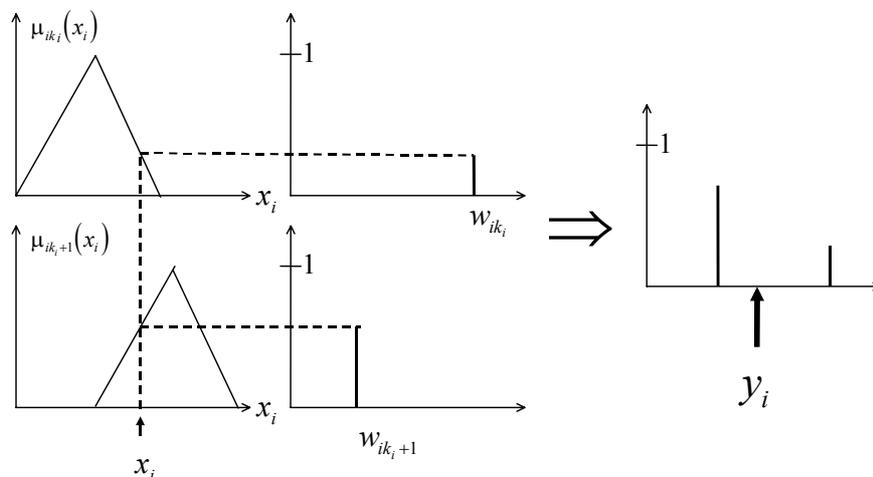


Figura 2.8 – Mecanismo de inferência do NFN [Caminbas et al. (1998)].

Como visto na Figura 2.8, somente ramos ativados são relevantes para o processamento da rede. Assim, só um ou dois pesos correspondentes aos ramos ativados são ajustados durante o treinamento do NFN. Este mecanismo sugere que um NFN deve exibir período de treinamento muito menor do que redes neurais convencionais [Nunes et al. (1996)].

2.4.1 Processo de aprendizagem

O processo de aprendizagem da rede pode ser feito de várias maneiras, entre elas destaca-se a utilização do método do gradiente e dos mínimos quadráticos. Será apresentada a regra de atualização dos pesos através do método do gradiente.

A função de custo J a ser minimizada utilizando o método do gradiente é o erro em relação à saída desejável que é descrita abaixo:

$$J = \frac{1}{2}(y_d - y)^2 \quad (2.5)$$

A regra de ajustes dos pesos segue abaixo:

$$w_{ik_i}^{j+1} = w_{ik_i}^j - \alpha^j \cdot \frac{\partial J(w_{ik_i}^j)}{\partial w_{ik_i}^j} \quad (2.6)$$

Derivando a função de custo em relação aos pesos e substituindo na equação (2.6), obtém-se a seguinte regra de ajustes dos pesos:

$$w_{ik_i}^{j+1} = w_{ik_i}^j + \alpha^j \cdot (y_d - y) \cdot \mu_{ik_i}(x_i) \quad (2.7)$$

O tamanho do passo ou ganho de treinamento, o termo α , que no caso de treinamento de redes neurais ou *neurofuzzy* é chamado de taxa de aprendizado, pode ser determinado de diversas maneiras. *Caminhas et al. (1998)* apresentam um método de cálculo desse parâmetro que faz com que se tenha erro nulo com somente um passo de treinamento.

2.5 Conclusões

Neste capítulo foi apresentado um breve histórico sobre o uso da inteligência computacional aplicada no projeto de controladores mostrando algumas estruturas de controle bem difundidas na literatura e a rede NFN que é a principal rede artificial explorada neste trabalho.

No breve histórico sobre o uso da inteligência computacional aplicada no projeto de controladores foi abordado o uso da lógica nebulosa e das redes neural e neurofuzzy nas estruturas de controle. Foi feita uma pequena introdução sobre as seguintes estruturas de controle de processos bem difundidas: controle ajustando parâmetros de controladores convencionais, nebulosos, preditivo, por modelo de referência e inverso.

Foi feita uma breve introdução sobre a rede NFN que é a rede artificial utilizada no desenvolvimento do controlador ONFC e que será abordado no capítulo seguinte. Essa rede será utilizada também no controle dinâmico inverso e controle dinâmico inverso aproximado no Capítulo 4 e Capítulo 5 respectivamente.

Capítulo 3

Controladores Baseados no ONFC

Neste capítulo será apresentada uma breve introdução do controlador neurofuzzy proposto por *Gouvêa (2005)*, o ONFC, e um estudo sobre este controlador através da análise matemática das equações envolvidas no controle de um processo. Será apresentado também o análogo neural deste controlador e o ONFC em modos deslizantes.

3.1 Introdução

A rede Neo Fuzzy Neuron (NFN) [*Yamakawa et al. (1992)*], veja seção 2.4, é uma rede bem simples que agrega as boas características dos sistemas nebulosos e das redes neurais. De lá deriva o On-Line Neurofuzzy Controller (ONFC) [*Gouvêa (2005)*], um controlador simples, de baixo custo computacional, se adapta à dinâmica do processo numa ampla faixa de operação e de fácil sintonia de seus parâmetros.

A Figura 3.1 exibe o diagrama de blocos do controle de um processo utilizando o ONFC. Pode-se observar que a estrutura de controle do processo com o ONFC é bastante similar a de um controlador convencional.

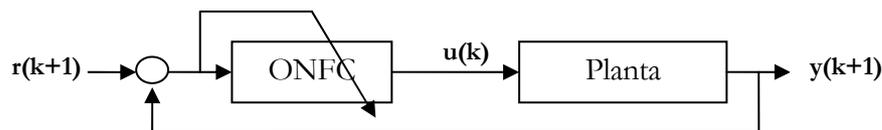


Figura 3.1 – Diagrama de blocos do controle com o ONFC.

O ONFC utiliza a rede NFN de menor estrutura possível (duas partições), veja Figura 3.2, e isso faz com que ela seja bem simples, principalmente quando comparada com outras estruturas neurais e neurofuzzy. As duas funções de pertinência são dadas pelas equações (3.1) e (3.2) onde e é o erro e é obtido pela diferença entre a referência e a saída da planta, veja equação (3.3).

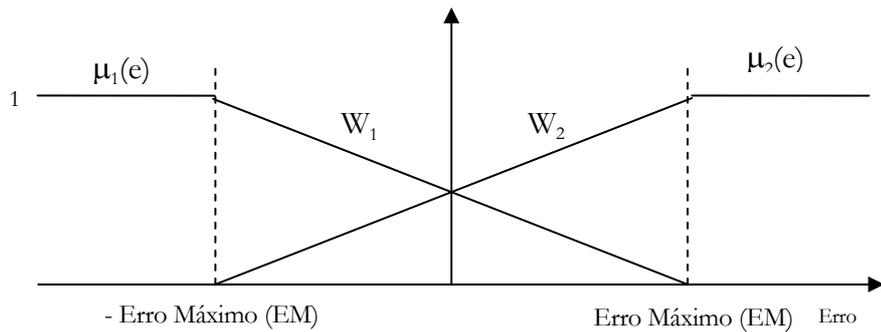


Figura 3.2 – Estrutura da rede NFN utilizada pelo ONFC.

$$\mu_1(e) = \begin{cases} 1 & e \leq -EM \\ \frac{EM - e}{2EM} & -EM < e < EM \\ 0 & e \geq EM \end{cases} \quad (3.1)$$

$$\mu_2(e) = \begin{cases} 0 & e \leq -EM \\ \frac{e + EM}{2EM} & -EM < e < EM \\ 1 & e \geq EM \end{cases} \quad (3.2)$$

$$e = r - y \quad (3.3)$$

O erro é utilizado como entrada e como direção no algoritmo do gradiente descendente para minimizar a função de custo J descrita na equação (3.4). Essa função de custo é o erro quadrático entre a referência e a saída da planta.

$$J = \frac{1}{2}(r - y)^2 = \frac{1}{2}e^2 \quad (3.4)$$

A Figura 3.3 exibe o diagrama de blocos da estrutura interna do ONFC. Pode-se observar que o erro é utilizado tanto no ajuste dos pesos quanto no cálculo do valor das funções de pertinência que é multiplicada pelos pesos de forma a produzir a ação de controle.

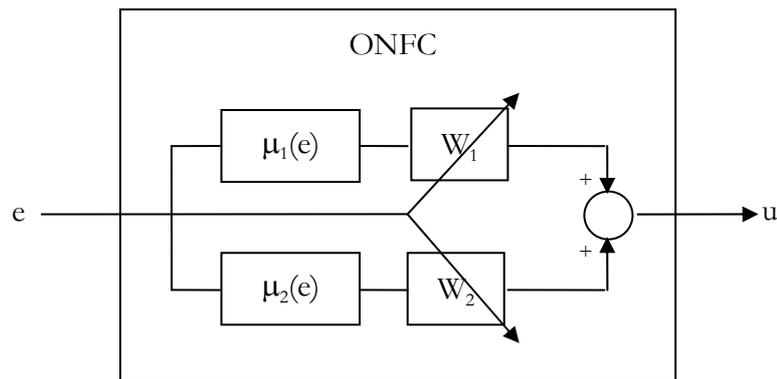


Figura 3.3 – Diagrama de blocos da estrutura interna do ONFC.

A ação de controle é calculada através da equação (3.5).

$$u = \mu_1(e) \cdot W_1 + \mu_2(e) \cdot W_2 \quad (3.5)$$

Como descrito anteriormente, o ajuste dos pesos é feito pelo método do gradiente descendente que minimiza a função de custo J , erro quadrático entre a referência e a saída da planta. A equação (3.6) descreve como deve ser feito o ajuste dos pesos minimizando tal função onde o ganho de treinamento engloba a derivada da saída da planta em relação a sua entrada.

$$W_{i(k)} = W_{i(k-1)} + \alpha_k \cdot e_k \cdot \mu_i(e_k) \quad (3.6)$$

Como pode ser visto, as expressões matemáticas envolvidas são bem simples. Além do mais, poucos são os parâmetros a serem ajustados em seu projeto: ganho de treinamento (α) e os limites das funções de pertinência. Isso tudo facilita não só o projeto do controlador, mas também o seu desenvolvimento e entendimento.

Algumas considerações práticas são feitas por *Gowêa (2005)* sobre o controlador. Por nunca se conseguir, em uma situação real, erro nulo no controle do processo, os pesos do controlador são sempre ajustados e os consecutivos ajustes podem levá-los a terem valores altos ou até mesmo divergirem quando o controle é realizado por um longo período de tempo. Assim, o sistema pode chegar a ficar instável.

A fim de solucionar esse problema, *Gowêa (2005)* utiliza duas estratégias simples: *Bloqueio por Zona Morta* e *Correção por Limite*. Basicamente no *Bloqueio por Zona Morta*, os pesos da rede somente são ajustados quando o erro for maior que o limite estabelecido pelo projetista. Na *Correção por Limite*, os pesos da rede são limitados em um valor a ser definido pelo projetista e recalculados sem alterar a condição de operação do sistema.

Um interessante comentário que pode ser feito sobre o ONFC é que o mesmo pode ser entendido como um caso particular do Sugeno de ordem zero, conforme discutido por *Gowêa (2005)*, onde somente se utiliza duas regras nebulosas para realizar o controle do processo. Essas regras são listadas abaixo.

1. Se (Erro é negativo) então (Ação de controle é W_1)
2. Se (Erro é positivo) então (Ação de controle é W_2)

Devido à simplicidade do ONFC, principalmente quanto aos cálculos envolvidos, esse controlador apresenta um baixo custo computacional o que faz com que o mesmo seja indicado para atuar em sistemas que necessitam realizar o controle em um curto período de tempo.

3.1.1 Análise matemática

A fim de se entender melhor o ONFC, será feita uma análise matemática do controlador através do desenvolvimento matemático de suas equações. Nesse desenvolvimento será considerado que o erro não ultrapassa os limites do universo de discurso das funções de

pertinência, veja Figura 3.2. Essa consideração é bem realista, já que normalmente projeta-se o valor desse limite de forma que a entrada da rede não o ultrapasse.

Algumas situações que normalmente fazem com que o erro aumente e possa fazer com que os limites do universo de discurso sejam atingidos é quando se aplica degrau como referência ou quando se têm grandes perturbações de carga no processo. O degrau como referência normalmente se dá no início do controle do processo e em alguns processos deve ser evitado. As perturbações de carga normalmente são previsíveis quanto a sua grandeza o que possibilita projetar os limites do universo de discurso do controlador de forma apropriada. Por essas situações não ocorrem a todo momento no processo, não há prejuízo em se desprazá-las ao se realizar o estudo do controlador.

A equação (3.7) apresenta o cálculo da ação de controle para um instante de tempo k qualquer.

$$u_k = \mu_1(e_k) \cdot W_{1k} + \mu_2(e_k) \cdot W_{2k} \quad (3.7)$$

As equações (3.8) e (3.9) apresentam as funções de pertinência quando o erro está compreendido entre o limite inferior e superior, veja Figura 3.2. EM representa o Erro Máximo do universo de discurso da função de pertinência e e o erro.

$$\mu_1(e) = \frac{EM - e}{2EM} \quad (3.8)$$

$$\mu_2(e) = \frac{e + EM}{2EM} \quad (3.9)$$

Considerando que os limites da função de pertinência não são atingidos, pode-se substituir (3.8) e (3.9) em (3.7) onde se obtém a equação (3.10).

$$u_k = \frac{EM - e_k}{2EM} \cdot W_{1k} + \frac{e_k + EM}{2EM} \cdot W_{2k} \quad (3.10)$$

Manipulando a equação (3.10) tem-se a equação (3.11). Observa-se que a ação de controle apresenta um termo proporcional ao erro e um termo de polarização.

$$u_k = \frac{W_{2k} - W_{1k}}{2EM} \cdot e_k + \frac{W_{1k} + W_{2k}}{2} \quad (3.11)$$

A regra de atualização dos pesos segue a equação (3.12). Considerando que os pesos da rede são iniciados com valor nulo, pode-se reescrever a equação (3.12) na forma da equação (3.13).

$$W_{1,2(k)} = W_{1,2(k-1)} + \alpha_k \cdot e_k \cdot \mu_{1,2}(e_k) \quad (3.12)$$

$$W_{1,2(k)} = \sum_{i=1}^k \alpha_i \cdot e_i \cdot \mu_{1,2}(e_i) \quad (3.13)$$

Substituindo a equação (3.13) em (3.11) e fazendo algumas manipulações algébricas, tem-se (3.14).

$$u_k = \frac{\sum_{i=1}^k \alpha_i \cdot e_i \cdot (\mu_2(e_i) - \mu_1(e_i))}{2EM} \cdot e_k + \frac{\sum_{i=1}^k \alpha_i \cdot e_i \cdot (\mu_1(e_i) + \mu_2(e_i))}{2} \quad (3.14)$$

A diferença das duas funções de pertinência é expressa na equação (3.15).

$$\mu_2(e) - \mu_1(e) = \frac{e + EM}{2EM} - \frac{EM - e}{2EM} = \frac{e}{EM} \quad (3.15)$$

Substituindo (3.15) em (3.14) e sabendo que as funções de pertinência são complementares, a soma das duas funções resulta na unidade, chega-se a equação (3.16). Nota-se que a ação de controle é da forma de um PI incremental adaptativo (devido ao ganho de treinamento (α) variante durante o tempo). Assim, esse controlador não possui a capacidade de predição, já que essa característica é obtida através da ação derivativa de um controlador convencional. Uma contribuição que poderia ser feita neste controlador é acrescentar essa característica. A princípio, basta utilizar a derivada do erro como entrada neste controlador de forma a poder sintetizar a ação derivativa de um PID.

$$u_k = \frac{\sum_{i=1}^k \alpha_i \cdot e_i^2}{2EM^2} \cdot e_k + \frac{\sum_{i=1}^k \alpha_i \cdot e_i}{2} \quad (3.16)$$

Analisando o caso onde se tem o ganho de treinamento constante, tem-se a equação (3.17). Para essa situação tem-se o KP e o KI dados pelas equações (3.18) e (3.19), respectivamente. A constante proporcional pode ser entendida como uma constante adaptativa que começa com valor nulo e vai aumentando em função do erro quadrático. Isso é um grande benefício do controlador que faz com que o seu projeto seja realizado mais facilmente. Em contrapartida, por não se conseguir erro nulo em uma situação real, essa constante pode atingir um valor que desestabilize o processo. Assim, nota-se porque *Gowêa (2005)* teve que sugerir o uso das estratégias de *bloqueio por zona morta* e *Correção por Limite* para que os processos controlados não ficassem instáveis.

$$u_k = \frac{\alpha \cdot \sum_{i=1}^k e_i^2}{2EM^2} \cdot e_k + \frac{\alpha}{2} \cdot \sum_{i=1}^k e_i \quad (3.17)$$

$$KP = \frac{\alpha \cdot \sum_{i=1}^k e_i^2}{2EM^2} \quad (3.18)$$

$$KI = \frac{\alpha}{2} \quad (3.19)$$

Essa análise matemática das equações do controlador ajuda a compreender melhor os pontos positivos e negativos do ONFC e como os seus parâmetros de projeto, ganho de treinamento e erro máximo do universo de discurso, influenciam no controle de um processo. Compreendendo melhor o controlador é possível desenvolver estratégias para se sintonizar os parâmetros do mesmo, já que normalmente são utilizados métodos empíricos para a sintonia do ganho de treinamento e do universo de discurso.

3.1.2 Resultados de simulação

Nesta seção serão apresentados resultados de simulação com o objetivo de se demonstrar a análise matemática desenvolvida através do comportamento dos pesos da rede e também para fazer futuras comparações com esse controlador. Os parâmetros do controlador utilizados são descritos na **Error! Reference source not found.** Tabela 3.1 e os parâmetros do motor CC são descritos no Apêndice A**Error! Reference source not found.**

A Figura 3.4 apresenta resultados de simulação do controle de velocidade do motor CC para uma trajetória em rampa até atingir um estado estacionário seguido de inversão da velocidade através de uma rampa. Aplica-se torque de carga nos instantes de 1,3 e 5,3 segundos e retira-se a carga nos instantes de 2,7 e 6,7 segundos. Observa-se que não foi aplicado degrau como referência isso porque degraus de referência geralmente não são utilizados na prática, aplica-se geralmente rampa, com taxa de crescimento adequada ao processo, como referência para que o controlador tenha uma ação de controle mais suave e force menos os atuadores do processo.

Nota-se através da Figura 3.4 que no início há um erro maior de velocidade. Isso ocorre porque o controlador está ajustando o seu ganho proporcional virtual o que era previsto na análise matemática desenvolvida. Através da Figura 3.5, onde é apresentado o comportamento dos pesos do controlador, pode-se observar que realmente o controlador está ajustando esse ganho proporcional porque esse ganho é função de W_2 menos W_1 e no início esse pesos tem valores próximos, com isso o ganho proporcional virtual está perto de zero e vai aumentando ao longo do tempo.

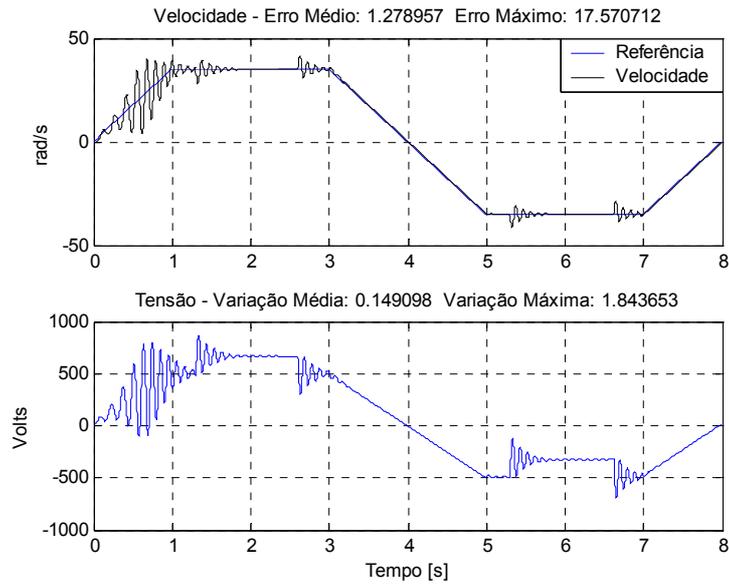


Figura 3.4 – Controle da velocidade do motor CC com aplicação e retirada de carga utilizando o ONFC.

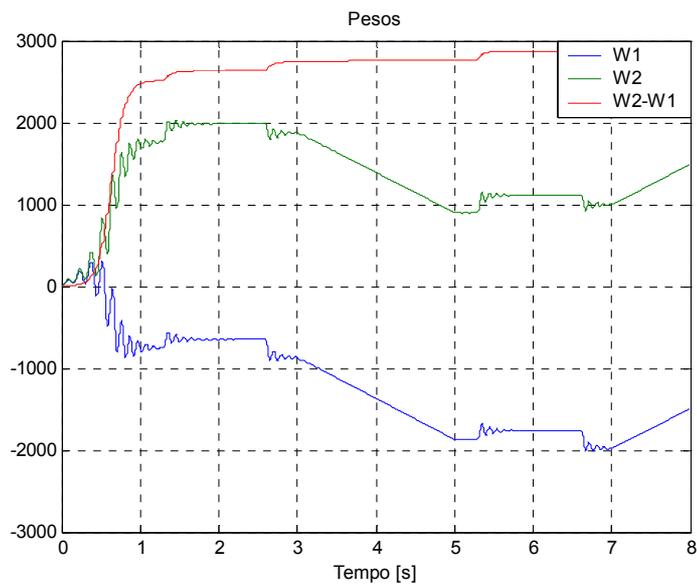


Figura 3.5 – Comportamento dos pesos do ONFC durante o controle de velocidade do motor.

Tabela 3.1 – Parâmetros da rede do controlador ONFC no controle da velocidade do motor CC.

Parâmetros da rede	Valor
<i>Límite da função de pertinência (EM)</i>	40
<i>Ganho de treinamento (α)</i>	4

A Figura 3.6 **Error! Reference source not found.** apresenta resultados de simulação do controle de velocidade do motor CC com uma função senoidal como referência de velocidade e inversão do torque de carga a cada dois segundos utilizando o ONFC. Embora normalmente não seja utilizado esse tipo de referência no controle de velocidade de motores elétricos e tão pouco a inversão do torque de carga, essa situação é explorada com o objetivo de se colocar o controlador em uma situação bem adversa para fazer futuras comparações com o mesmo e para que se possa observar problemas em sua utilização.

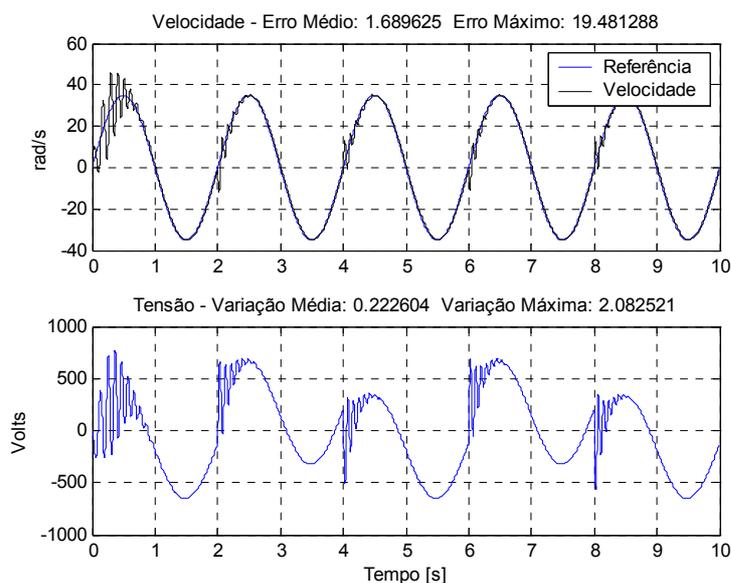


Figura 3.6 – Controle de velocidade do motor CC utilizando o ONFC.

3.2 On-Line Neuro Controller

A rede NFN quando utiliza duas partições em sua estrutura e quando as variáveis de entrada ficam limitadas ao universo de discurso preestabelecido apresenta a característica de ser uma rede linear. Assim, onde se utiliza a rede NFN com essa estrutura e com as variáveis de entrada limitadas ao universo de discurso é possível utilizar uma rede neural linear. Uma diferença significativa entre essas redes ocorre no treinamento. No NFN há uma normalização automática dos dados de entrada através das funções de pertinência e isso é benéfico no aprendizado de uma rede.

O ONFC utiliza a rede NFN com a característica citada anteriormente porque essa rede possui duas partições e normalmente o erro, que é a variável de entrada, fica confinado na grande maioria do tempo ao universo de discurso. Sendo assim, pode-se utilizar o Adaline [Widrow (1962)], a rede linear neural, com o intuito de se obter um controlador neural análogo ao ONFC, o On-Line Neuro Controller (ONC).

A Figura 3.7 apresenta o neurônio representativo do ONC. Na equação (3.20) tem-se a entrada da rede com o termo de polarização. A seguir serão apresentadas as equações desse controlador neural e a análise matemática do mesmo através do desenvolvimento matemático de suas equações.

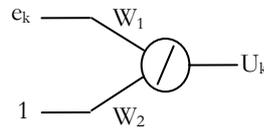


Figura 3.7 – Neurônio representativo do ONC.

$$x_k = [e_k \quad 1] \quad (3.20)$$

A ação de controle para um instante de tempo k é calculada através da equação (3.21). Nota-se que W_1 representa diretamente o ganho proporcional do controlador e W_2 a contribuição integral à ação de controle. No ONFC o ganho proporcional e a contribuição integral são funções dos dois pesos como se pode observar na equação (3.11).

$$u_k = \sum_{i=1}^2 W_{ik} \cdot x_{ik} = W_{1k} \cdot e_k + W_{2k} \quad (3.21)$$

Na equação (3.22) tem-se o ajuste dos pesos onde os pesos iniciais da rede têm valor nulo e cada um possui uma taxa de aprendizado própria.

$$W_{1,2(k)} = W_{1,2(k-1)} + \alpha_{1,2(k)} \cdot e_k \cdot x_k = \sum_{i=1}^k \alpha_{1,2(i)} \cdot e_k \cdot x_k \quad (3.22)$$

Substituindo a equação (3.20) na equação (3.22) tem-se as equações (3.23) e (3.24) de ajuste de cada peso da rede.

$$W_{1(k)} = W_{1(k-1)} + \alpha_{1(k)} \cdot e_k \cdot e_k = \sum_{i=1}^k \alpha_{1(i)} \cdot e_i^2 \quad (3.23)$$

$$W_{2(k)} = W_{2(k-1)} + \alpha_{2(k)} \cdot e_k \cdot 1 = \sum_{i=1}^k \alpha_{2(i)} \cdot e_i \quad (3.24)$$

Substituindo as equações (3.23) e (3.24) em (3.21) tem-se a equação (3.25). Comparando essa equação com a equação do ONFC (3.16), nota-se que as duas equações possuem a mesma forma, diferenciando-se devido à forma com que os ganhos são apresentados. As taxas de aprendizado da rede neural correspondem à rede NFN conforme equações (3.26) e (3.27).

$$u_k = \left(\sum_{i=1}^k \alpha_{1(i)} \cdot e_i^2 \right) \cdot e_k + \sum_{i=1}^k \alpha_{2(i)} \cdot e_i \quad (3.25)$$

$$\alpha_1 = \frac{\alpha}{2EM^2} \quad (3.26)$$

$$\alpha_2 = \frac{\alpha}{2} \quad (3.27)$$

Trabalhando com as taxas de aprendizado constantes, obtém-se a equação (3.28). Dessa forma, pode-se verificar como é a forma do ganho proporcional e do ganho integral através das equações (3.29) e (3.30) respectivamente. Como era de se esperar, nota-se que o ONC apresenta os mesmos problemas do ONFC, o ganho proporcional pode divergir devido ao somatório do erro quadrático, por não se conseguir erro nulo em uma situação real, e em consequência fazer com que o processo se desestabilize. Assim, devem-se ter os mesmos cuidados que se têm no ONFC.

$$u_k = \left(\alpha_1 \cdot \sum_{i=1}^k e_i^2 \right) \cdot e_k + \alpha_2 \cdot \sum_{i=1}^k e_i \quad (3.28)$$

$$KP = \alpha_1 \cdot \sum_{i=1}^k e_i^2 \quad (3.29)$$

$$KI = \alpha_2 \quad (3.30)$$

O análogo neural do ONFC apresenta um custo computacional menor, já que não há o cálculo das funções de pertinência, mas isto não chega a ser uma grande vantagem em relação ao ONFC porque o ONFC já apresenta um custo computacional baixo.

O ONFC apresenta um comportamento não linear de sua rede neurofuzzy quando o erro está fora dos limites do universo de discurso. Essa característica é interessante para o controlador, em contrapartida, isso leva a uma pequena dificuldade na escolha dos parâmetros do controlador por não poder levar em consideração em 100% dos casos as equações levantadas do ganho proporcional (3.18) e ganho integral (3.19) na análise matemática do controlador. Por utilizar uma rede linear, as equações da análise matemática do ONC são sempre válidas. Além disso, no ONC, os ganhos proporcional e integral são influenciados diretamente pelos ganhos de treinamento, o que facilita o projeto do controlador.

3.2.1 Resultados de simulação

A Figura 3.8 apresenta os resultados de simulação do controle da velocidade do motor CC utilizando o On-Line Neuro Controller. Nota-se que no início do controle existe um erro maior de velocidade e que esse erro vai diminuindo ao longo do tempo. Isso ocorre porque, como descrito anteriormente na seção 3.1.1, a constante proporcional do controlador vai se ajustando de forma que o erro diminua. A Figura 3.9 exhibe o comportamento dos pesos da rede neural onde pode se ter uma idéia melhor do ajuste do peso W_1 que representa a constante proporcional do controlador.

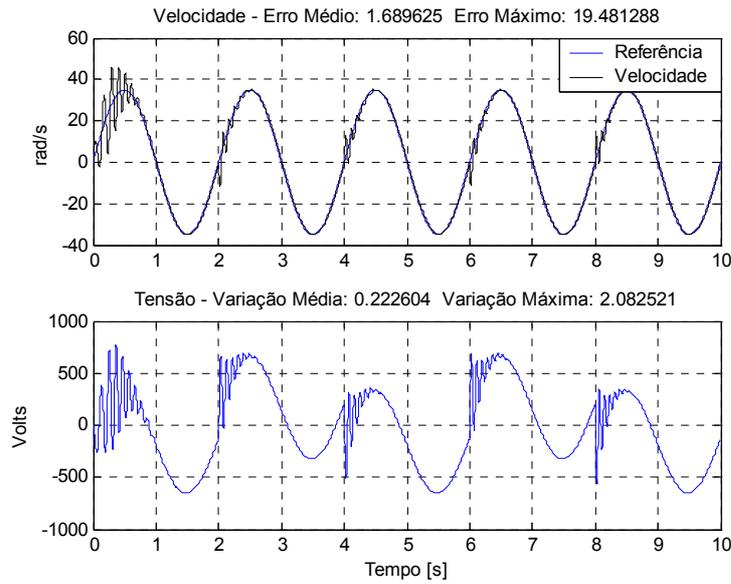


Figura 3.8 – Controle de velocidade do motor CC utilizando o controlador neural análogo ao ONFC.

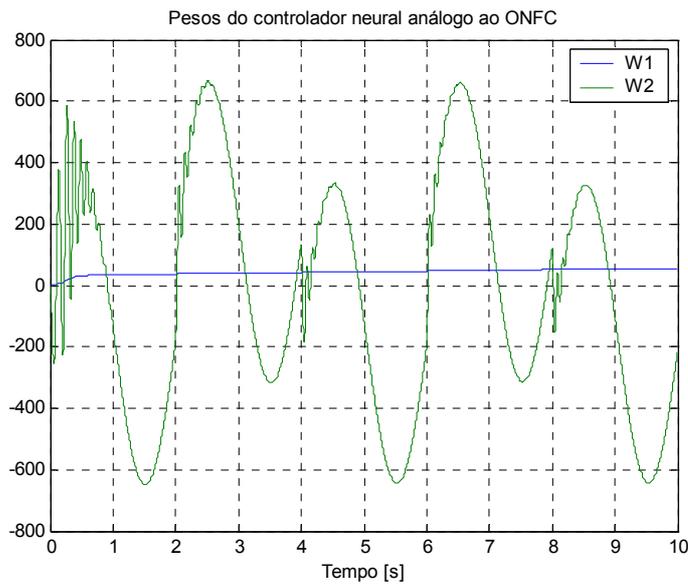


Figura 3.9 – Comportamento dos pesos da rede neural no controle de velocidade do motor CC.

Na Tabela 3.2 têm-se os parâmetros do controlador utilizados no controle da velocidade do motor CC apresentado na Figura 3.8. Esses parâmetros foram escolhidos através das equações (3.26) e (3.27) onde EM e o ganho de treinamento foram descritos na Tabela 3.1 **Error! Reference source not found.** Como os parâmetros foram escolhidos através das

equações de equivalência entre os parâmetros dos dois controladores, o controle da velocidade do motor CC é, como esperado, idêntico em ambos controladores, compare a Figura 3.8 com a Figura 3.6 **Error! Reference source not found.**

Tabela 3.2 – Parâmetros do controlador neural análogo ao ONFC.

Ganho de treinamento em relação aos pesos	Valor
$\alpha_1(W_1)$	0,0013
$\alpha_2(W_2)$	2

3.3 ONFC em modos deslizantes

Parma (2000) propõe a utilização da teoria de Sistemas de Estruturas Variáveis (SEV) [*Itkis (1976), Utkin (1977)*], mais especificamente, Controle por Modos Deslizantes (CMD) [*Utkin (1978)*], para realizar o treinamento de redes neurais artificiais. Essa abordagem possibilita tratar o treinamento da rede como se fosse um processo controlado e possibilita também analisar matematicamente os algoritmos de treinamento. Assim, é possível determinar os limites das taxas de aprendizado envolvidas no treinamento. Essa análise matemática é importante, já que nos algoritmos clássicos as taxas de aprendizado são, em sua maioria, determinadas empiricamente [*Parma (2000)*].

Propõem-se a utilização da teoria de Controle por Modos Deslizantes para realizar o treinamento da rede NFN do ONFC com o objetivo de se obter um sistema mais robusto às variações paramétricas.

No desenvolvimento do algoritmo de ajustes dos pesos para a rede NFN através do Controle por Modos Deslizantes será utilizado como base o algoritmo que corresponde à primeira proposta on-line apresentada por *Parma (2000)* aplicada ao treinamento da rede, mas utilizando uma abordagem discreta direta.

A superfície de deslizamento é dada em função do erro da saída da rede e pela variação deste erro. Essa superfície de deslizamento é dada pela equação (3.31). Onde C é a inclinação da superfície, X_{1i} é o erro na saída da rede (3.33) e X_{2i} a variação de X_{1i} .

$$S_{i(k)} = C \cdot X_{1i(k)} + X_{2i(k)} \quad (3.31)$$

$$X_{1i(k)} = e_k \quad (3.32)$$

$$e_k = y_{d(k)} - y_{(k)} \quad (3.33)$$

$$X_{2i(k)} = X_{1i(k)} - X_{1i(k-1)} \quad (3.34)$$

A regra de atualização dos pesos é dada pela equação (3.35) e (3.36).

$$\Delta W_{i(k)} = \alpha \cdot |X_{1i(k)}| \cdot \text{sgn}(S_{i(k)}) \cdot \mu_{i(k)}(x) \quad (3.35)$$

$$W_{i(k)} = W_{i(k-1)} + \Delta W_{i(k)} \quad (3.36)$$

Onde $\text{sgn}(S_{(k)})$ é dado pela equação (3.37).

$$\text{sgn}(S_{(k)}) = \begin{cases} +1, & S_{(k)} \geq 0 \\ -1, & S_{(k)} < 0 \end{cases} \quad (3.37)$$

Como o ajuste dos pesos é feito utilizando a direção da superfície de deslizamento e não em relação à direção do gradiente descendente, o ganho proporcional virtual deste controlador não será em função do erro quadrático. Desta maneira, ela será adaptativa podendo aumentar e diminuir o seu valor, o que não ocorre com a versão original onde só é possível aumentar o seu valor. Esse ponto é uma grande contribuição em relação ao ONFC, pois o fato do ganho proporcional virtual somente poder aumentar é o principal ponto negativo do controlador já que a divergência do mesmo pode fazer com que o processo chegue à instabilidade.

3.3.1 Convergência

Após a definição da superfície de deslizamento (3.31) e as regras de atualização dos pesos (3.35) e (3.36), é possível determinar o limite da taxa de aprendizado, o termo α . Este estudo

baseia-se na convergência do ponto representativo (PR) pelo plano das variáveis que a definem. Essa convergência é dada pelo Teorema mostrado abaixo [Parma (2000)].

Teorema

Seja a superfície definida por $S_{(k)} = -A \cdot X_{(k)} \cdot \text{sgn}(S_{(k)}) + Y_{(k)}$, tal que em regime de deslizamento $S_{(k)} \rightarrow 0$. $S_{(k)} : \mathfrak{R}^2 \rightarrow \mathfrak{R}, \{X_{(k)}, Y_{(k)}, A\} \in \mathfrak{R}$, tal que $X_{(k)} > 0$, $A > 0$ e

$\text{sgn}(S_{(k)}) = \begin{cases} +1, & S_{(k)} \geq 0 \\ -1, & S_{(k)} < 0 \end{cases}$. Então, um ponto representativo (PR) do sistema no espaço de

estado bidimensional, irá convergir à superfície de deslizamento se:

$$A < \min \left\{ \frac{|Y_{(k)}|}{X_{(k)}}, \frac{|Y_{(k-1)}| - |Y_{(k)}|}{X_{(k-1)} - X_{(k)}} \right\} \quad (3.38)$$

A fim de aplicar o teorema, serão feitas manipulações matemáticas de forma que a equação que define a superfície de deslizamento apresente estrutura semelhante à estrutura apresentada no Teorema. Assim, será possível utilizar a equação (3.38) para se definir o limite da taxa de aprendizado, α .

Substituindo (3.32) em (3.34) tem-se:

$$X_{2i(k)} = (y_{d(k)} - y_{(k)}) - (y_{d(k-1)} - y_{(k-1)}) \quad (3.39)$$

Reescrevendo (3.39) tem-se (3.40):

$$X_{2i(k)} = -\Delta y_{(k)} + \Delta y_{d(k)} \quad (3.40)$$

A equação (3.41) exibe uma forma de se obter um delta y.

$$\Delta y_{(k)} = \mu_{i(k)} \cdot \Delta W_{i(k)} \quad (3.41)$$

Substituindo a equação (3.41) em (3.40):

$$X_{2i(k)} = -(\mu_{i(k)} \cdot \Delta W_{i(k)}) + \Delta y_{d(k)} \quad (3.42)$$

Substituindo em (3.42) na equação (3.35):

$$X_{2i(k)} = -(\mu_{i(k)}^2 \cdot \alpha \cdot |X_{1i(k)}| \cdot \text{sgn}(S_{i(k)})) + \Delta y_{d(k)} \quad (3.43)$$

Substituindo (3.43) e (3.32) na equação da superfície de deslizamento (3.31), tem-se a superfície de deslizamento na seguinte forma:

$$S_{i(k)} = -\alpha \cdot |X_{1i(k)}| \cdot \text{sgn}(S_{i(k)}) \cdot \mu_{i(k)}^2 + (C \cdot X_{1i(k)} + \Delta y_{d(k)}) \quad (3.44)$$

Definindo $X_{(k)}$ e $Y_{(k)}$ através das equações (3.45) e (3.46) respectivamente, pode-se reescrever a equação (3.44) na equação (3.47) de forma a utilizar o Teorema e definir o limite da taxa de aprendizado, α , apresentado na equação (3.48).

$$X_{(k)} = |X_{i(k)}| \mu_{i(k)}^2(x) \quad (3.45)$$

$$Y_{(k)} = C e_{(k)} + \Delta y_{d(k)} \quad (3.46)$$

$$S_{i(k)} = -\alpha \cdot X_{(k)} \cdot \text{sgn}(S_{i(k)}) + Y_{(k)} \quad (3.47)$$

$$\alpha < \min \left\{ \frac{|Y_{(k)}|}{X_{(k)}}, \frac{|Y_{(k-1)}| - |Y_{(k)}|}{X_{(k-1)} - X_{(k)}} \right\} \quad (3.48)$$

É importante salientar que embora os limites da taxa de aprendizado tenham sido estabelecidos na rede NFN, eles não são aplicáveis no treinamento da rede NFN do ONFC. Isso porque quando se utiliza a rede para atuar como um controlador utiliza-se o erro entre a saída da planta e a referência na superfície de deslizamento e não o erro entre a saída da rede e o padrão de saída para treinamento. Assim, a princípio, não é possível usar os limites para se determinar a taxa de aprendizado que poderia ser utilizada em cada iteração. Para ser possível, deve-se incorporar o efeito do sistema a ser controlado nas equações de ajustes dos pesos. Basicamente, o efeito a ser considerado é a variação da saída da planta em relação à

variação da entrada. Mas o primeiro passo já está feito que é realizar o treinamento da rede NFN com o controle em modos deslizantes.

3.3.2 Resultados de simulação

A Figura 3.10 apresenta os resultados de simulação do controle da velocidade do motor CC com inversão do torque de carga a cada dois segundos utilizando o ONFC em modos deslizantes onde se utilizou os parâmetros do controlador exibidos na Tabela 3.3. Esses parâmetros foram escolhidos utilizando como base os parâmetros da Tabela 3.1 **Error! Reference source not found.**

A Tabela 3.4 exhibe a comparação entre o ONFC e o ONFC em modos deslizantes no controle da velocidade do motor CC apresentado na Figura 3.10 e na Figura 3.6 **Error! Reference source not found.** Nota-se que o ONFC em modos deslizantes apresenta um erro máximo menor que o ONFC, em contra partida ele apresenta um erro médio maior.

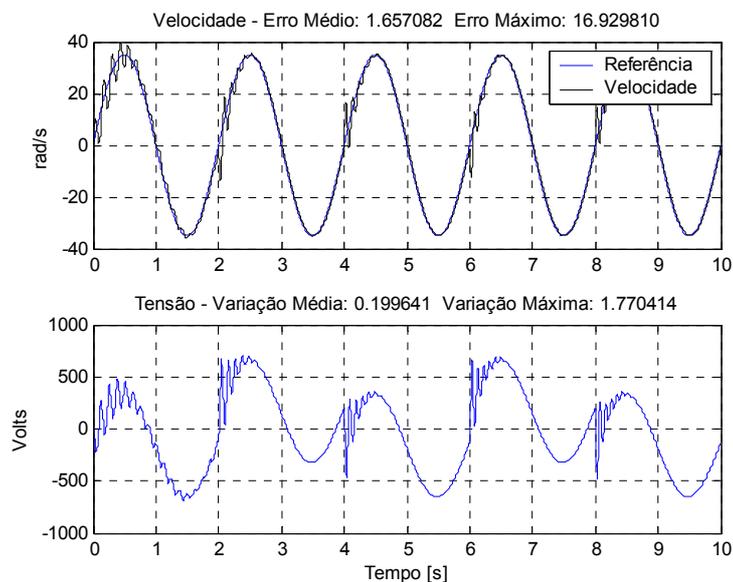


Figura 3.10 – Controle de velocidade do motor CC utilizando o ONFC em modos deslizantes.

Tabela 3.3 – Parâmetros da rede do controlador ONFC em modos deslizantes utilizado no controle da velocidade do motor CC.

Parâmetros da rede	Valor
Limite da função de pertinência (EM)	40
Taxa de aprendizado (α)	4
Inclinação da superfície de deslizamento (C)	0,1

Tabela 3.4 – Comparação entre o ONFC e o ONFC em modos deslizantes no controle da velocidade do motor CC.

Índice	ONFC	ONFC em modos deslizantes
Erro Máximo	19,481288	16,929810
Erro Médio	1,689685	1,658592

3.4 Conclusões

Neste capítulo foi feita uma breve introdução do controlador neurofuzzy proposto por *Gouvêa (2005)*, o ONFC, e um estudo sobre o mesmo através da análise matemática de suas equações. Foi apresentado também o análogo neural deste controlador, o On-Line Neuro Controller (ONC), e o ONFC em modos deslizantes.

Através da análise matemática das equações envolvidas no controle de um processo do ONFC foi possível entender que ele é, em síntese, um controlador PI adaptativo. Essa análise matemática foi importante também para propor um controlador neural análogo ao ONFC, o On-Line Neuro Controller (ONC). Por utilizar uma rede linear, o que não ocorre com o ONFC, as equações da análise matemática desse controlador são sempre válidas. Além disso, os pesos da rede desse controlador possuem uma relação mais direta com a ação de controle e com as constantes de um PI do que no ONFC convencional. Por ser ainda mais simples do que o ONFC, ele apresenta um custo computacional ainda menor do que o ONFC, mas que não chega a ser uma grande vantagem já que o ONFC apresenta um custo computacional baixo. Devido a essas características, o uso desse controlador se faz interessante em plantas simples onde não é necessário utilizar o ONFC com o erro ultrapassando os limites do universo de discurso.

O ONFC em modos deslizantes apresentou resultados semelhantes ao ONFC convencional. Uma vantagem em se utilizar sistemas de estrutura variáveis, mais especificamente controle por modos deslizantes, para ajustar os pesos da rede é que devido à superfície de deslizamento é possível obter resultados melhores do que no caso convencional e também sistemas mais robustos as variações paramétricas. Em contrapartida, se aumenta o custo computacional do controlador e também a sua complexidade, já que é necessário ajustar mais um parâmetro no controlador, a inclinação da superfície de deslizamento (C).

Outra vantagem em se utilizar controle por modos deslizantes para ajustar os pesos da rede é, caso se incorpore o efeito do sistema a ser controlado nas equações de ajustes dos pesos, a possibilidade de determinar automaticamente a taxa de aprendizado que deve ser utilizada no controlador. Isso é possível devido a análise de convergência do algoritmo de treinamento da rede, onde são determinados os limites da taxa de aprendizado. Por ser um controlador mais complexo do que o ONFC, o seu uso é indicado em processos onde é possível incorporar esse efeito do sistema a ser controlado nas equações de convergência do controlador possibilitando assim uma maior robustez no controle do processo quanto às variações paramétricas.

Capítulo 4

Controle Neurofuzzy Baseado na Dinâmica Inversa

Neste capítulo será apresentado o controle baseado na dinâmica inversa do processo. O controle utilizando a dinâmica inversa será aplicado no controle de velocidade de um motor CC. Serão mostradas as dificuldades em se obter o modelo artificial da função inversa da planta em um sistema dinâmico e as características desse controlador.

4.1 Introdução

Uma maneira intuitiva de se controlar um processo é pensar da seguinte maneira: qual é a ação de controle que é preciso aplicar na planta de forma que ela possa sair do estado atual para atingir outro estado desejado? Uma maneira de responder a esta pergunta é projetar o controlador pelo método direto, que neste caso equivale a obter a função inversa da planta. O problema é que os processos são sistemas dinâmicos e devido a isso, existe uma dificuldade em se obter tal função.

Em um sistema dinâmico, a saída da planta dependente de valores atuais e passados da entrada e da própria saída como é mostrado na equação (4.1).

$$y(k+1) = f(y(k), y(k-1), \dots, y(k-n), u(k), u(k-1), \dots, u(k-m)) \quad (4.1)$$

A obtenção da função inversa da planta para implementar o controle utilizando a dinâmica inversa segue a equação (4.2).

$$u(k) = g(y(k+1), y(k), y(k-1), \dots, y(k-n), u(k-1), u(k-2), \dots, u(k-m)) \quad (4.2)$$

A Figura 4.1 mostra o diagrama de blocos de controle utilizando a dinâmica inversa, cujo objetivo é anular o efeito da planta colocando-a em série com a sua função inversa, já que o produto da planta pela sua inversa resultaria na unidade. Assim, a referência será a saída da planta no instante seguinte de amostragem.

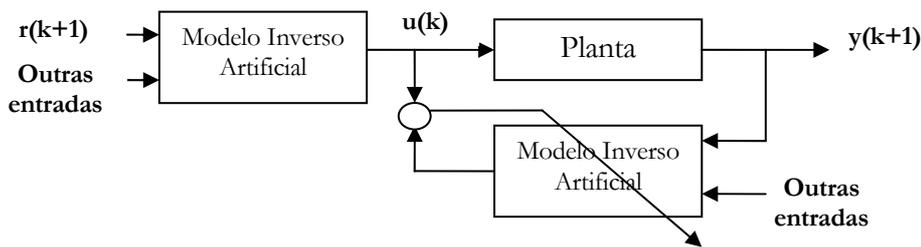


Figura 4.1 – Diagrama de blocos de controle utilizando dinâmica inversa.

Outra forma de entender como se pode utilizar a dinâmica inversa é através da equação (4.2). Substituindo $y(k+1)$ por $r(k+1)$ na equação se consegue obter a entrada da planta desejada para que se alcance esse objetivo.

Encontrar o modelo inverso da planta é bem difícil, mesmo utilizando técnicas de inteligência computacional. Uma das dificuldades em se obter a inversa de um sistema dinâmico pode ser vista analisando a resposta ao degrau mostrada na Figura 4.2. Na resposta ao degrau tem-se que a planta sai de um estado, que depende das condições iniciais, para atingir outro estado estacionário. Dessa forma, uma mesma entrada é aplicada na planta e, como consequência, têm-se várias saídas. Não existe uma função simples que mapeie a entrada em função da saída para funções desse tipo.

É importantíssimo que a amostragem do sinal seja feita com um período de tempo adequado. Uma super amostragem ou um período de amostragem pequeno prejudica bastante a obtenção da inversa já que os estados da planta praticamente não mudam entre uma amostragem e outra. Eles somente mudarão depois de vários períodos de amostragem. Uma amostragem de cinco a dez vezes menor que a constante de tempo da planta é adequada.

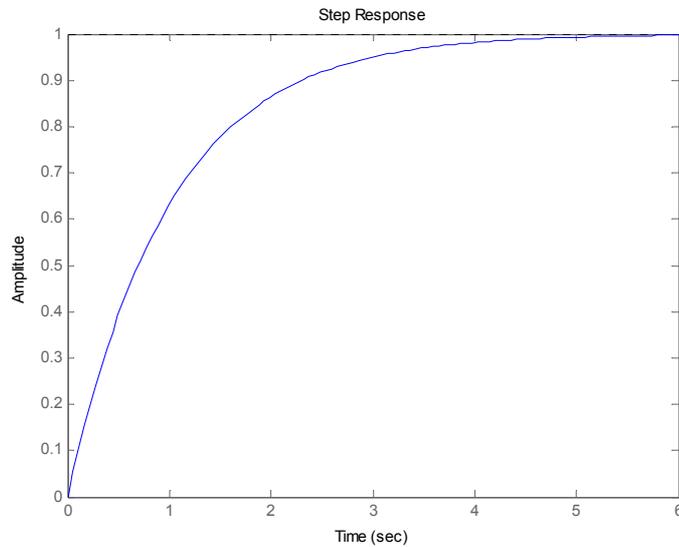


Figura 4.2 – Resposta ao degrau para a planta $G(s) = 1/(s+1)$.

Para processos rápidos que não necessitam aumentar a sua velocidade em malha fechada pode-se amostrar o sinal de forma que a dinâmica não seja notada. Nesse caso, basta mapear a entrada com a saída e se teria condição de controlar um processo de uma maneira bem simples.

Uma característica do controlador que utiliza a lógica inversa é que ele tem capacidade de previsão. Devido a isso, o controlador teoricamente apresenta erro nulo para acompanhar uma trajetória em rampa o que não acontece com os controladores que utilizam o erro para calcular a ação de controle como é o caso do PI.

Depois que a função inversa estima a ação de controle tem-se sempre que verificar se essa ação de controle pode ser aplicada e se ela pode ser fornecida ao processo. Isso porque o controlador estima uma ação de controle que seja capaz de fazer com que o erro seja nulo no instante seguinte, mesmo que para isso a ação de controle tenha que variar bastante e assumir um valor que não pode ser aplicado na planta ou no qual o processo não tem capacidade de fornecer. Através da formulação mostrada na equação (4.3) garante-se que não é aplicada nenhuma entrada acima dos limites da planta, assim como, uma variação da ação de controle maior que o permitido. Isso é necessário para garantir que as limitações físicas do processo serão respeitadas.

$$\begin{aligned}
& \text{Se } \tilde{u}_k \geq u_{k-1} \\
& \quad u_k = \min(\tilde{u}_k, u_{\max}, u_{k-1} + \Delta u_{\max}) \\
& \text{Se } \text{n\~{a}o} \\
& \quad u_k = \max(\tilde{u}_k, u_{\max}, u_{k-1} - \Delta u_{\max})
\end{aligned} \tag{4.3}$$

4.2 Controle de velocidade do motor CC

Para obter o modelo artificial da função inversa utilizam-se os dados históricos do processo. A rede utilizada aqui será o NFN por ser uma rede de baixo custo computacional e por apresentar treinamento por mínimos quadrados.

Utilizar uma massa de dados que contenha o máximo de informação possível sobre os estados da planta é de fundamental importância para se obter um bom modelo inverso. Os dados utilizados no treinamento são provenientes de diversos degraus aplicados no motor de amplitude aleatória e tempo de duração também aleatório, o mesmo acontece com o conjugado de carga que é aplicado no motor, de forma a tentar atingir o objetivo acima.

Em uma situação real, os dados para treinamento off-line do modelo podem ser obtidos através de dados históricos do processo. Caso não se possua esses dados históricos, situação de uma planta em que ainda não foi utilizada no processo, deve-se submeter à planta a várias situações de operação atendo-se com o objetivo de obter o máximo de informação possível sobre os estados da planta. Essa preocupação também deve existir com os dados históricos do processo porque, como descrito anteriormente, caso o mesmo não tenha boas informações sobre a planta, não será possível obter um bom modelo. Isso pode acontecer, por exemplo, em uma planta que fica trabalhando em grande parte do tempo em um único estado. Uma boa maneira de obter padrões distintos sobre os estados da planta é utilizar métodos de separação de cluster como, por exemplo, o **kmeans** [Spath (1985)].

Nas simulações a seguir é utilizada a rede NFN com duas partições. A Figura 4.3 exibe o diagrama de blocos com as entradas utilizadas na rede para modelar a dinâmica inversa onde ω representa a velocidade, u a tensão e I a corrente de armadura.

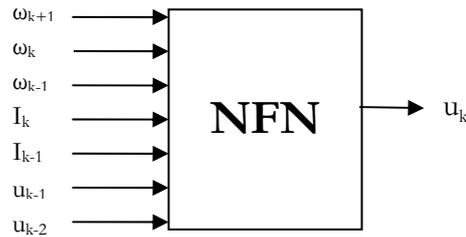


Figura 4.3 – Entradas utilizadas na rede NFN com duas partições para modelar a dinâmica inversa do motor CC.

A corrente de armadura é um estado interno do motor que facilita muito a obtenção da função inversa da planta como poderá ser visualizado adiante. Ela é também uma ótima informação sobre o conjugado de carga que é aplicado no motor, o que faz com que o controlador fique robusto às variações de carga.

Será feita a comparação do controlador dinâmico inverso com o PI, onde os parâmetros do controlador PI foram encontrados através do método de otimização do Algoritmo Genético (AG) com o objetivo de obter a melhor resposta de um PI e para não sermos parciais em nossas conclusões.

Para encontrar os parâmetros do controlador PI com o método de otimização do Algoritmo Genético, utilizou-se como função objetivo a minimização da função de custo J , somatório do erro quadrático, dada na equação (4.4). O controle da velocidade do motor CC foi executado como uma senóide de referência e com inversão do torque de carga a cada 2 segundos durante 10 segundos.

$$J = \sum_k (r_k - y_k)^2 \quad (4.4)$$

A Figura 4.4 apresenta a validação do treinamento do modelo da função inversa quando se utiliza o método dos mínimos quadráticos e a Figura 4.5 apresenta a validação utilizando o método do gradiente. Observa-se que o erro médio utilizando o método dos mínimos quadráticos é quase que três vezes menor do que utilizando o método do gradiente. A primeira vista, o erro médio de validação igual a 39,408973 pode parecer um erro médio grande, mas considerando que a função inversa está prevendo também uma condição de

funcionamento do motor com conjugado de carga variável, esse erro passa a ser considerado satisfatório.

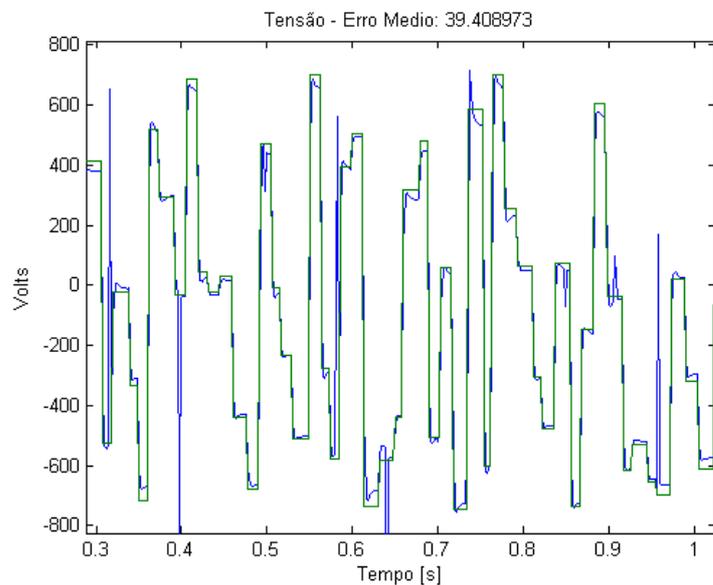


Figura 4.4 – Validação da rede NFN com treinamento através dos mínimos quadráticos.

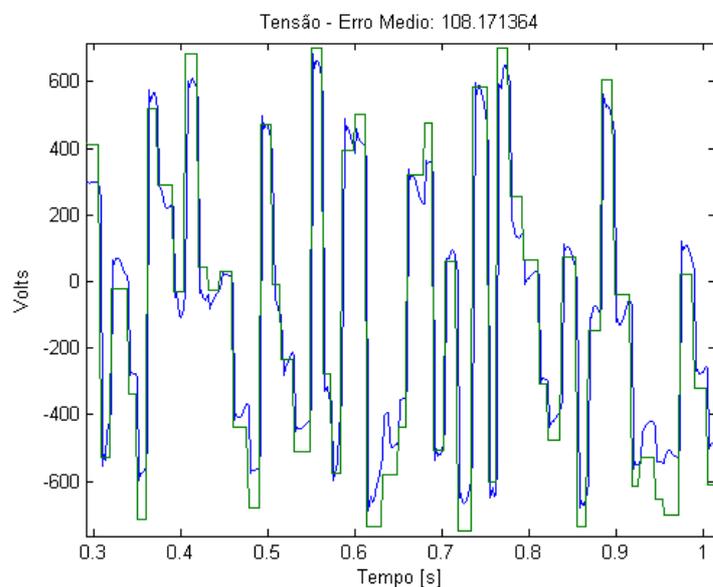


Figura 4.5 – Validação da rede NFN com treinamento através do gradiente.

Nas simulações a seguir será utilizada a rede que foi treinada através do método dos mínimos quadráticos, já que os resultados obtidos com a rede treinada através do gradiente não apresentaram resultados satisfatórios.

4.2.1 Simulações com treinamento off-line

Nas simulações a seguir a rede NFN é treinada para modelar a função inversa do motor CC primeiramente e depois é utilizada para controlar a velocidade do motor CC sem treinamento on-line.

A Figura 4.6 apresenta o controle da velocidade do motor CC utilizando o modelo artificial inverso da planta para uma referência em rampa com inversão de velocidade e aplicação e retirada de torque de carga. Praticamente não se vê diferença entre a referência e a velocidade, com exceção no momento onde ocorre aplicação e retirada do torque de carga. Observa-se que o controlador rejeita bem as perturbações de carga. Comparando com o ONFC na mesma condição através da Figura 3.4, observa-se que a dinâmica inversa apresenta resultados semelhantes, não considerando a parte inicial do controle no ONFC já que o mesmo ainda está ajustando o seu ganho proporcional virtual.

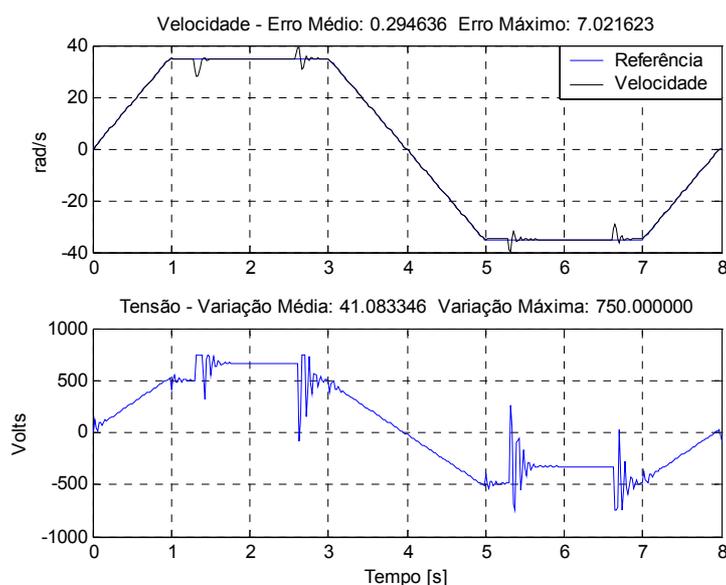


Figura 4.6 – Controle de velocidade do motor CC utilizando a dinâmica inversa com aplicação e retirada de carga.

A Figura 4.7 apresenta o controle da velocidade do motor CC utilizando o modelo artificial inverso da planta. Praticamente não se vê diferença entre a referência e a velocidade, logo o erro médio é bem pequeno, mesmo tendo uma senoíde como referência. A Figura 4.8 apresenta o controle de velocidade do motor CC utilizando um controlador PI. Observa-se que o PI apresenta erro maior do que utilizando a dinâmica inversa. Um dos motivos do erro ser maior no PI é porque ele sempre apresentará erro para seguir uma trajetória não estacionária.

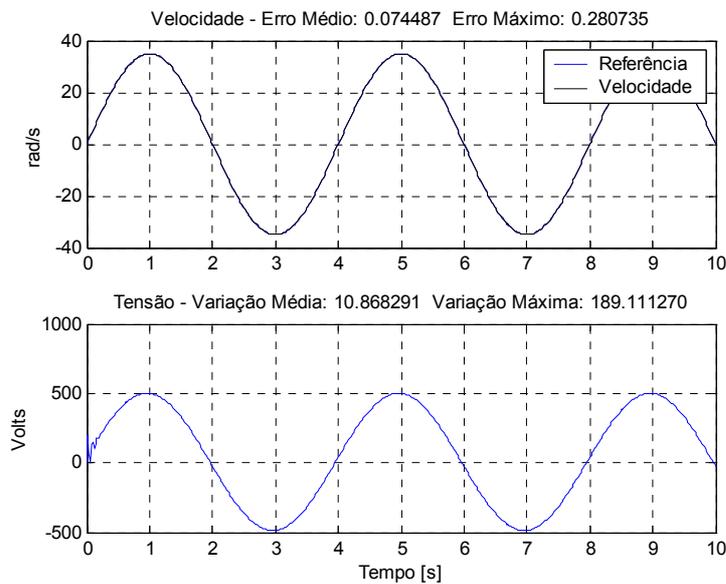


Figura 4.7 – Controle de velocidade do motor CC utilizando a dinâmica inversa.

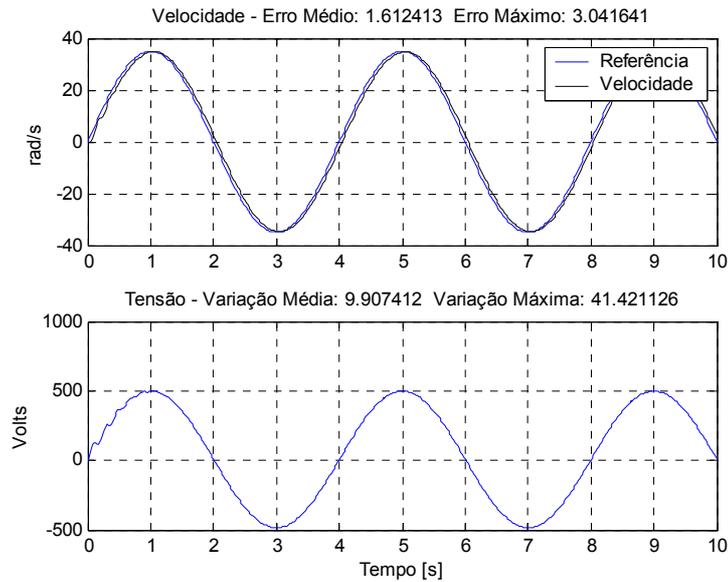


Figura 4.8 – Controle de velocidade do motor CC utilizando um PI.

4.2.1.1 Inserção de ruído na medição de velocidade

Nas simulações a seguir é inserido ruído na medição da velocidade de média zero e Variância igual a $0,5^2$. Esse ruído é inserido para se avaliar o comportamento do controlador frente a uma situação real onde existe uma limitação de instrumentação.

A Figura 4.9 apresenta o controle da velocidade do motor CC com ruído na medição da velocidade. O erro aumentou em relação à medição sem ruído como era de se esperar. Observa-se uma ação de controle que varia bastante, bem mais do que no controle sem ruído na medição, com o objetivo de se ter erro nulo de velocidade no instante seguinte. Essa ação de controle normalmente não deve ser aplicada na planta e por isso deve ser limitada pela formulação apresentada na equação (4.3), mas aqui a utiliza sem a limitação para mostrar as características da ação de controle nessas condições. O comportamento do controlador nessa situação pode ser comparado a de um PID que amplifica o ruído de medição. A ação derivativa do PID tem um comportamento preditivo e devido ao ruído, o comportamento do controlador nessa situação não é bom. O controlador utilizando a dinâmica inversa também apresenta essa característica já que também tem a capacidade de previsão. Por isso, o controlador estima uma ação de controle que seja capaz de fazer com que se tenha erro nulo no instante seguinte mesmo que para isso a ação de controle tenha que variar bastante

assumindo valores que não podem ser aplicados na planta e que a fonte não tem capacidade de fornecer.

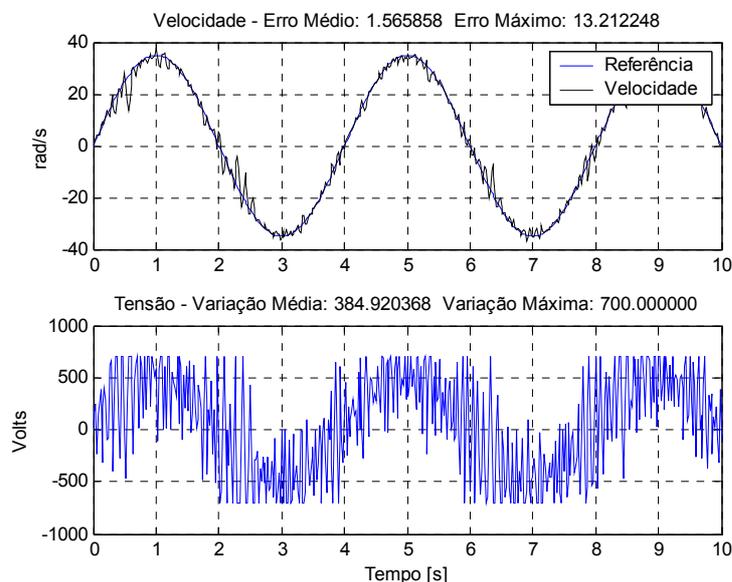


Figura 4.9 – Controle de velocidade do motor CC utilizando dinâmica inversa com ruído na medição de velocidade.

Comparando o controle apresentado na Figura 4.9 com o controle, utilizando o PI, mostrado na Figura 4.10 tem-se que o erro médio é equivalente e o erro máximo é bem menor no PI. A ação de controle aplicada pelo PI é bem mais suave. O PI tem a vantagem de a ação integradora diminuir a velocidade do processo atuando como um filtro passa baixa que minimiza o efeito do ruído.

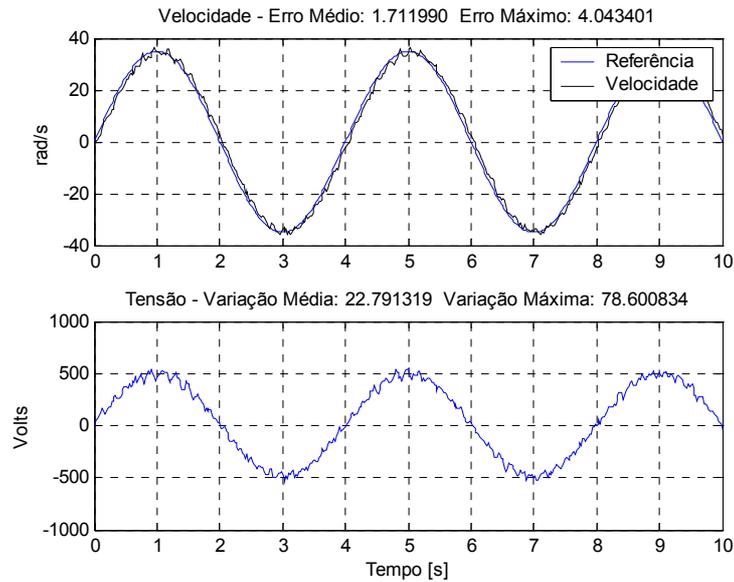


Figura 4.10 – Controle de velocidade do motor CC com PI com ruído na medição de velocidade.

4.2.1.2 Inversão do torque de carga

Nas simulações a seguir é aplicado torque de carga no motor, onde o torque de carga é invertido ($Tl_t = -Tl_{t-2}$) a cada dois segundos. O Torque de carga que é aplicado no motor inicialmente é igual a 500.000 Nm.

A Figura 4.11 apresenta o controle da velocidade do motor CC com a inversão do torque. O erro aumentou em relação à medição sem ruído como era de se esperar. A ação de controle que é aplicada no motor varia bastante com o objetivo de se ter erro nulo no instante seguinte. A variação média da tensão é bem maior do que na situação sem ruído na medição apresentada na Figura 4.7.

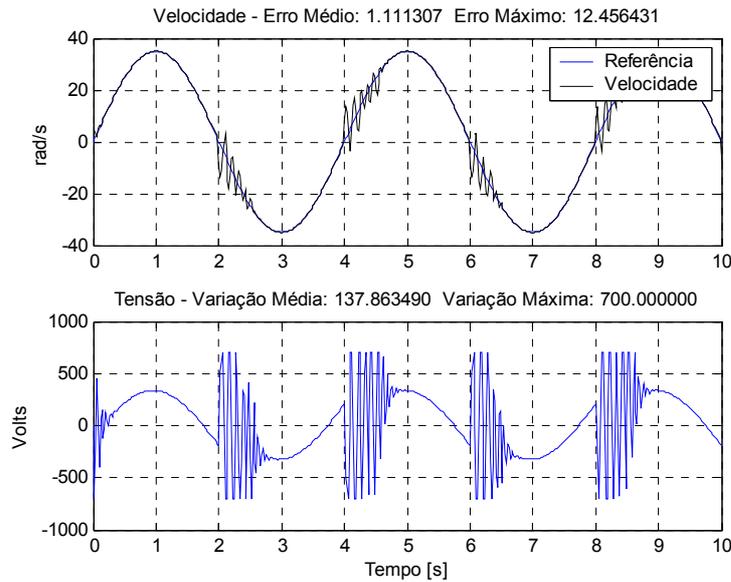


Figura 4.11 – Controle de velocidade do motor CC utilizando dinâmica inversa com inversão de torque a cada dois segundos.

O controle com o PI com a inversão do torque é apresentado na Figura 4.12. A rejeição ao efeito de carga é melhor no controle com a dinâmica inversa tanto em relação ao erro máximo quanto ao tempo gasto para que o erro seja nulo. O erro máximo no PI é de 13,5427 contra 12,4564 na dinâmica inversa. Em compensação a ação de controle aplicada pelo PI varia bem menos.

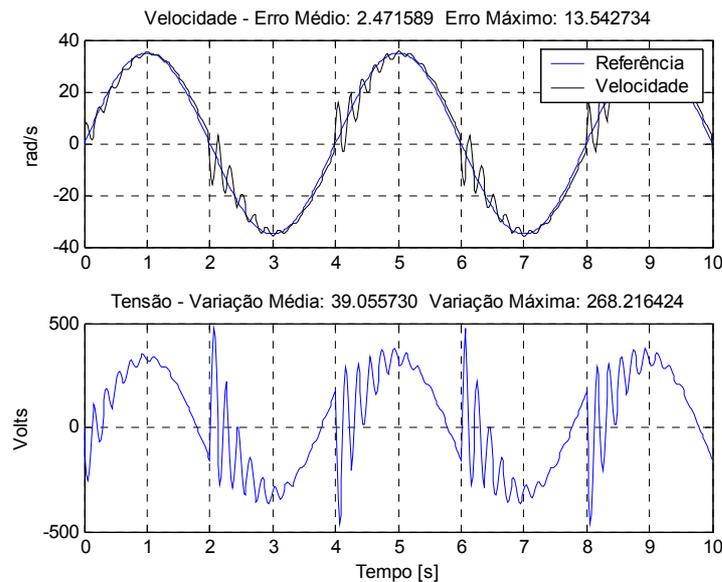


Figura 4.12 – Controle de velocidade do motor CC utilizando PI com inversão de torque a cada dois segundos.

Como descrito anteriormente, a função inversa não prevê que a ação de controle tem que ser limitada. A seguir têm-se simulações limitando a variação da ação de controle.

4.2.1.3 Variação da ação de controle limitada em 70 volts com ruído na medição da velocidade

A Figura 4.13 apresenta o controle da velocidade do motor com a inserção de ruído na medição da velocidade onde a variação da ação de controle é limitada em 70 volts, aproximadamente 10 vezes menor que a tensão nominal. Observa-se que a velocidade fica sempre oscilando em torno da referência.

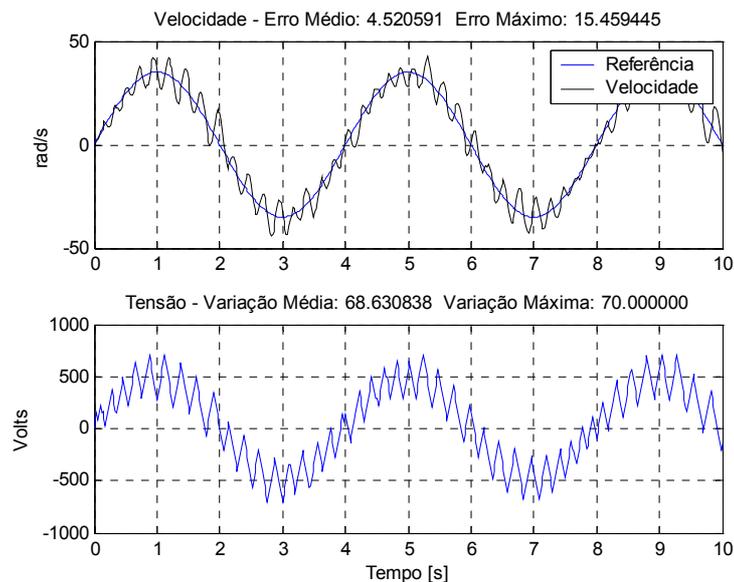


Figura 4.13 – Controle da velocidade do motor CC com ruído na medição de velocidade e com a variação da ação de controle limitada em 70 volts.

4.2.1.4 Variação da ação de controle limitada em 70 volts com inversão do torque de carga

A Figura 4.14 apresenta o controle da velocidade do motor com a inversão do torque onde a variação da ação de controle é limitada em 70 volts, aproximadamente 10 vezes menor que a tensão nominal. Observa-se que o controle da velocidade não é feito com êxito, já que a

velocidade fica sempre oscilando em torno da referência da mesma maneira que na condição com ruído na medição de velocidade, veja Figura 4.13.

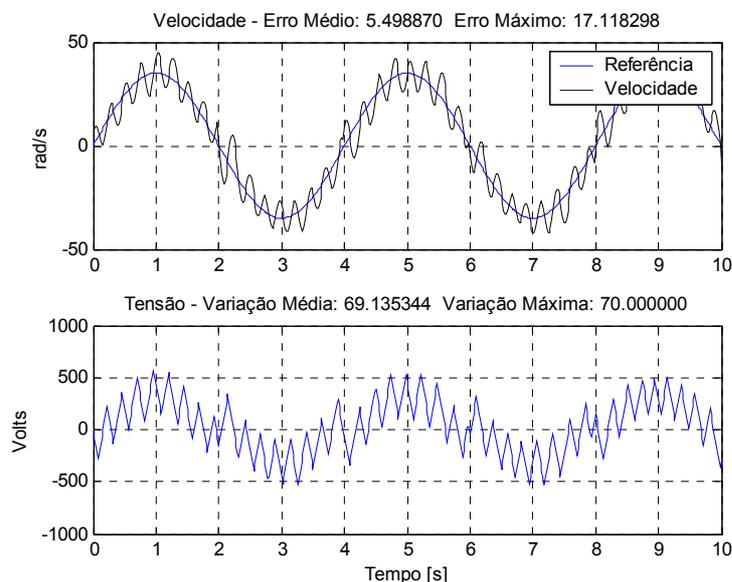


Figura 4.14 – Controle da velocidade do motor CC com inversão de torque e com a variação da ação de controle limitada em 70 volts.

4.2.1.5 Variação da ação de controle limitada em 17,5 volts com ruído na medição da velocidade

A Figura 4.15 apresenta o controle da velocidade do motor com a adição de ruído na medição de velocidade onde a variação da ação de controle é limitada em 17,5 volts, aproximadamente 40 vezes menor que a tensão nominal. Diferentemente da situação apresentada anteriormente onde a ação de controle é limitada em 70 volts, a velocidade apresenta um pequeno erro e consegue convergir na referência.

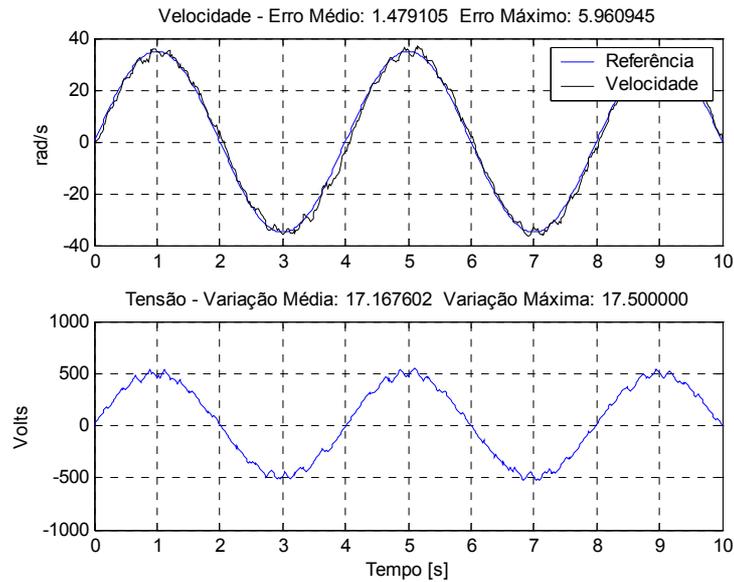


Figura 4.15 – Controle da velocidade do motor CC com ruído na medição da velocidade e com a variação da ação de controle limitada em 17,5 volts.

4.2.1.6 Variação da ação de controle limitada em 17,5 volts com inversão do torque de carga

A Figura 4.16 apresenta o controle da velocidade do motor com a inversão do torque onde a variação da ação de controle é limitada em 17,5 volts, aproximadamente 40 vezes menor que a tensão nominal. Diferentemente da situação apresentada anteriormente onde a ação de controle é limitada em 70 volts, a velocidade apresenta um erro tolerável e consegue convergir na referência.

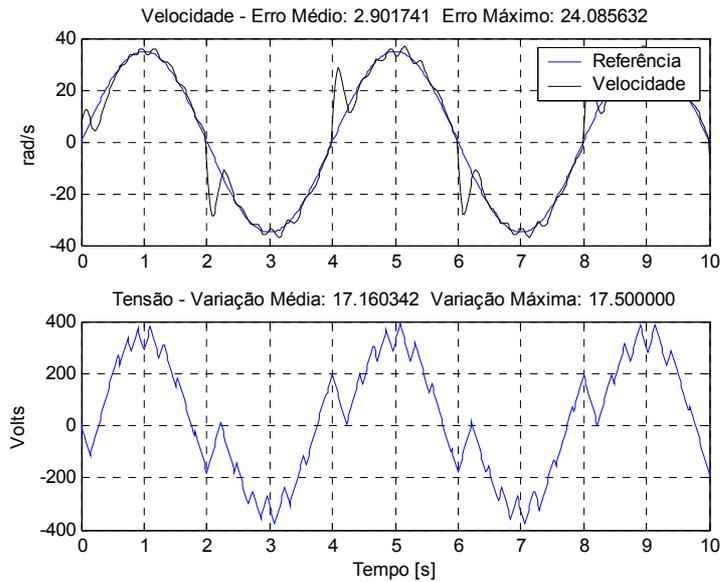


Figura 4.16 – Controle da velocidade do motor CC com inversão de torque e com a variação da ação de controle limitada em 17,5 volts.

4.2.1.7 Comparativo entre o PI e o controlador utilizando a dinâmica inversa

Nesta seção é feita a comparação entre os dois controladores para outras condições de operação. Serão utilizados os mesmos índices de desempenho que foram utilizados antes e que são listados a seguir: Erro médio de velocidade, erro máximo de velocidade, variação média da ação de controle e variação máxima da ação de controle. O erro de velocidade e a variação da ação de controle são índices de objetivos conflitantes, normalmente quando se obtém um bom valor de um determinado índice o outro índice fica prejudicado.

A Tabela 4.1 apresenta dados para análise comparativa entre o controle da velocidade do motor CC utilizando o PI e a dinâmica inversa. Não há inserção de ruído na medição de velocidade e também não há inversão do torque de carga. Pode-se observar que o controle utilizando a dinâmica inversa apresentou resultados melhores em relação ao erro médio e máximo de velocidade. A exceção é quando se utiliza a máxima variação da ação de controle de 70 volts. É interessante comentar que o controle baseado na dinâmica inversa apresenta erro médio de velocidade com máxima variação de 17,5 volts menor que o PI, mesmo quando se utiliza a máxima variação de 70 volts no mesmo. Em termos de variação média da

ação de controle, nota-se que o melhor resultado que se obteve foi quando se utilizou a máxima variação de 700 volts. Isso se deve ao fato que o controlador conseguiu colocar a planta em um estado que corresponde mais com o modelo da função inversa da planta.

Tabela 4.1 – Comparação entre o PI e o controle baseado na Dinâmica Inversa (DI) sem inserção de ruído na medição de velocidade e inversão do torque de carga.

Máxima Variação da Ação de Controle [Volts]	Erro Médio Velocidade [rad/s]		Erro Máximo Velocidade [rad/s]		Variação Média Ação de Controle [Volts]		Variação Máxima Ação de Controle [Volts]	
	DI	PI	DI	PI	DI	PI	DI	PI
700,0	0,074487	1,6124	0,280735	3,0416	10,8682	9,9074	189,1112	41,4211
70,0	4,6694	1,6124	12,1472	3,0416	68,3282	9,9074	70,0000	41,4211
17,5	0,8925	1,6115	4,0627	4,3468	16,6642	9,9477	17,5000	17,5000

A Tabela 4.2 apresenta dados para análise comparativa entre o controle da velocidade do motor CC utilizando o PI e a dinâmica inversa onde é inserido ruído na medição de velocidade. Assim como nos resultados apresentados na Tabela 4.1, observa-se que o controle utilizando a dinâmica inversa apresentou resultados melhores em relação ao erro médio de velocidade. A exceção ocorre quando se utiliza a máxima variação da ação de controle de 70 volts. Em termos de variação média da ação de controle, nota-se que os índices são menores no PI. Utilizando a variação de 17,5 volts na dinâmica inversa, os índices de desempenho são melhores do que o PI mesmo comparando, em alguns casos, com a máxima variação de 70 volts.

Tabela 4.2 – Comparação entre o PI e o controle baseado na Dinâmica Inversa na condição de inserção de ruído na medição de velocidade.

Máxima Variação da Ação de Controle [Volts]	Erro Médio Velocidade [rad/s]		Erro Máximo Velocidade [rad/s]		Variação Média Ação de Controle [Volts]		Variação Máxima Ação de Controle [Volts]	
	DI	PI	DI	PI	DI	PI	DI	PI
-	1,0547	1,7119	4,1332	4,0434	735,9511	22,7913	3244,744	78,6008
700,0	1,5658	1,7119	13,2122	4,0434	384,9203	22,7913	700,0000	78,6008
70,0	4,5205	1,6945	15,4594	4,3658	68,6308	21,7775	70,0000	70,0000
17,5	1,4791	1,9106	5,9609	6,1304	17,1676	15,5486	17,5000	17,5000

A Tabela 4.3 apresenta dados para análise comparativa entre o controle da velocidade do motor CC utilizando o PI e a dinâmica inversa onde ocorre a inversão do torque de carga a cada dois segundos. Aqui nota-se grande diferença do erro médio e máximo de velocidade do

PI com o controle utilizando a dinâmica inversa. Como a dinâmica inversa possui a capacidade de previsão e utiliza-se a corrente como uma entrada do modelo, o controlador consegue fazer com que o erro diminua consideravelmente em relação ao PI porque a corrente é uma ótima informação sobre o conjugado de carga.

Tabela 4.3 – Comparação entre o PI e o controle baseado na dinâmica inversa na condição de inversão do torque de carga.

Máxima Variação da Ação de Controle [Volts]	Erro Médio Velocidade [rad/s]		Erro Máximo Velocidade [rad/s]		Variação Média Ação de Controle [Volts]		Variação Máxima Ação de Controle [Volts]	
	DI	PI	DI	PI	DI	PI	DI	PI
-	0,3088	2,4715	8,2005	13,5427	121,8840	39,0557	3587,174	268,2164
700,0	1,1113	2,4715	12,4564	13,5427	137,8634	39,0557	700,0000	268,2164
70,0	5,4988	9,9021	17,1182	35,6699	69,1353	66,9580	70,0000	70,0000
17,5	2,9017	7,4744	24,0856	25,3358	17,1603	16,9613	17,5000	17,5000

4.2.2 Simulações com treinamento on-line

Deseja-se que o controlador dinâmico inverso possua a característica de ser “inteligente”. O controlador “inteligente” possui a capacidade de absorver conhecimento sobre o processo ao qual está controlando. Depois que o controlador reter o conhecimento sobre o processo, não é mais necessário treinar a rede e o controle é efetuado com sucesso. Isso acontece com o controlador que utiliza somente o treinamento off-line, ou então quando se utiliza treinamento on-line por certo período e depois não se treina mais a rede. Essa característica é um dos motivos que faz com que o uso desse controlador seja interessante. Pensando dessa maneira, algumas características do controle com treinamento on-line devem ser consideradas.

As seguintes características do treinamento on-line podem ser observadas [*Parma (2000)*]:

- Não possui, a priori, os vetores de dados com os quais a rede deve ser treinada;
- Os vetores de treinamento não são, na sua maioria, armazenados;
- O fator tempo tem que ser considerado como elemento seqüenciador dos vetores de treinamento, os quais no treinamento off-line são, geralmente, apresentados de forma aleatória.

Deve-se levar em consideração, no treinamento on-line, o efeito destrutivo sobre o prévio treinamento da rede durante a apresentação contínua de um mesmo vetor de treinamento, fato que pode ocorrer durante uma operação em regime estacionário do sistema [*Parma (2000)*].

Como, a princípio, não se armazena os vetores de dados de treinamento é necessário utilizar o método do gradiente para realizar o treinamento da rede. O treinamento off-line com o método do gradiente apresentou resultados bem inferiores ao método dos mínimos quadráticos, compare a validação dos treinamentos através da Figura 4.4 e Figura 4.5. Devido a isso, não se pode esperar que a rede que já foi treinada (off-line) com o método dos mínimos quadráticos possa melhorar a função inversa da planta com treinamento on-line utilizando o gradiente.

Outro ponto a se observar é que dificilmente se conseguirá obter um melhor modelo da função inversa com treinamentos on-line do que o modelo com treinamento off-line através do gradiente. Considerando isso, os resultados com treinamentos on-line, a princípio, também não poderão ser melhores dependendo do tipo de trajetória de referência.

Devido a esses pontos destacados, não se conseguirá utilizar o controlador com a característica de ser “inteligente” e sim com a característica de ser adaptativo e desta maneira o controlador nunca poderá ter o seu treinamento interrompido.

Normalmente no treinamento da rede com o método do gradiente deseja-se minimizar o erro quadrático entre o valor desejado e o valor de saída da rede. No treinamento on-line com o método do gradiente, utilizasse como função de custo a ser minimizada o erro quadrático entre a referência e o valor de saída da planta ao invés da função citada anteriormente.

A Figura 4.17 apresenta o controle de velocidade do motor CC utilizando somente o treinamento on-line. Nota-se que os resultados são bem inferiores quando comparado com o controle com treinamento off-line na mesma condição, veja Figura 4.6.

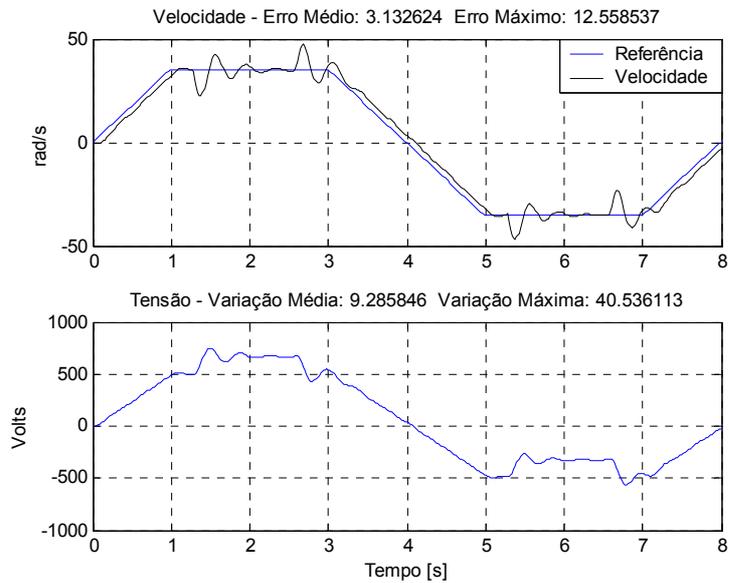


Figura 4.17 – Controle da velocidade do motor CC com aplicação e retirada de carga.

A Figura 4.18 apresenta o controle de velocidade do motor CC utilizando somente o treinamento on-line para uma trajetória em senóide. Nota-se que o erro entre a velocidade e a referência aumenta ao passar do tempo. É evidente aqui que o treinamento on-line não consegue obter a função inversa da planta. A tensão aplicada no motor varia muito e não é possível obter o valor ideal para ser aplicado no motor a cada iteração.

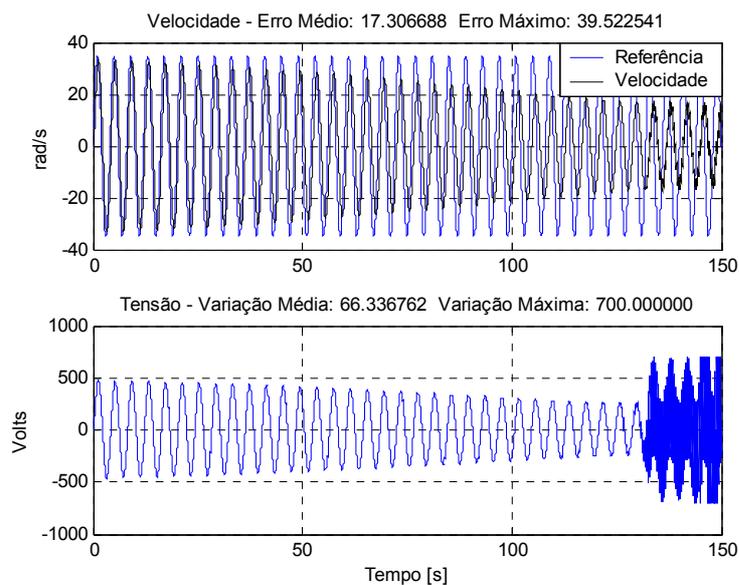


Figura 4.18 – Controle da velocidade do motor CC com treinamento on-line.

A senóide que é a referência de velocidade contribui muito para a não obtenção de bons resultados porque faz com que sempre se tenha erro e com isso sempre é necessário realizar o treinamento da rede. Em contra partida, a senóide de referência contribui para a apresentação de dados que faz com que o motor passe por vários estados que é interessante para o treinamento com vetor de dados.

A trajetória de referência apresentada na Figura 4.17 não apresentará o problema de instabilidade da tensão aplicada no motor para aquele período de tempo porque não é preciso ajustar demasiadamente os pesos da rede.

Um problema que se deve preocupar em relação ao treinamento da rede é que não se pode ficar treinando a rede a toda iteração porque isso pode levar a um efeito destrutivo do prévio treinamento da mesma. O ruído na medição faz com que se tenha sempre erro e isso leva ao ajuste dos pesos da rede. É importante que o ganho de treinamento seja ajustado em relação ao erro médio porque diminui a variância do ruído, o que faz com que esteja trabalhando com o sinal mais próximo do sinal sem ruído. A equação (4.5) apresenta a função de ajuste da taxa de aprendizado proposta em função do erro ou erro médio. Observa-se na Figura 4.19 a visualização gráfica dessa função. O parâmetro **A** mostra onde o ganho de treinamento será igual a um. O parâmetro **B** está relacionado com a concavidade da função, quanto maior **B** maior será a concavidade.

$$alfa(x) = \min \left(AlfaMax, abs \left(\left(\frac{x}{A} \right)^{2B} \right) \right) \quad (4.5)$$

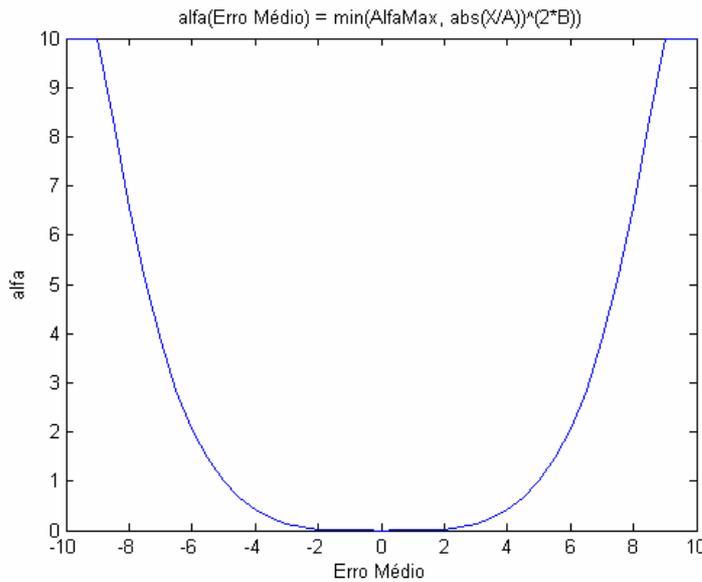


Figura 4.19 – Função do ganho de treinamento em função do erro médio com o $A = 5$ e $B = 2$.

Uma vantagem de ter uma trajetória de referência não estacionária, como é o caso da senóide, é que isso faz com que a planta passe por vários estados que pode ajudar na obtenção da função inversa. Assim, pode-se optar por uma estratégia de controle com método de treinamento híbrido utilizando os métodos do gradiente e dos mínimos quadráticos. Para isso, é necessário armazenar os vetores de dados para treinar a rede de tempos em tempos com o método dos mínimos quadrados e em toda iteração utilizar o método do gradiente para realizar o treinamento da rede. É importante que os dados armazenados sejam de padrões distintos para poder extrair do mesmo o modelo inverso da planta. Para separar os padrões, alguns métodos de separação de cluster podem ser utilizados como, por exemplo, o **kmeans** [Spath (1985)]. Dessa maneira, estará utilizando padrões que representem bem o sistema e não estará apresentando padrões que estão praticamente em um único estado.

A Figura 4.20 apresenta o controle da velocidade do motor utilizando essa estratégia de treinamento onde os padrões são separados com o **kmeans**. Observa-se que no tempo próximo de três segundos, tempo em que foi executado o método dos mínimos quadrados para o treinamento, o erro diminui consideravelmente.

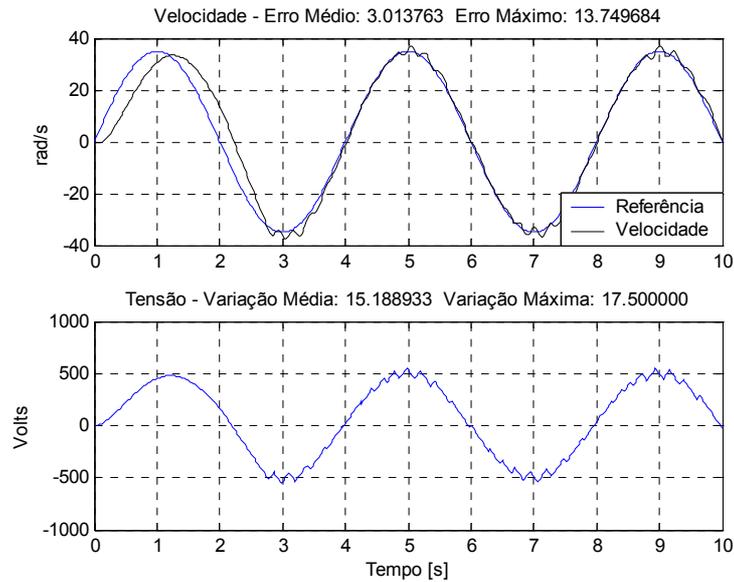


Figura 4.20 – Controle da velocidade do motor CC utilizando treinamento híbrido.

Uma desvantagem dessa estratégia é que ao realizar o treinamento através dos mínimos quadráticos, os pesos da rede são totalmente alterados em função do vetor de dados. Assim, pode acontecer que a nova função inversa seja pior do que a função inversa que estava sendo treinada com o método do gradiente. Isto acontece quando o vetor de dados não for bem representativo da planta.

4.3 Conclusões

Neste capítulo foram apresentados os resultados de simulações de controle de velocidade do motor CC utilizando a dinâmica inversa. Mostrou-se que a obtenção do modelo artificial da função que mapeia a dinâmica inversa da planta não é feita de forma simples e que alguns fatores dificultam ainda mais a obtenção da mesma como, por exemplo, a super amostragem do sinal. A obtenção do modelo inverso que mapeia a dinâmica inversa da planta é ainda mais difícil de ser feita no treinamento on-line, o que prejudica bastante o controle em malha fechada.

O controle utilizando a dinâmica inversa apresentou a característica de rejeitar bem as perturbações de carga, seguir trajetórias não estacionárias com erro pequeno ou nulo e variar bastante a ação de controle em situações que gerem erro (ruído na medição, torque de carga).

Os resultados do controlador utilizando a dinâmica inversa foram comparados com o controlador PI onde seus parâmetros foram encontrados através do método de otimização do Algoritmo Genético. A principal vantagem sobre o PI foi com relação à rejeição às perturbações de carga e por apresentar um erro muito menor para seguir trajetórias não estacionárias, como a rampa e a senóide. Uma desvantagem observada em relação ao PI foi que a ação de controle varia bastante quando há ruído na medição de velocidade ou quando se aplica torque de carga.

No capítulo seguinte será apresentada uma proposta simples de como obter a função inversa da planta de forma indireta.

Capítulo 5

Controle Dinâmico Inverso

Aproximado

Neste capítulo será apresentada uma proposta de controle de sistemas utilizando o modelo inverso aproximado de uma planta a cada iteração. Serão mostradas as dificuldades ao utilizar essa estratégia de controle. Esse controlador será utilizado para controlar a velocidade de um motor CC e dois sistemas não lineares.

5.1 Introdução

Controlar um processo através da dinâmica inversa da planta é tentador, pois é uma forma bem simples de se obter um controlador onde não precisaria ficar ajustando ganhos de controladores que muitas vezes são feitas de forma empírica e resulta em processos mal controlados. Utilizando dados históricos do processo, a princípio, é possível treinar uma rede para extrair o modelo inverso da planta e assim projetar facilmente esse controlador. Mas como mostrado no capítulo anterior, existe algumas dificuldades em se obter o modelo que mapeia a inversa da planta, mesmo utilizando técnicas de inteligência computacional.

Devido às dificuldades apresentadas em se implementar um controlador que utilize a dinâmica inversa da planta, principalmente quando se fecha a malha através de treinamentos on-line, uma proposta de aproximar a dinâmica inversa da planta é apresentada neste capítulo.

A aproximação da dinâmica inversa é feita utilizando o modelo da planta, que é mais fácil de ser obtido do que o modelo inverso. Por utilizar o modelo da planta ao invés do modelo inverso, esse controlador consegue obter indiretamente um melhor modelo inverso da planta e com o benefício de se adaptar ao processo mais facilmente. Assim, o esforço de sintonia do controlador é diminuído consideravelmente. Além disso, é possível obter melhores resultados com esse controlador do que o controlador dinâmico inverso, principalmente quando se fecha a malha de controle através de treinamentos on-line..

A Figura 5.1 apresenta o diagrama de blocos do controlador aproximando a função inversa a cada iteração. O Controlador Dinâmico Inverso Aproximado é formado por um aproximador da inversa e pelo modelo da planta. O aproximador da inversa utiliza o modelo da planta para encontrar a ação de controle desejada para se obter a referência no instante seguinte. O modelo da planta pode ser ajustado ao longo do controle do processo (on-line) permitindo, assim, fechar a malha de controle através de uma realimentação indireta já que o erro não é uma entrada do controlador. Uma entrada do modelo deve ser a ação de controle que é aplicada na planta e têm-se também outras entradas para ajudar na obtenção do mesmo. Embora o modelo da planta não necessite ser encontrado através de técnicas de inteligência computacional, utilizar técnicas de inteligência computacional possibilita encontrá-lo de forma mais fácil, assim como o ajuste do mesmo ao longo do controle do processo.

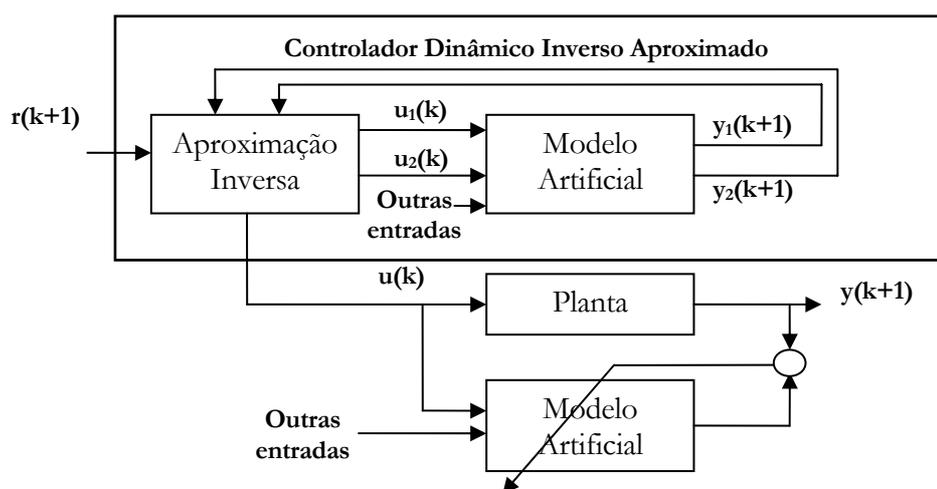


Figura 5.1 – Diagrama de blocos do controlador aproximando a função inversa a cada iteração.

Pode-se ver na Figura 5.2 como é feito a escolha da ação de controle em função da referência. Aplicam-se na rede as outras entradas e na entrada que representa a ação de controle aplica-se $u_1(k)$ e obtém como saída da rede $y_1(k+1)$. O mesmo é feito com a outra ação de controle $u_2(k)$ e tem-se através da rede como saída $y_2(k+1)$. Através de uma equação de uma reta se calcula qual é o valor estimado de $u(k)$ que deve ser aplicado para que se tenha na saída da planta $r(k+1)$. O que está sendo feito é encontrar uma reta que represente a inversa da planta naquele momento. Como se tem uma reta, a inversa é obtida facilmente.

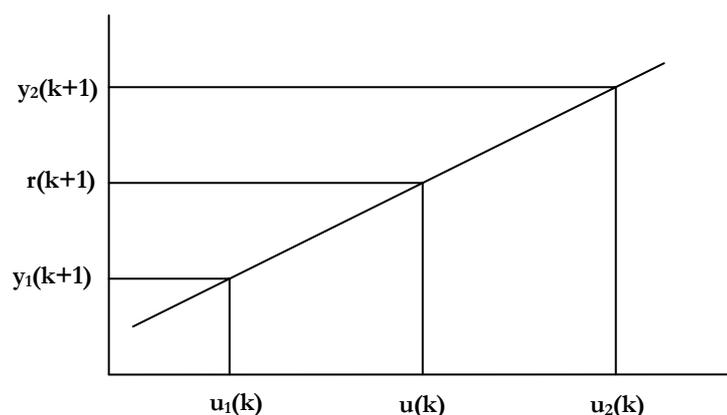


Figura 5.2 – Gráfico demonstrando a escolha da ação de controle.

É necessário que $u_1(k)$ e $u_2(k)$ tenham valores tais que quando aplicado no modelo obtenha-se $y_1(k+1)$ diferente de $y_2(k+1)$, pois a inversa é obtida através da equação de uma reta e caso esses valores sejam iguais, qualquer valor $u(k)$ satisfará a equação da reta. Por isso, novamente frisa-se que é importantíssimo que a amostragem do sinal seja feita com um período de tempo adequado.

Outra forma de encontrar a ação de controle que deve ser aplicada no instante seguinte de forma a obter a referência utilizando o modelo da planta é com um método de otimização, semelhante ao controle preditivo (seção 2.3.3). Neste caso, pode-se usar o erro quadrático entre a referência e o valor de saída do modelo como função objetivo. Dessa forma, a princípio, se conseguiria encontrar a ação de controle. Assim, o método da aproximação da inversa a cada iteração pode ser visto como um método de otimização indireto de baixo custo computacional. Através do método da aproximação da inversa são utilizadas somente

duas avaliações da rede, o que provavelmente não aconteceria com qualquer outro método de otimização.

Após ser estimada a ação de controle que deve ser aplicada na planta, utiliza-se a formulação que foi apresentada na equação (4.3) para escolher a ação de controle que será aplicada na planta e que, aqui novamente, é repetida na equação (5.1) por conveniência.

$$\begin{aligned}
 & \text{Se } \tilde{u}_k \geq u_{k-1} \\
 & \quad u_k = \min(\tilde{u}_k, u_{\max}, u_{k-1} + \Delta u_{\max}) \\
 & \text{Se } \textit{n\~{a}o} \\
 & \quad u_k = \max(\tilde{u}_k, u_{\max}, u_{k-1} - \Delta u_{\max})
 \end{aligned} \tag{5.1}$$

Como descrito anteriormente no Capítulo 4, a função inversa não prevê que o processo possui limitações físicas. Através da formulação mostrada na equação (4.3) é garantido que não é aplicada nenhuma entrada acima dos limites da planta, assim como, uma variação da ação de controle maior que o permitido. Isso é necessário para garantir a estabilidade física do processo.

5.2 Controle de velocidade do motor CC

Nas simulações a seguir é utilizada a rede NFN com duas partições e utiliza-se a velocidade, tensão e corrente de armadura no instante atual e instante anterior para encontrar a velocidade do motor CC no instante seguinte, veja diagrama de blocos mostrado na Figura 5.3.

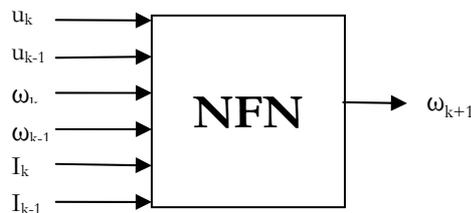


Figura 5.3 – Entradas utilizadas na rede NFN para modelar a velocidade do motor CC.

Serão apresentados resultados de simulação com treinamentos off-line e on-line. No treinamento off-line é utilizado o método dos mínimos quadráticos e no treinamento on-line o método do gradiente com taxa de aprendizado ótimo [Caminhas *et al.* (1998)].

5.2.1 Simulações com treinamento off-line

A rede NFN é treinada com o método dos mínimos quadráticos, pois garante um melhor modelo do que com o método do gradiente. Assim como no capítulo anterior, os dados utilizados no treinamento são provenientes de diversos degraus aplicados no motor de amplitude aleatória e tempo de duração também aleatório, o mesmo acontece com o conjugado de carga que é aplicado no motor, com o objetivo de extrair o máximo possível de informações para modelar o motor.

Nas simulações a seguir, a rede NFN é treinada para obter o modelo direto do motor CC primeiramente e depois é utilizada para controlar a velocidade do motor CC sem treinamento on-line. Aqui não serão mostrados os gráficos do controle de velocidade do motor utilizando o PI, já que os mesmos já foram apresentados no capítulo anterior.

A Figura 5.4 apresenta o controle da velocidade do motor CC utilizando a aproximação da inversa da planta. Assim como no controle utilizando a dinâmica inversa na mesma condição, que foi apresentado na Figura 4.6, praticamente não se vê diferença entre a referência e a velocidade, com exceção no momento onde ocorre aplicação e retirada do torque de carga, e o controlador apresenta uma boa rejeição as perturbações de carga.

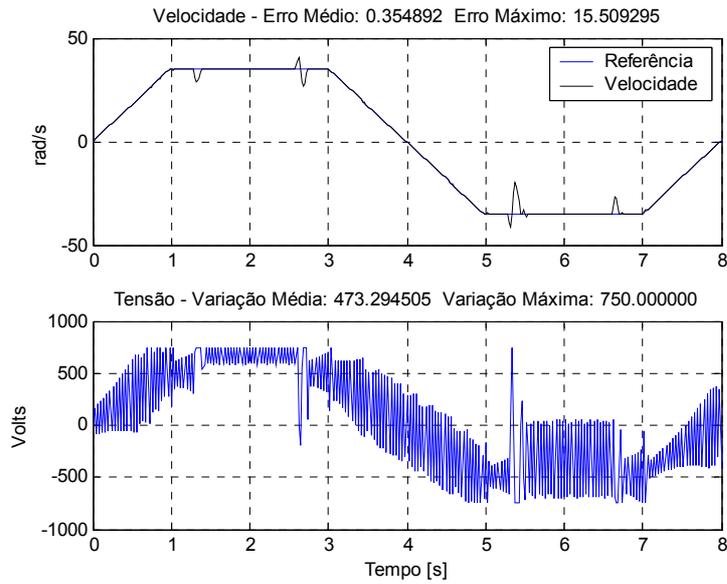


Figura 5.4 – Controle da velocidade do motor CC com aplicação e retirada de carga utilizando aproximação da inversa.

A Figura 5.5 apresenta o controle da velocidade do motor CC utilizando a aproximação da inversa da planta. No início o erro de velocidade é maior, mas depois praticamente não se vê diferença entre a referência e a velocidade, em consequência o erro médio é bem pequeno mesmo tendo uma senóide como referência. Nota-se que a ação de controle varia bastante, bem mais do que quando se utiliza o controle com a dinâmica inversa, veja Figura 4.7. Frisa-se novamente que essa ação de controle normalmente não pode ser aplicada na planta e por isso deve ser limitada pela formulação apresentada na equação (5.1), mas aqui é utilizada para evidenciar as características do controlador dinâmico inverso aproximado.

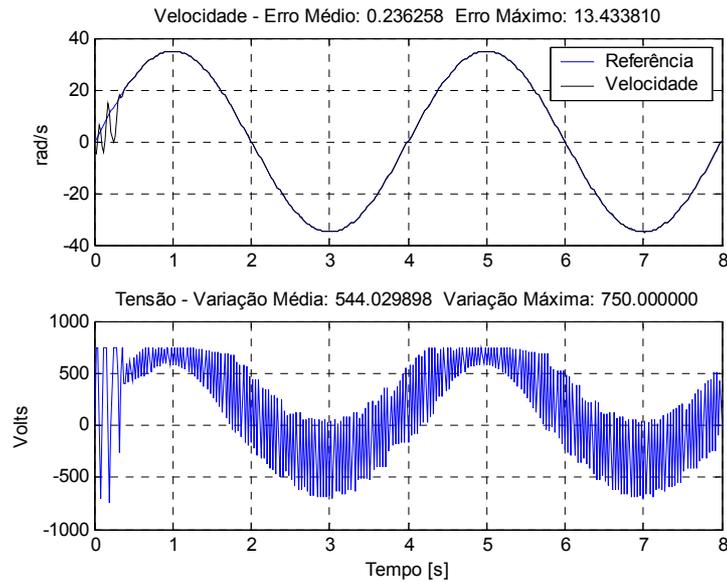


Figura 5.5 – Controle de velocidade do motor CC utilizando a aproximação da dinâmica inversa.

5.2.1.1 Inserção de ruído na medição de velocidade

A Figura 5.6 apresenta o controle da velocidade do motor CC com ruído na medição da velocidade. O ruído inserido na medição da velocidade tem média zero e Variância igual a $0,5^2$.

O erro aumentou em relação à medição sem ruído como era de se esperar. Assim como no controle utilizando a dinâmica inversa, observa-se que a ação de controle varia bastante, bem mais do que no controle sem ruído na medição, com o objetivo de se ter erro nulo de velocidade no instante seguinte. O controlador apresenta essa ação de controle com essa característica já que possui a capacidade de previsão, assim como o controlador utilizando a dinâmica inversa. Devido a essa capacidade, o controlador estima uma ação de controle que seja capaz de fazer com que se tenha erro nulo no instante seguinte, mesmo que para isso a ação de controle tenha que variar bastante, assumindo valores que não podem ser aplicados na planta e que a fonte não tem capacidade de fornecer.

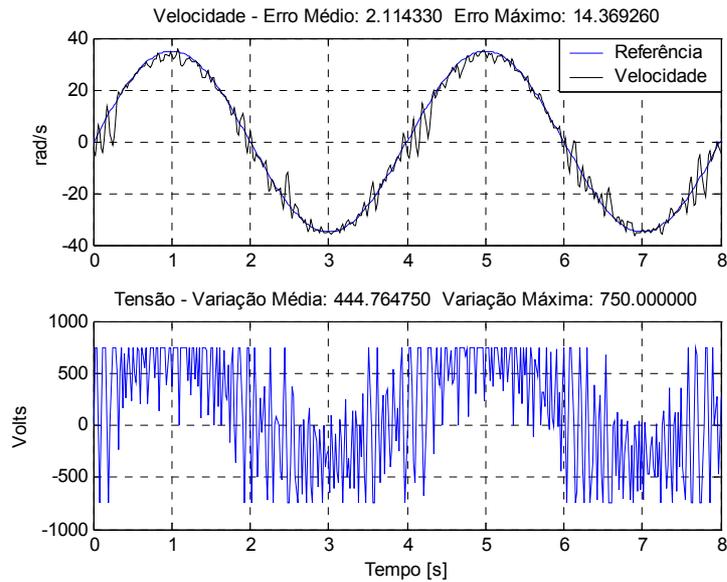


Figura 5.6 – Controle de velocidade do motor CC utilizando dinâmica inversa com ruído na medição de velocidade.

5.2.1.2 Inversão do torque de carga

A Figura 5.7 apresenta o controle da velocidade do motor CC com a inversão do torque de carga, onde o torque de carga é invertido ($Tl_t = -Tl_{t-2}$) a cada dois segundos. O Torque de carga que é aplicado no motor inicialmente é igual a 500.000 Nm.

O erro aumentou em relação à medição sem ruído como era de se esperar, veja Figura 5.5. A ação de controle que é aplicada no motor varia bastante com o objetivo de se ter erro nulo no instante seguinte. Ela varia bem mais do que a ação de controle quando se utiliza a dinâmica inversa Figura 4.11.

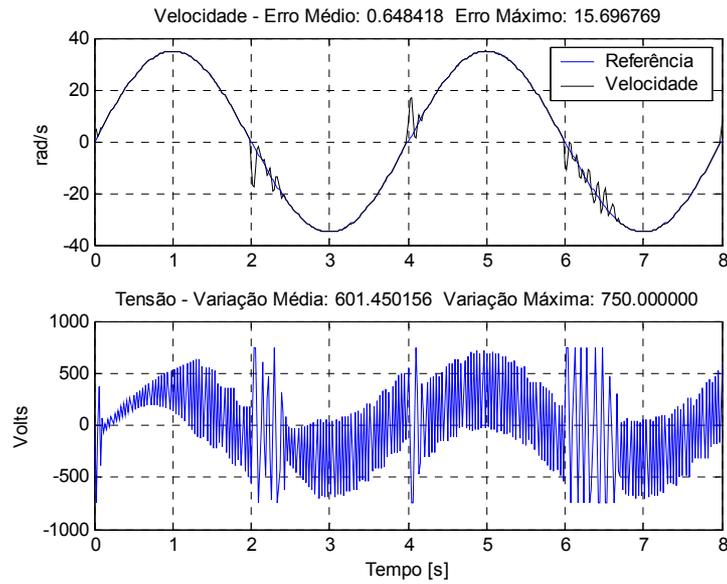


Figura 5.7 – Controle de velocidade do motor CC utilizando aproximação da dinâmica inversa com inversão de torque a cada dois segundos.

5.2.1.3 Variação da ação de controle limitada em 70 volts com ruído na medição da velocidade

A Figura 5.8 apresenta o controle da velocidade do motor com a inserção de ruído na medição da velocidade onde a variação da ação de controle é limitada em 70 volts, aproximadamente 10 vezes menor que a tensão nominal. Observa-se que a velocidade fica sempre oscilando em torno da referência.

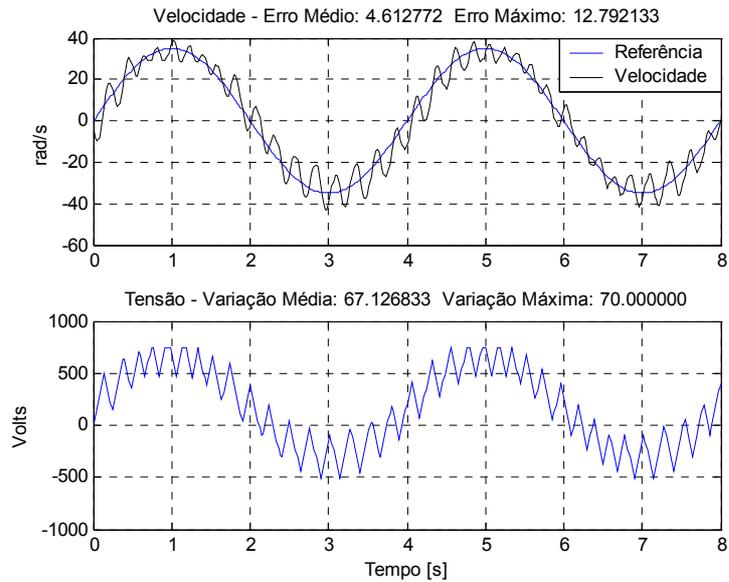


Figura 5.8 – Controle da velocidade do motor CC com ruído na medição de velocidade e com a variação da ação de controle limitada em 70 volts.

5.2.1.4 Variação da ação de controle limitada em 70 volts com inversão do torque de carga

A Figura 5.9 apresenta o controle da velocidade do motor com a inversão do torque onde a variação da ação de controle é limitada em 70 volts. Observa-se que a velocidade fica sempre oscilando em torno da referência, da mesma maneira que na condição com inserção de ruído na medição de velocidade.

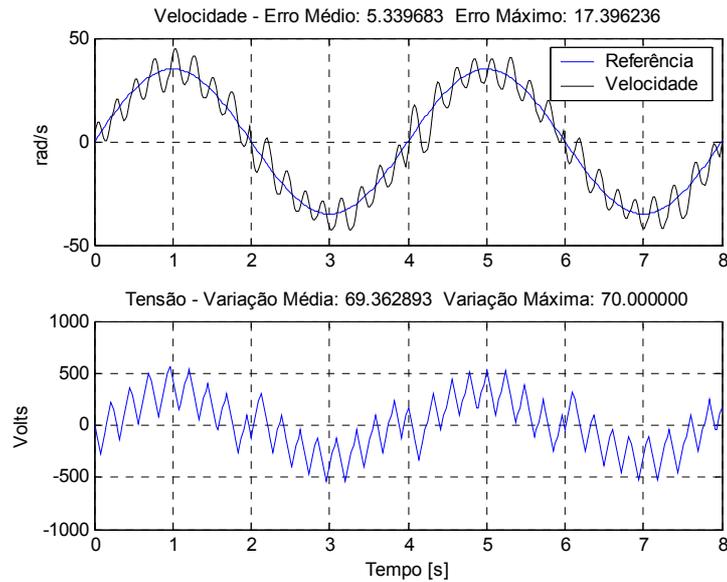


Figura 5.9 – Controle da velocidade do motor CC com inversão de torque e com a variação da ação de controle limitada em 70 volts.

5.2.1.5 Variação da ação de controle limitada em 17,5 volts com ruído na medição da velocidade

A Figura 5.10 apresenta o controle da velocidade do motor com a adição de ruído na medição de velocidade onde a variação da ação de controle é limitada em 17,5 volts, aproximadamente 40 vezes menor que a tensão nominal. Diferentemente da situação com a máxima variação de 70 volts apresentada anteriormente, a velocidade apresenta um pequeno erro e consegue convergir para a referência.

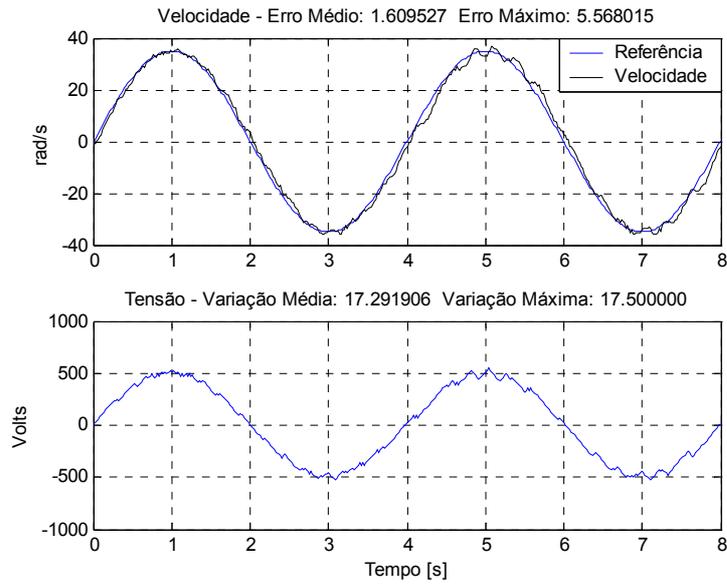


Figura 5.10 – Controle da velocidade do motor CC com ruído na medição da velocidade e com a variação da ação de controle limitada em 17,5 volts.

5.2.1.6 Variação da ação de controle limitada em 17,5 volts com inversão do torque de carga

A Figura 5.11 apresenta o controle da velocidade do motor com a inversão do torque onde a variação da ação de controle é limitada em 17,5 volts. Diferentemente da situação com a máxima variação de 70 volts apresentada anteriormente, a velocidade apresenta um pequeno erro em relação à referência, conseguindo convergir.

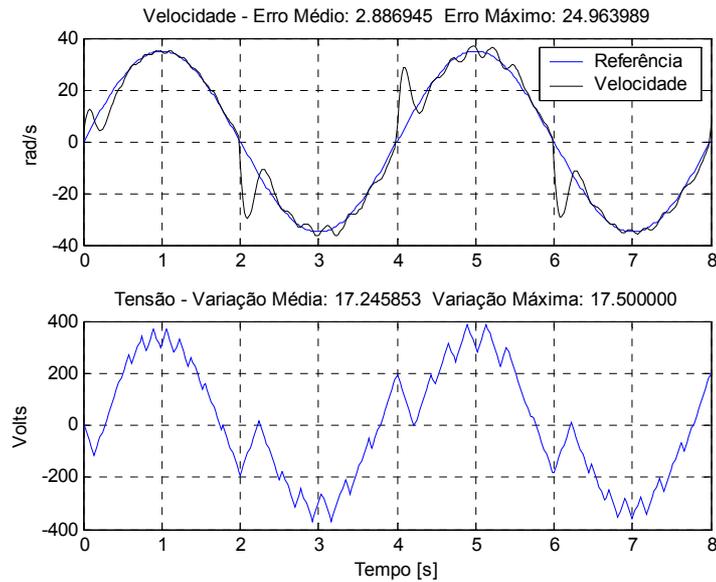


Figura 5.11 – Controle da velocidade do motor CC com inversão de torque e com a variação da ação de controle limitada em 17,5 volts.

5.2.2 Simulações com treinamento on-line

O treinamento on-line da rede é importantíssimo para se poder efetuar o controle do processo em malha fechada. Como visto no capítulo anterior sobre a dinâmica inversa com treinamento on-line, o controlador apresentou algumas dificuldades em controlar a velocidade do motor. O principal motivo se deve ao fato que a obtenção do modelo inverso não é fácil, principalmente quando se utiliza treinamentos on-line. Como o modelo artificial da planta é mais fácil de obter do que o modelo artificial da função inversa, espera-se que se tenha maior sucesso no controle em malha fechada do processo.

Nas simulações apresentadas a seguir usa-se a mesma estrutura da rede que é utilizada nas simulações com treinamento off-line. A rede é treinada a cada iteração de controle utilizando 0,1 como a taxa de aprendizado ótimo. A rede não possui nenhum conhecimento prévio sobre o processo no qual está controlando.

Na Figura 5.12 é exibida a simulação do controle da velocidade do motor CC utilizando a aproximação da inversa com treinamentos on-line para trajetória em rampa com inversão de velocidade e com aplicação e retirada de carga. Nota-se que no início do controle tem-se erro

grande de velocidade e que ao passar o tempo o mesmo vai diminuindo progressivamente. Quando se compara o controle de velocidade do motor na mesma situação na dinâmica inversa, apresentado na Figura 4.17, observa-se que a velocidade fica oscilando em torno da referência e que a ação de controle varia bem mais no controle utilizando a aproximação da inversa.

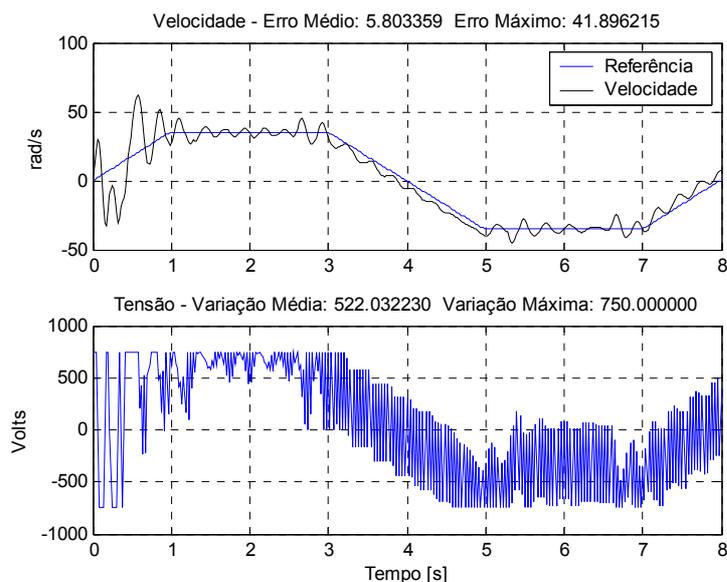


Figura 5.12 – Controle da velocidade do motor CC com aplicação e retirada de carga utilizando aproximação da inversa.

Na Figura 5.13 é exibida a simulação do controle da velocidade do motor CC utilizando a aproximação da inversa com treinamentos on-line. Nota-se que no início do controle tem-se erro grande de velocidade e a ação de controle varia bastante. Ao passar o tempo, o erro vai diminuindo progressivamente assim como a variação da ação de controle. Quando se compara o controle de velocidade do motor na mesma situação, mas com treinamento off-line da rede apresentado na Figura 5.5, observa-se que a ação de controle não varia tanto, parte final do controle, quanto no treinamento off-line. O controlador poderia utilizar uma variação grande da ação de controle com o objetivo de se ter erro nulo no instante seguinte. Um dos motivos da variação da ação de controle ser menor é porque no treinamento on-line o modelo da planta está representando melhor o ponto de operação no qual está trabalhando, em contrapartida, o modelo não generaliza bem a planta para todos os seus estados.

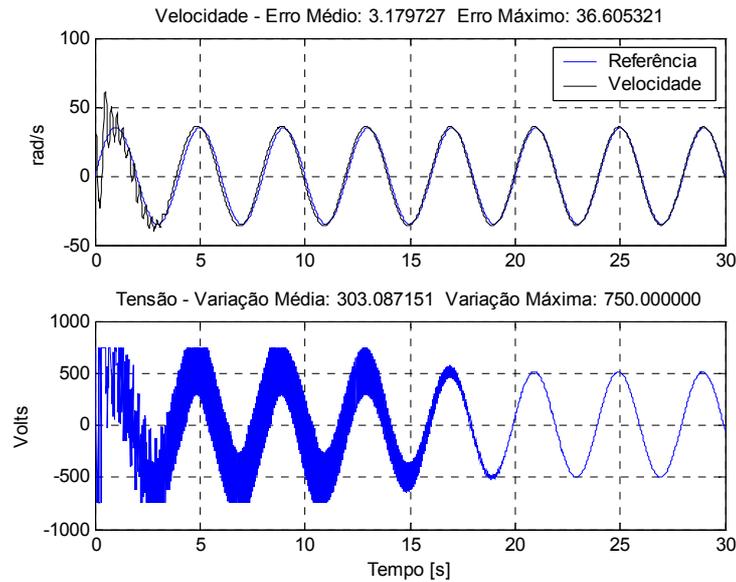


Figura 5.13 – Controle da velocidade do motor CC utilizando aproximação da inversa.

5.2.2.1 Inserção de ruído na medição de velocidade

A Figura 5.14 exibe o controle da velocidade do motor CC utilizando a aproximação da inversa com treinamento on-line onde é inserido ruído na medição de velocidade. Nota-se que assim como no controle sem ruído na velocidade, que no início do controle tem-se erro grande de velocidade, a ação de controle varia bastante devido ao não conhecimento prévio sobre a planta. Ao passar o tempo, o erro vai diminuindo progressivamente assim como a variação da ação de controle, já que a rede vai adquirindo conhecimento quando é feito o controle. Comparando com o controle apresentado na Figura 5.6 onde se utilizou treinamento off-line da rede, observa-se que a ação de controle não varia tanto ao passar o tempo quanto no treinamento off-line, mesmo com o controlador podendo utilizar uma variação grande da ação de controle com o objetivo de se ter erro nulo no instante seguinte.

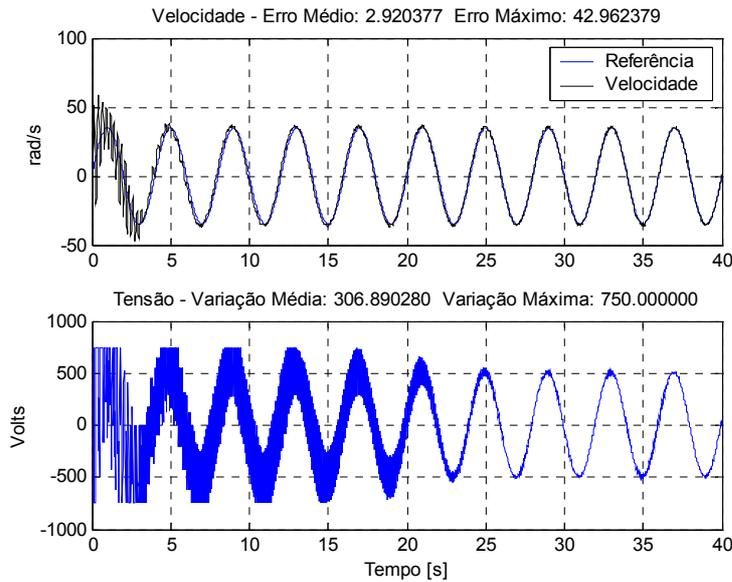


Figura 5.14 – Controle da velocidade do motor CC utilizando aproximação da inversa com ruído na medição de velocidade.

5.2.2.2 Inversão do torque de carga

Nas simulações a seguir é aplicado torque de carga no motor, onde o torque de carga é invertido ($Tl_t = -Tl_{t-2}$) a cada dois segundos. O Torque de carga que é aplicado no motor inicialmente é igual a 500.000 Nm.

A Figura 5.15 apresenta o controle da velocidade do motor CC com a inversão do torque. Pode-se observar que o erro diminui progressivamente ao passar o tempo, o que não acontece com a variação da ação de controle que continua variando muito. Aqui diferentemente da situação onde se tem ruído na medição de velocidade e na situação sem ruído, a variação da ação de controle não diminui progressivamente, ela praticamente fica constante ao longo do tempo. Comparando com o controle off-line com a inversão do toque de carga apresentado na Figura 5.5 a variação da ação de controle é bem maior, o que não aconteceu na situação sem ruído e com ruído comparando com o treinamento on-line e off-line.

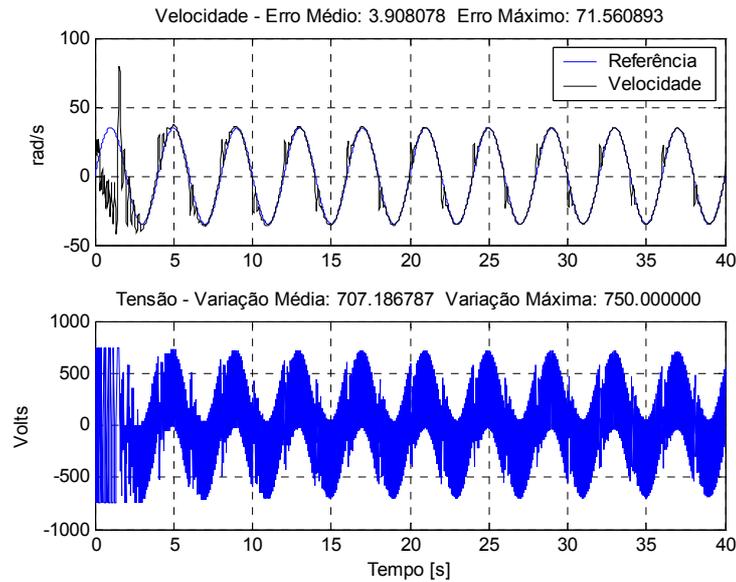


Figura 5.15 – Controle da velocidade do motor CC utilizando aproximação da inversa com inversão do torque de dois em dois segundos.

5.2.2.3 Variação da ação de controle limitada em 70 volts com ruído na medição da velocidade

A Figura 5.16 apresenta o controle da velocidade do motor com ruído na medição de velocidade com a variação da ação de controle limitada em 70 volts. Nota-se que no início do controle tem-se erro grande de velocidade e a ação de controle varia bastante. Ao passar o tempo, o erro vai diminuindo progressivamente assim como a variação da ação de controle. Diferentemente do controle efetuado com treinamento off-line na mesma condição, mesma limitação da ação de controle e ruído na medição de velocidade, apresentado na Figura 5.8 o controle da velocidade do motor converge. Observa-se ainda que a variação média da ação de controle é bem menor no controle on-line.

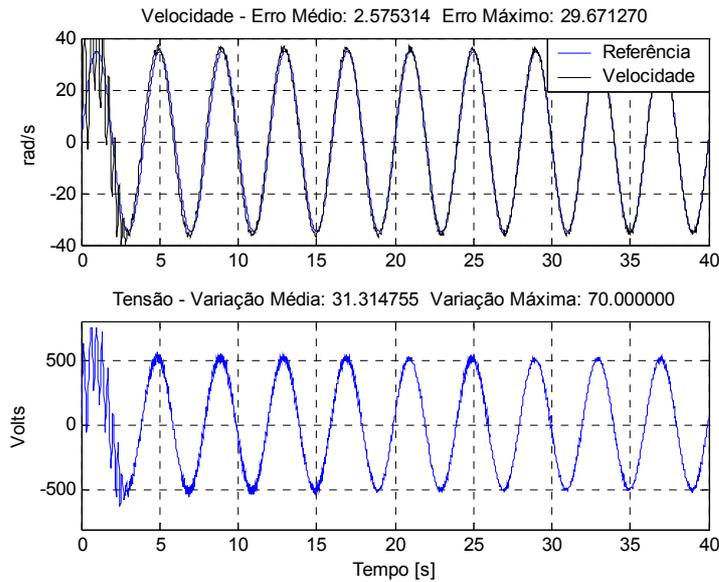


Figura 5.16 – Controle da velocidade do motor CC com ruído na medição de velocidade e com a variação da ação de controle limitada em 70 volts.

5.2.2.4 Variação da ação de controle limitada em 70 volts com inversão do torque de carga

A Figura 5.17 apresenta o controle da velocidade do motor com inversão do torque de carga com a variação da ação de controle limitada em 70 volts. Nota-se que no início do controle tem-se erro grande de velocidade e a ação de controle varia bastante. Ao passar o tempo, o erro vai diminuindo progressivamente assim como a variação da ação de controle. Diferentemente do controle efetuado com treinamento off-line na mesma condição, mesma limitação da ação de controle e inversão do torque de carga, apresentado na Figura 5.9 o controle da velocidade do motor converge. Observa-se ainda que a variação média da ação de controle é bem menor no controle on-line.

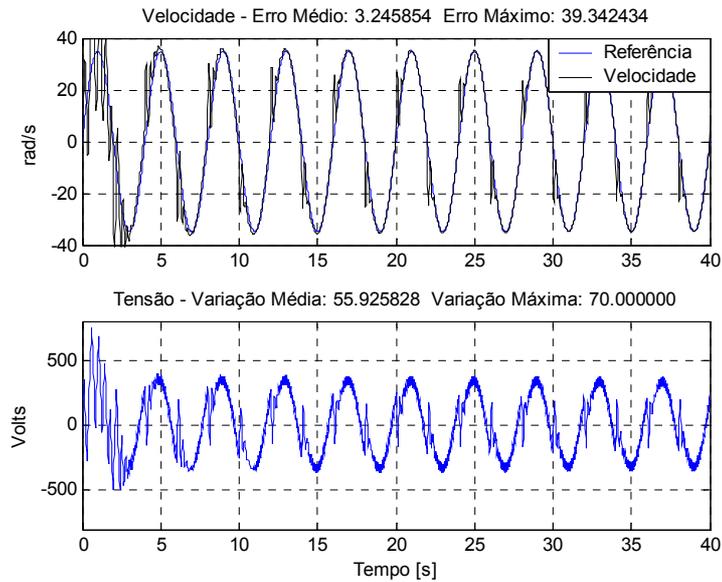


Figura 5.17 – Controle da velocidade do motor CC com inversão do torque de carga e com a variação da ação de controle limitada em 70 volts.

5.2.2.5 Variação da ação de controle limitada em 17,5 volts com ruído na medição da velocidade

A Figura 5.18 apresenta o controle da velocidade do motor com ruído na medição de velocidade com a variação da ação de controle limitada em 17,5 volts. Nota-se que no início do controle tem-se erro grande de velocidade e a ação de controle varia bastante. Ao passar o tempo, o erro vai diminuindo progressivamente assim como a variação da ação de controle. O controle efetuado com treinamento off-line na mesma condição, mesma limitação da ação de controle e ruído na medição de velocidade, apresentado na Figura 5.10 apresenta resultados semelhantes. A diferença entre os dois é que no início do controle com treinamento off-line existe um erro maior de velocidade devido ao não conhecimento sobre a planta.

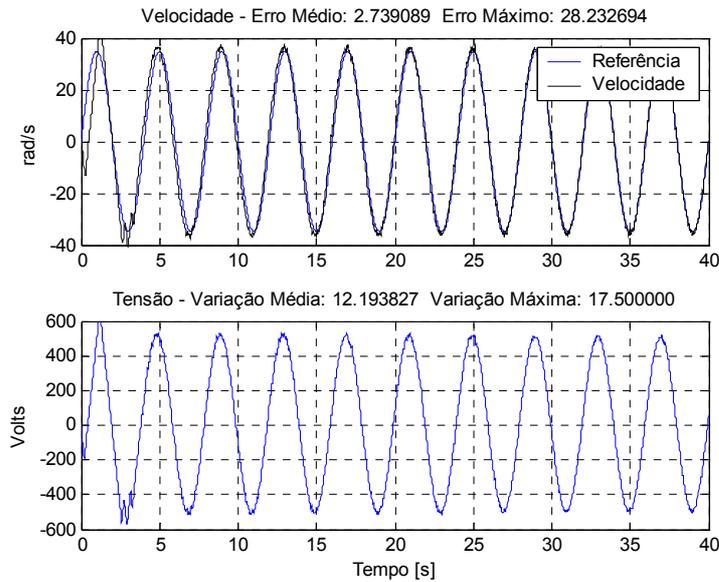


Figura 5.18 – Controle da velocidade do motor CC com ruído na medição de velocidade e com a variação da ação de controle limitada em 17,5 volts.

5.2.2.6 Variação da ação de controle limitada em 17,5 volts com inversão do torque de carga

A Figura 5.19 apresenta o controle da velocidade do motor com inversão do torque de carga com a máxima limitação da ação de controle em 17,5 volts. Nota-se que no início do controle tem-se erro grande de velocidade e a ação de controle varia bastante. Ao passar o tempo, o erro vai diminuindo progressivamente assim como a variação da ação de controle. O controle efetuado com treinamento off-line na mesma condição, mesma limitação da ação de controle e ruído na medição de velocidade, apresentado na Figura 5.11 apresenta resultados semelhantes. A diferença entre os dois é no início do controle com treinamento off-line, onde existe um erro maior de velocidade devido ao não conhecimento prévio sobre a planta.

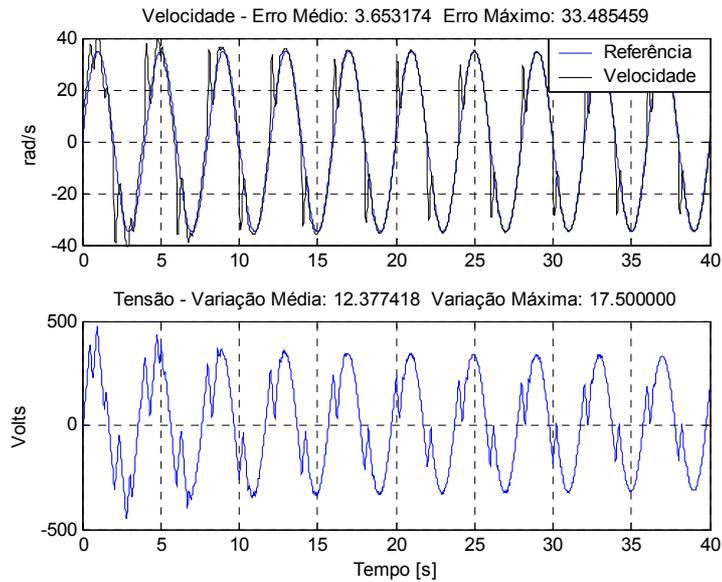


Figura 5.19 – Controle da velocidade do motor CC com inversão do torque de carga e com a variação da ação de controle limitada em 17,5 volts.

5.2.2.7 Interrupção do treinamento da rede com ruído na medição de velocidade

Serão apresentadas simulações nessa seção e na seguinte com o intuito de observar se o controle utilizando a aproximação da inversa possui a capacidade de retenção de conhecimento sobre a planta enquanto se realiza o controle do processo. Para isso, o treinamento on-line da rede será interrompido durante o controle da velocidade do motor CC.

A Figura 5.20 apresenta o controle da velocidade do motor CC com inserção de ruído na medição de velocidade, onde o treinamento on-line da rede foi interrompido no tempo $t = 60$ segundos. Observa-se que, nessa situação, a rede conseguiu adquirir o conhecimento sobre o processo no qual está controlado, mesmo utilizando o método do gradiente como forma de treinamento da rede e os dados serem apresentados de forma contínua. A ação de controle é suave não apresentando grandes variações no intervalo de tempo entre 56 e 64 segundos, mesmo podendo variar muito devido à baixa limitação que é imposta na ação de controle do controlador.

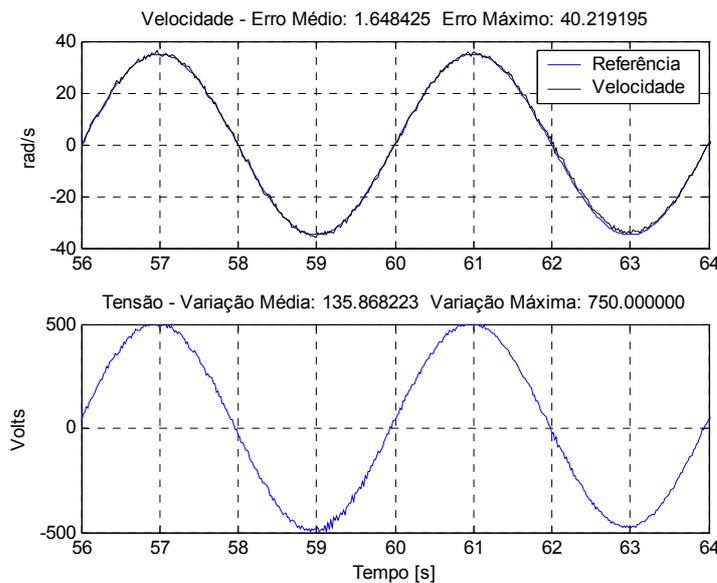


Figura 5.20 – Controle da velocidade do motor CC com treinamento on-line interrompido em $t = 60$ segundos.

Como o controlador mostrou que tem a capacidade de absorver conhecimento sobre o processo, pode-se e deve-se fazer com que o ganho de treinamento seja adaptativo e que diminua em função, por exemplo, do erro. Isso porque o treinamento contínuo da rede não é interessante, pois pode ocasionar uma perda do prévio treinamento da rede durante a apresentação contínua de um mesmo vetor de treinamento, isso pode ocorrer durante uma operação em regime estacionário do sistema. A equação (4.5) que foi proposta no Capítulo 4 como uma forma de se ajustar o ganho de treinamento em função do erro ou erro médio, também pode ser utilizada aqui para se ajustá-lo.

5.2.2.8 Interrupção do treinamento da rede com inversão do torque de carga

Nesta seção é apresentada a interrupção do controle da velocidade do motor CC com inversão do torque de carga. Na Figura 5.21 tem-se a interrupção do treinamento on-line no tempo $t = 60$ segundos. Pode-se observar que a rede não conseguiu adquirir um bom conhecimento sobre a planta, já que o controle vinha sendo executado com erro baixo até a interrupção do treinamento e logo depois, o erro aumentou. Nota-se que a parte negativa da

velocidade apresenta erro pequeno, o que não acontece na parte positiva da velocidade. Isso se deve ao fato que a rede reteve conhecimento sobre os últimos dados apresentados no treinamento serem referentes à parte negativa da velocidade. O mesmo acontece no controle de velocidade apresentado na Figura 5.22 onde o treinamento foi interrompido no tempo $t = 62$ segundos para a parte positiva do treinamento. Como os últimos dados apresentados para o treinamento da rede foram referentes à parte positiva obteve-se erro maior na parte negativa da velocidade.

Diferentemente da situação onde se tem inserção de ruído na medição de velocidade, a inversão do torque de carga faz com que a planta passe por estados bem distintos e isso faz com que o treinamento não consiga generalizar bem o modelo da planta, principalmente porque os dados são apresentados de forma contínua.

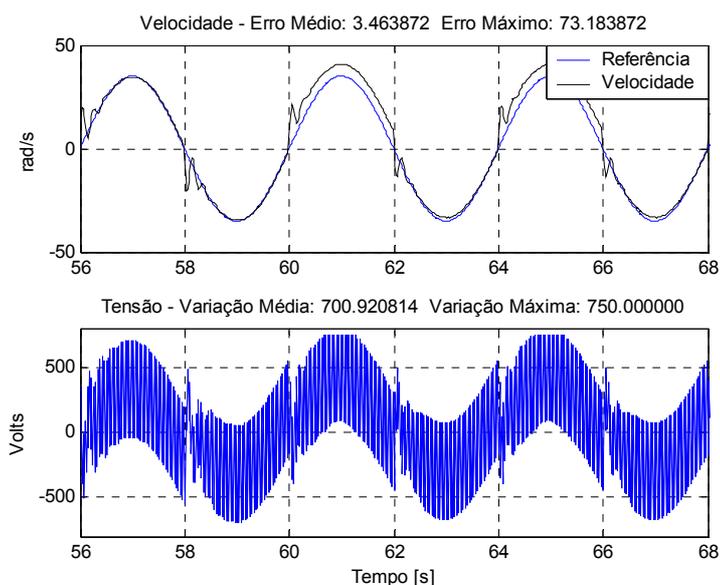


Figura 5.21 – Controle da velocidade do motor CC com inversão do torque de carga onde o treinamento on-line foi interrompido em $t = 60$ segundos.

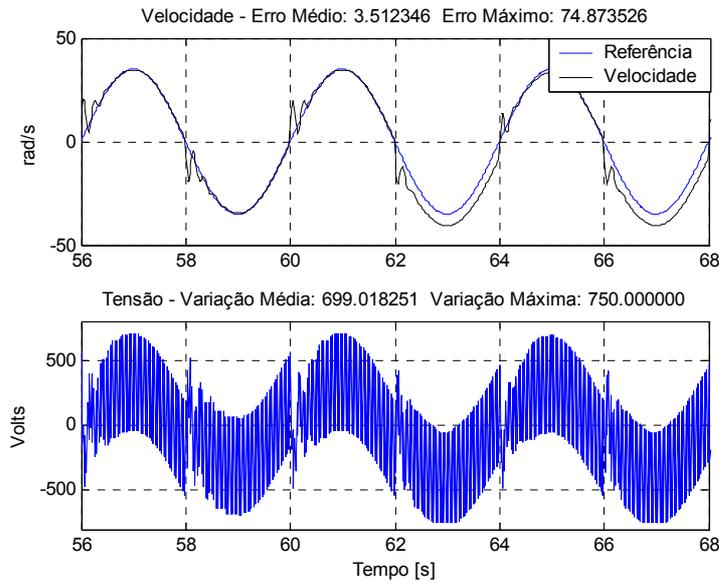


Figura 5.22 – Controle da velocidade do motor CC com inversão do torque de carga onde o treinamento on-line foi interrompido em $t = 62$ segundos.

Um dos motivos da rede não conseguir absorver conhecimento sobre o processo poderia ser devido à interrupção prematura do treinamento. Na Figura 5.23 tem-se a interrupção do treinamento no tempo $t = 100$ segundos. Observa-se que a velocidade fica oscilando em torno da referência mesmo antes da interrupção e o erro na parte positiva da velocidade diminui um pouco em relação ao erro apresentado na Figura 5.21 embora ele oscile bem mais.

O efeito da oscilação da velocidade em torno da referência se deve ao fato que a rede está sobre treinada e também porque a limitação sobre a variação da ação de controle ser baixa. Nas próximas duas seções será estudado esse efeito colocando o controlador para trabalhar durante um grande tempo.

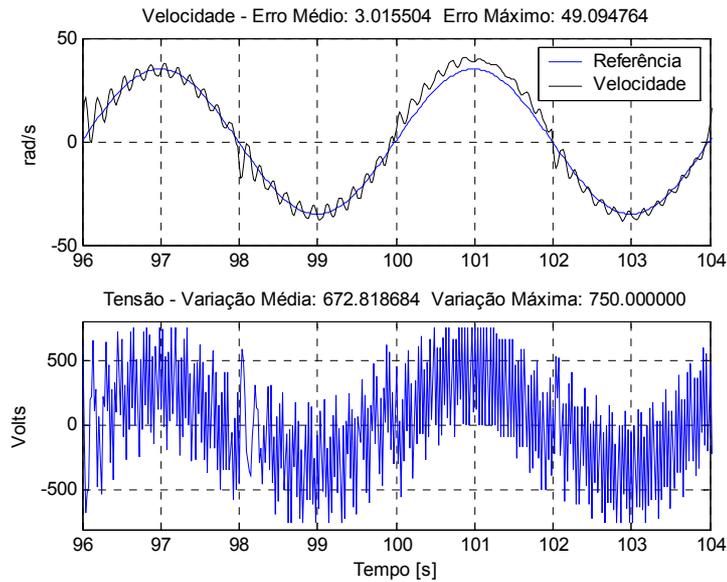


Figura 5.23 – Controle da velocidade do motor CC com inversão do torque de carga onde o treinamento on-line foi interrompido em $t = 100$ segundos.

5.2.2.9 Controle por um longo período com inserção de ruído na medição de velocidade

Como descrito anteriormente, o treinamento on-line possui algumas características que devem ser levadas em consideração para que o controle não fique instável. As simulações apresentadas a seguir têm o objetivo de verificar o que pode ocorrer com a ação de controle quando se sobre treina a rede.

A inserção de ruído na medição de velocidade faz com que se tenha sempre erro entre o modelo e a planta e com isso os pesos da rede estarão sempre sendo ajustados. Esse efeito é desejável para a investigação dos efeitos que ocorrem quando se sobre treina a rede. A senóide de referência também contribui para que se tenha o ajuste dos pesos.

A Figura 5.24 apresenta o controle da velocidade do motor efetuado por um longo período de tempo com ruído na medição de velocidade. Pode-se observar que a ação de controle apresenta patamares diferentes ao longo do tempo. Isso se deve ao sobre treinamento da rede que atingiu um estágio de bom conhecimento sobre a planta e depois começou a perdê-lo novamente. É interessante observar também que depois do nível da ação de controle

atingir patamares altos, o patamar da ação de controle ficou oscilando entre níveis altos e níveis baixos, mas não chegou a apresentar ação de controle desestabilizada.

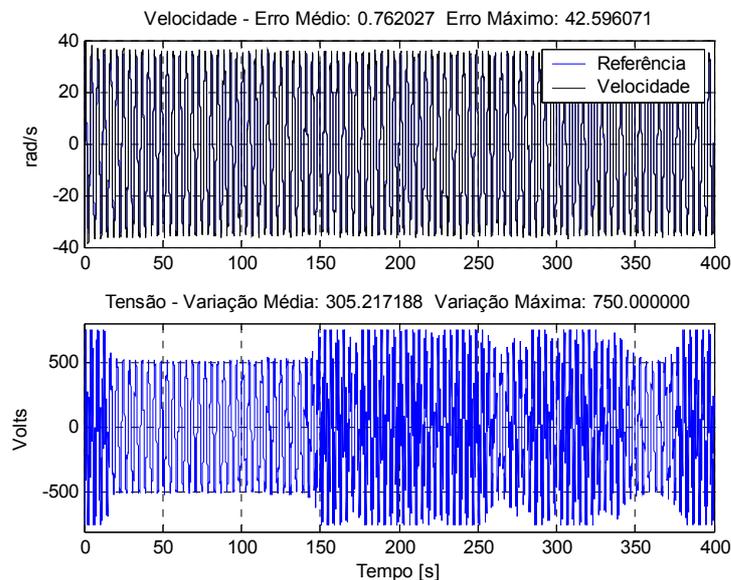


Figura 5.24 – Controle da velocidade do motor CC com ruído na medição de velocidade onde o controle é feito por um longo período de tempo.

A Figura 5.25 apresenta o *zoom* do controle da velocidade por volta do tempo $t = 200$ segundos, onde aparentemente tem-se o maior nível da ação de controle e também a maior variação. Observa-se que a velocidade segue bem a referência, não apresentando grandes oscilações em torno da referência, mesmo com ruído na medição da velocidade e com uma elevada variação da ação de controle.

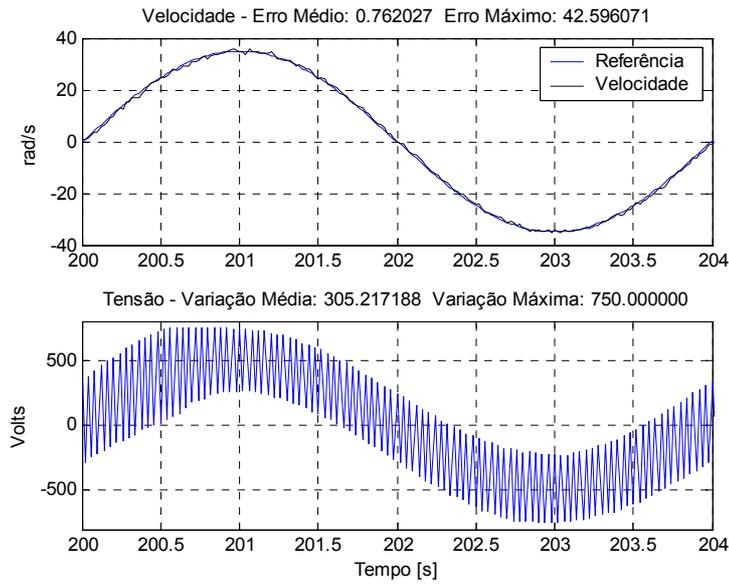


Figura 5.25 – Zoom do controle da velocidade do motor CC com ruído na medição de velocidade onde o controle é feito por um longo período de tempo.

Em situação oposta ao *zoom* apresentado na Figura 5.25 tem-se o *zoom* do controle da velocidade por volta do tempo $t = 40$ segundos, apresentado na Figura 5.26, onde aparentemente tem-se o menor nível da ação de controle e também a menor variação. Observa-se que a velocidade segue bem a referência não apresentando grandes oscilações em torno da referência mesmo com ruído na medição da velocidade. A ação de controle, praticamente, varia somente o necessário para controlar a velocidade ficando bem suave.

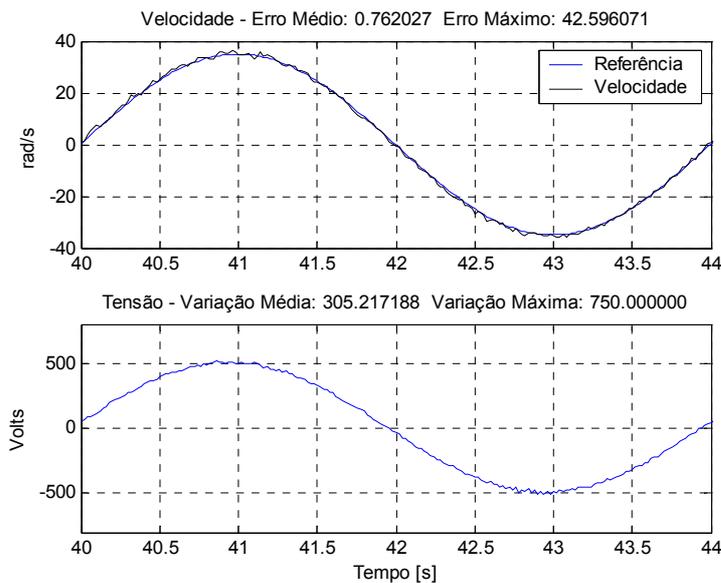


Figura 5.26 – Zoom do controle da velocidade do motor CC com ruído na medição de velocidade onde o controle é feito por um longo período de tempo.

Observou-se o efeito destrutivo quando se treina continuamente a rede. Algumas estratégias simples podem ser empregadas de forma a amenizar bem esses problemas já que eles sempre existirão. Uma delas é ter o ganho de treinamento adaptativo em função do erro entre a saída da planta e a referência. Como descrito anteriormente, a equação (4.5) que foi proposta no Capítulo 4 como uma forma de se ajustar o ganho de treinamento em função do erro ou erro médio, também pode ser utilizada aqui para se ajustá-lo.

Outra maneira de se solucionar o problema do efeito destrutivo do prévio treinamento da rede é limitar a variação da ação de controle através da formulação apresentada na equação (5.1). Quando se limita a variação da ação de controle, não ocorrem grandes variações dos estados internos da planta e isso faz com que o modelo esteja representando melhor o estado interno da planta no momento do controle. Assim, o erro entre o modelo e a planta diminui consideravelmente e em conseqüência o ajuste dos pesos é feito com menos intensidade. Foi utilizado no controle da velocidade, apresentado na Figura 5.25, a variação máxima de 750 volts, variação essa que normalmente não deve ocorrer no processo, mas é utilizada aqui para verificar os problemas que podem ocorrer no processo utilizando o controlador nessa situação.

A Figura 5.27 apresenta o controle da velocidade onde a máxima variação da ação de controle imposta é de 70 volts. Essa limitação pode ainda ser considerada alta, muito mais do que suficiente para que se efetue o controle da velocidade do motor CC com uma boa eficiência. Nota-se primeiramente que os patamares diferentes não ocorrem com essa limitação e que não existe uma tendência da ação de controle em se desestabilizar.

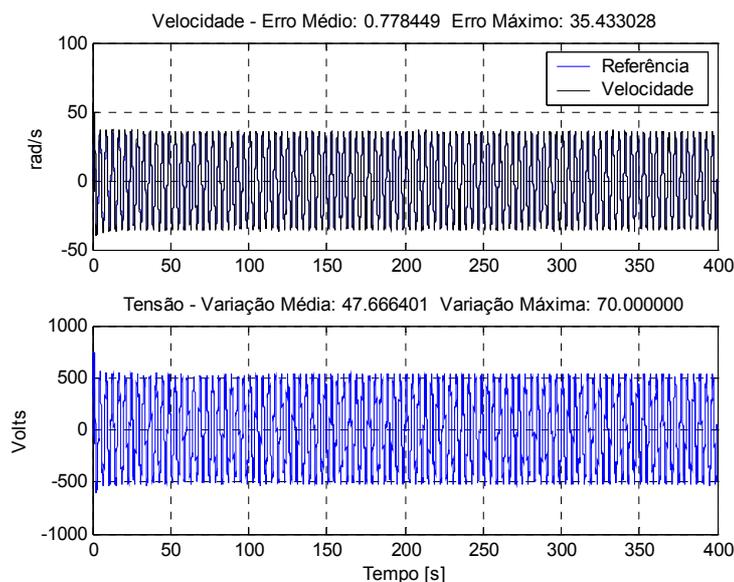


Figura 5.27 – Controle da velocidade do motor CC com ruído na medição de velocidade onde o controle é feito por um longo período de tempo com limitação máxima de 70 volts.

5.2.2.10 Controle por um longo período com inversão do torque de carga

Assim como na seção anterior, o controle aqui será realizado durante um longo período a fim de se verificar o que pode ocorrer com a ação de controle quando a rede é sobre treinada e tem-se a inversão do torque de carga.

A Figura 5.28 apresenta o controle da velocidade do motor efetuado por um longo período de tempo com inversão do torque de carga. Mesmo utilizando a máxima variação de 70 volts da ação de controle, que foi utilizada com sucesso no treinamento da rede com inserção de ruído na medição de velocidade, não se obteve sucesso no controle da velocidade do motor CC por um longo período. Observa-se que a rede não consegue generalizar bem o modelo da

planta e com isso há o sobre treinamento da rede, ocorrendo a perda de conhecimento que foi obtido através do próprio treinamento realizado ao longo do controle. Conseqüentemente, tem-se a ação de controle aplicada no processo que não é capaz de fazer com que a velocidade atinja a referência de velocidade.

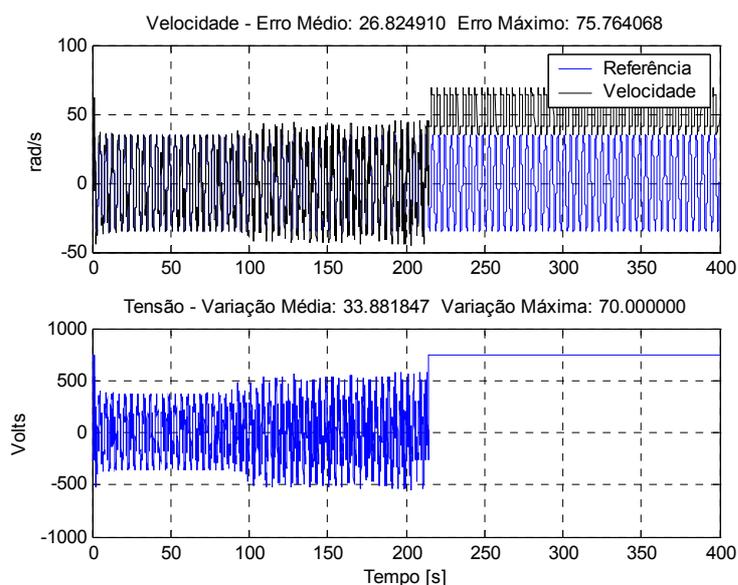


Figura 5.28 – Controle da velocidade do motor CC com inversão do torque de carga onde o controle é feito por um longo período de tempo com limitação máxima de 70 volts.

Diferentemente da situação onde se tem inserção de ruído na medição de velocidade, a inversão do torque de carga faz com que a planta passe por estados bem distintos e isso faz com que o treinamento não consiga generalizar bem o modelo da planta, principalmente porque os dados são apresentados de forma contínua. Como descrito anteriormente, é importante ressaltar que o controlador está exposto a uma situação bem adversa, pois o torque de carga está sendo invertido a cada dois segundos.

A Figura 5.29 apresenta o controle da velocidade do motor CC utilizando ajuste automático do ganho de treinamento em função do erro através da formulação apresentada na equação (4.5). Utiliza-se o parâmetro $A = 30$ e o parâmetro $B = 3$. Nota-se que o problema do treinamento contínuo da rede foi amenizado, mas em compensação apresentou erro médio grande.

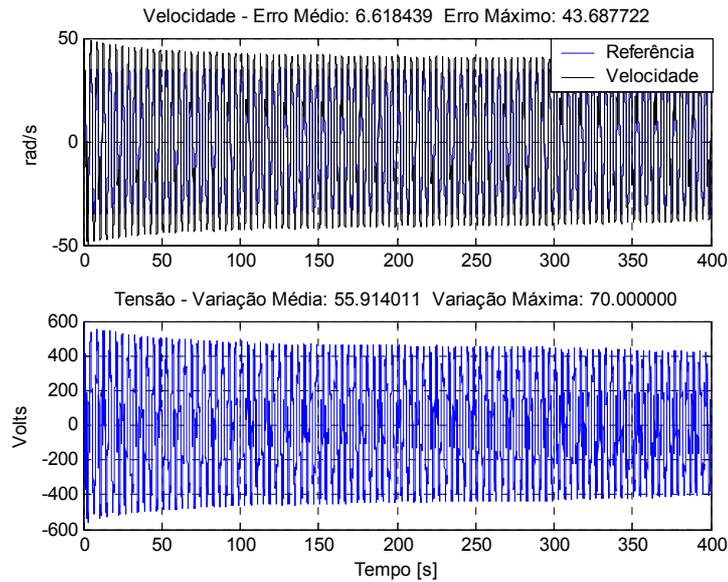


Figura 5.29 – Controle da velocidade do motor CC com inversão do torque de carga onde o controle é feito por um longo período de tempo com limitação máxima de 70 volts e ajuste automático do ganho de treinamento.

Quando se inverte o torque de carga, o erro de velocidade se torna grande, ou seja, existe um degrau entre a referência e a saída da planta. Como descrito anteriormente, o controlador tenta reduzir esse erro variando bastante a ação de controle para que no instante seguinte tenha-se erro nulo. Isso faz com que a planta passe por estados bem diferentes que prejudica o treinamento da rede sobrecarregando o ajuste dos pesos. Caso esse degrau entre a referência e a saída da planta for menor, a rede conseguiria generalizar melhor a planta, pois a transição entre os estados não seria tão rápida a ponto de a rede conseguir acompanhar sem ajustar demasiadamente os pesos. A formulação apresentada na equação (5.2), que é nomeada aqui como limitação da referência, faz com que se tenha um menor degrau entre a referência e a saída da rede.

$$\begin{aligned}
 & \text{Se } r_k \geq y_{k-1} \\
 & \quad \tilde{r}_k = \min(r_k, y_{k-1} + \Delta y_{\max}) \\
 & \text{Se } \textit{n\~{a}o} \\
 & \quad \tilde{r}_k = \max(r_k, y_{k-1} - \Delta y_{\max})
 \end{aligned} \tag{5.2}$$

A limitação da referência faz com que a planta passe por estados intermediários antes de se atingir a referência. Basicamente a formulação faz com que se tenha uma referência em

rampa até que se atinja a referência real desejada. Isso reflete no aprendizado da rede porque pode sincronizar o estado da planta com o estado da rede, diminuindo o erro e em consequência menor é o ajuste dos pesos da rede. Além disso, o degrau de referência deve ser sempre evitado mesmo que ele seja provocado por perturbações ou carga do processo, pois faz com que a ação de controle varie bastante forçando bastante os atuadores do processo.

Degraus de referência geralmente não são aplicados na prática, aplica-se geralmente rampa, com taxa de crescimento adequada ao processo, como referência para que o controlador tenha uma ação de controle mais suave e force menos os atuadores do processo. Aplicar uma rampa, além de não forçar os atuadores do processo, pode fazer com que se atinja a referência em um tempo menor do que quando se aplica um degrau e ocorre *overshoot* no sinal. Aplicando uma rampa de referência, o *overshoot*, caso haja nessa situação, será menor do que na situação de degrau e isso reduz o tempo de assentamento de forma que o tempo gasto para atingir a referência seja menor.

A Figura 5.30 apresenta o controle da velocidade do motor utilizando limitação da referência através da formulação apresentada na equação (5.2) onde se utiliza o delta de referência máximo igual a 2 rad/s. Nota-se que o problema da desestabilização da ação de controle provocada pelo sobre treinamento da rede é bastante amenizado, pois o patamar da ação de controle é praticamente o mesmo durante todo o treinamento, exceto pelo início onde não se tem conhecimento nenhum sobre a planta, e muito menos se observa uma tendência da ação de controle em se desestabilizar. O erro médio de velocidade (4,7726) é reduzido quando comparado com a técnica de ajuste automático de ganho de treinamento (6,6184), assim como a variação média da tensão que apresenta 10,111 volts com a limitação da referência e 55,914 volts com a técnica de ajuste automático do ganho de treinamento.

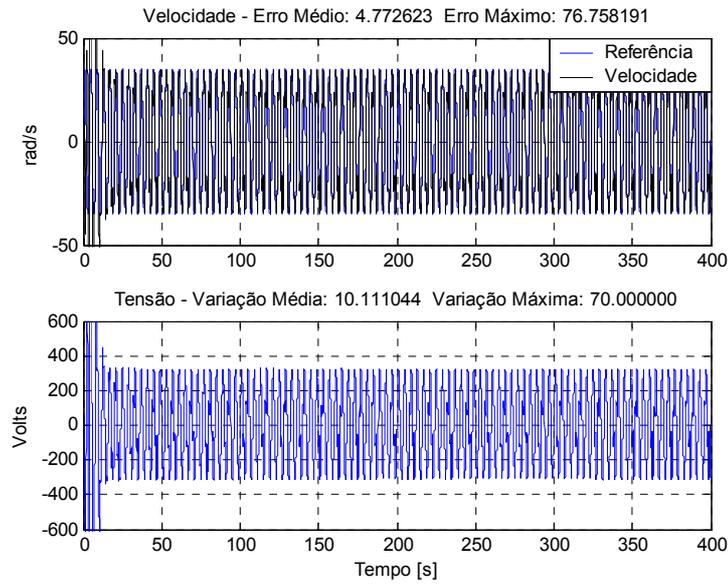


Figura 5.30 – Controle da velocidade do motor com inversão do torque de carga onde o controle é feito por um longo período de tempo e com limitação da referência.

A maneira mais simples de se amenizar o problema do sobre treinamento é a de simplesmente diminuir a máxima variação da ação de controle que pode ser aplicada no processo. Na Figura 5.31 tem-se o controle utilizando a máxima variação da ação de controle igual a 17,5 volts, um bom valor para poder ser utilizado no processo. Observa-se que a rede consegue modelar a planta e com isso tem-se a ação de controle aplicada no processo que é capaz de fazer com que a velocidade atinja a referência de velocidade com um erro menor do que na situação com limitação da referência, onde se obteve bons resultados, mas com uma variação média da tensão maior.

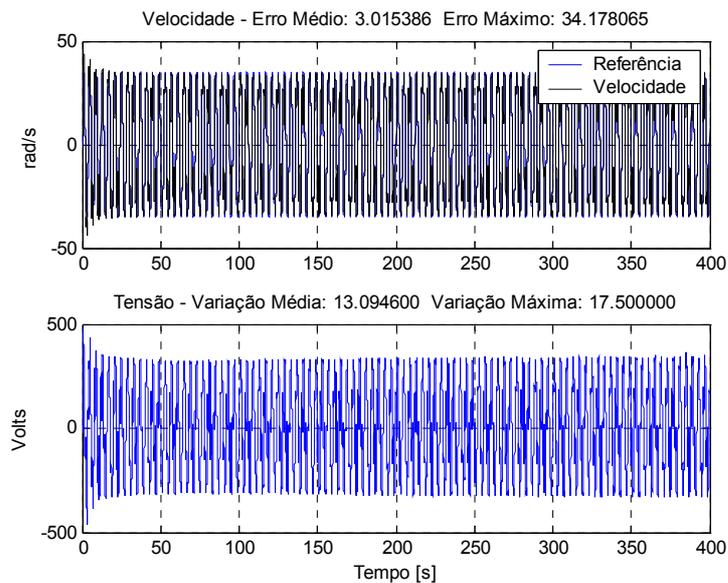


Figura 5.31 – Controle da velocidade do motor CC com inversão do torque de carga onde o controle é feito por um longo período de tempo com limitação máxima de 17,5 volts.

5.3 Redes lineares e não lineares

Nos resultados de simulação apresentados na seção 5.2 foi utilizada a rede NFN com duas partições, o que faz com que a mesma tenha um comportamento de uma rede linear quando as entradas estão dentro do universo de discurso. Foram feitas diversas simulações utilizando redes não lineares como o NFN com mais de duas partições e a MLP. Os resultados obtidos com essas redes foram inferiores aos resultados da rede NFN com 2 partições. A principal razão para esse insucesso se deve ao fato que a inversa é obtida através de uma equação de reta e utilizando essas redes não se consegue uma saída linear em relação à entrada que representa a ação de controle da planta para todo universo de discurso.

Contudo, adaptações podem ser feitas nessas redes não lineares para que se possa utilizá-las para fazer aproximação da inversa. Basta que a mesma seja linear em relação à entrada que representa a ação de controle.

No NFN, caso se utilize mais de duas partições para as outras entradas que não representa a ação de controle e duas partições para a entrada que representa a ação de controle, se

consegue fazer com que se tenha a saída linear em função da entrada que representa a ação de controle.

No caso da rede MLP o que se pode fazer é utilizar u_{1k} e u_{2k} perto um do outro de forma que para esses dois pontos escolhidos obtenha-se uma aproximação mais local possível. Desta forma, a aproximação local, reta construída a partir desses pontos, tende a coincidir com o sistema real a ser controlado. Assim, a ação de controle tenderá a fazer com que o sistema controlado atinja a referência na próxima amostragem. Mas deve-se ter o cuidado para que u_{1k} e u_{2k} não gerem $y_{1(k+1)}$ e $y_{2(k+1)}$ iguais ou de valores muito próximos, já que, como descrito anteriormente, a inversa é obtida através da equação de uma reta e para esses valores iguais qualquer valor de u_k satisfaz a equação.

Além das redes não lineares conseguirem generalizar melhor a planta, outra vantagem quando se utiliza essas redes apropriadas para utilizar a aproximação da inversa é que diminui a perda do prévio conhecimento devido ao sobre treinamento da rede que pode provocar a desestabilização da ação de controle. Isso é possível porque a rede generaliza melhor a planta e com isso menor é o erro entre o modelo e planta, diminuindo o ajuste dos pesos. Essa característica se destaca ainda mais quando se utiliza a rede NFN não linear apropriada para o controle com a aproximação da inversa. Além de a rede NFN generalizar bem a planta, o ajuste global dos pesos ocorre com menos intensidade porque somente duas partições estão ativas por entrada a cada iteração. Isso porque, os ajustes nos pesos ficam divididos entre os mesmos, o que não ocorre com a versão linear já que sempre são ajustados os mesmos pesos a cada amostragem de controle.

Na Figura 5.32 pode-se visualizar o controle da velocidade do motor efetuado por um longo período de tempo com inversão do torque de carga utilizando a rede NFN com 4 partições para as outras entradas que não representam a ação de controle e 2 partições para a entrada que representa. Mesmo utilizando a máxima variação de 70 volts da ação de controle, que foi utilizada sem sucesso com a rede NFN linear apresentada na Figura 5.28, a ação de controle não se desestabilizou diferentemente da situação apresentada na Figura 5.32 e apresentou erro médio e máximo próximos do controle com a variação da ação de controle limitada em 17,5 volts, veja Figura 5.31.

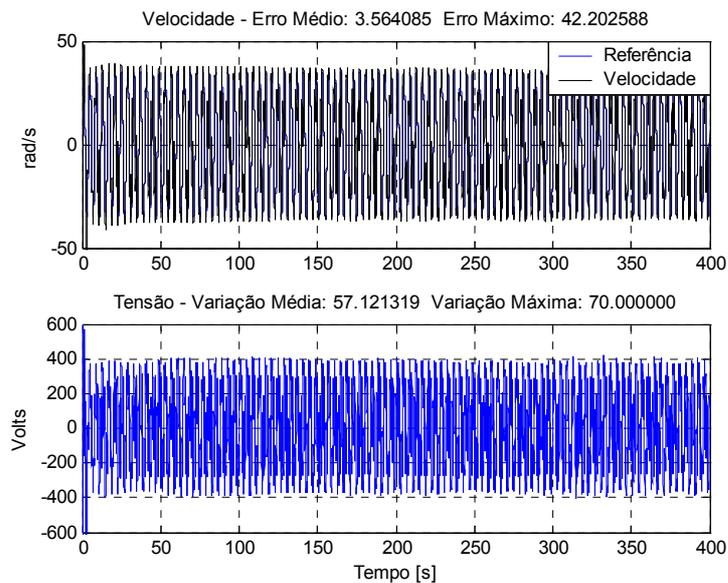


Figura 5.32 – Controle da velocidade do motor CC com inversão do torque de carga onde o controle é feito por um longo período de tempo com limitação máxima de 70 volts.

Uma desvantagem apresentada do NFN não linear apropriado para se utilizar o método da aproximação da inversa em relação ao NFN com duas partições, linear, é que o aprendizado da rede é mais lento. Na rede linear, quando se apresenta um dado para treinamento se está treinando toda rede, o que não acontece com a rede não linear, onde só se treina as partições ativas. Isso não chega a ser um grande problema porque em uma situação prática, a princípio, não há motivos para não se realizar um prévio treinamento off-line da rede. Aliás, o treinamento off-line da rede modela muito melhor a planta do que o treinamento on-line, principalmente quando se utiliza dados amostrados que têm uma boa informação sobre o processo. Por modelar melhor a planta, pode-se utilizar um ganho de treinamento menor e uma máxima variação da ação de controle menor. Isso diminui muito os problemas que podem ocorrer em um controle com treinamento on-line que é necessário para fechar a malha.

Uma estratégia interessante de controle com a aproximação da inversa é utilizar uma rede não linear apropriada, fazendo um treinamento prévio off-line, fechando a malha com treinamento on-line, colocando um ganho de treinamento baixo, limitando a referência e a ação de controle para que se efetue um bom controle e para que o mesmo seja robusto.

5.4 Controle de sistemas não lineares

Embora o motor de corrente contínua seja um sistema não linear, ele pode ser modelado como um sistema linear. Devido a isso, uma rede linear modela bem esse sistema e o controlador dinâmico inverso aproximado, normalmente, apresentará resultados satisfatórios no controle desse tipo de planta, já que a dinâmica inversa aproximada se baseia na obtenção de uma reta em relação à ação de controle.

A fim de se diversificar o uso desse controlador, será feita simulações controlando dois sistemas não lineares. Esses sistemas foram utilizados em diversos trabalhos como, por exemplo, *Narendra & Parthasarathy (1990)*, *Wang (1994)* e *Caminhas et al. (1998)*.

5.4.1 1º Sistema não linear

Seja o sistema não linear definido pela equação de diferenças (5.3).

$$y_{k+1} = 0.3 \cdot y_k + 0.6 \cdot y_{k-1} + f(u_k) \quad (5.3)$$

O sistema apresenta uma dependência linear entre valores passados da entrada e uma dependência não linear com relação à entrada da planta, $u(k)$, mostrada na equação (5.4).

$$f(u_k) = 0.6 \cdot \sin(\pi \cdot u_k) + 0.3 \cdot \sin(3\pi \cdot u_k) + 0.1 \cdot \sin(5\pi \cdot u_k) \quad (5.4)$$

A dependência não linear em relação à entrada do sistema é, a princípio, um problema para o controlador, já que a ação de controle é obtida através de uma equação de reta em relação a essa entrada.

No treinamento off-line da rede NFN utilizou-se duas partições em sua estrutura e os seguintes padrões de entrada: $y(k)$, $y(k-1)$ e $u(k)$. Os dados utilizados no treinamento off-line foram provenientes de uma distribuição uniforme no intervalo de $[-1,1]$. O treinamento da rede foi realizado com o método dos mínimos quadráticos.

A Figura 5.33 apresenta a validação do treinamento com a rede NFN onde se utilizou os dados provenientes da equação (5.5). Nota-se que a rede conseguiu aproximar bem a planta não linear.

$$u_k = \sin\left(\frac{2\pi \cdot k}{250}\right) \quad k = 1..600 \quad (5.5)$$

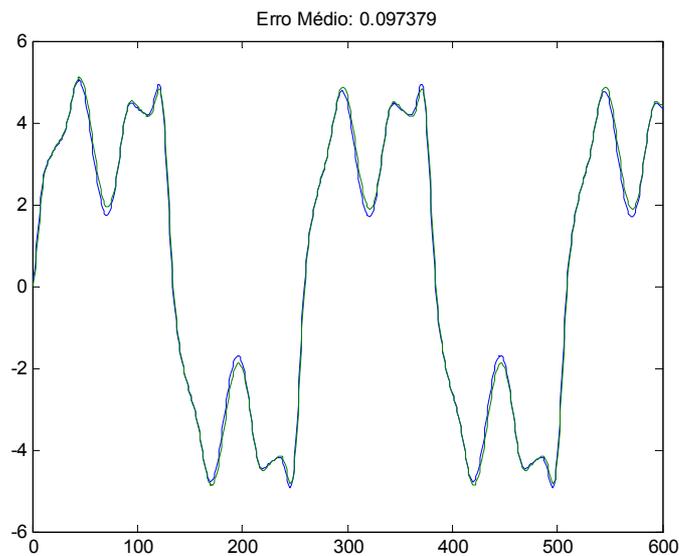


Figura 5.33 – Validação do treinamento do 1º sistema não linear utilizando a rede NFN.

A Figura 5.34 apresenta o controle do sistema não linear utilizando aproximação da inversa com treinamento off-line da rede NFN. Nota-se que a saída do sistema é bem controlada por este controlador. A razão do sucesso é devido à utilização de um delta máximo de variação da ação de controle pequeno (0,002). Isso faz com que os estados do sistema variem pouco e assim, o sistema assume características locais que são de mais fácil aproximação para a rede.

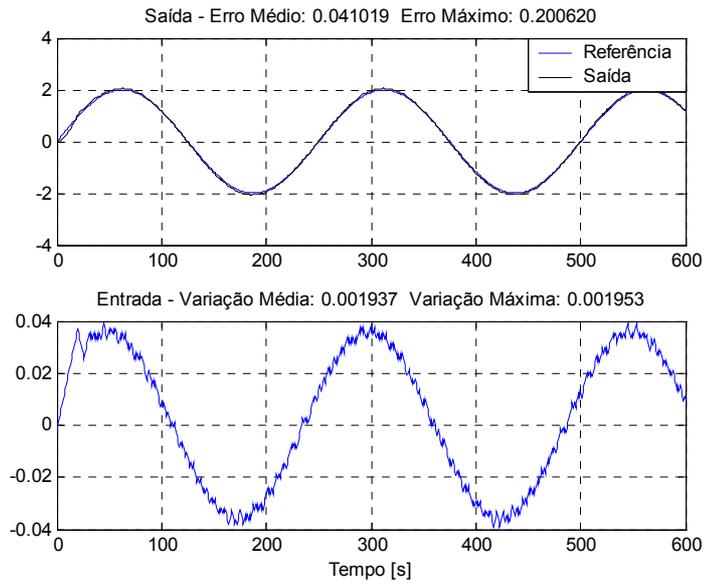


Figura 5.34 – Controle do sistema não linear utilizando aproximação da inversa com treinamento off-line da rede NFN.

A Figura 5.35 apresenta o controle do sistema não linear utilizando aproximação da inversa com treinamento on-line da rede NFN. O controle com treinamentos on-line apresenta resultados melhores do que no sistema controlado com treinamento off-line.

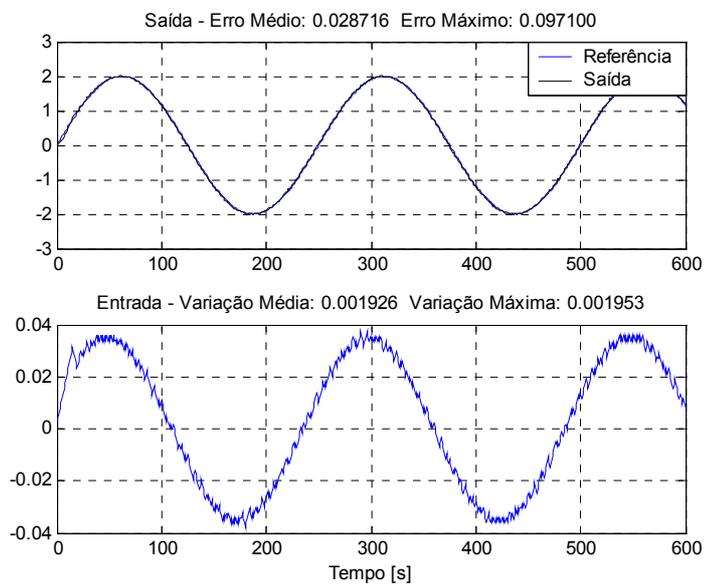


Figura 5.35 – Controle do sistema não linear utilizando aproximação da inversa com treinamento on-line da rede NFN.

5.4.2 2º Sistema não linear

Seja o sistema não linear definido pela equação de diferenças (5.6).

$$y_{k+1} = \frac{y_k \cdot y_{k-1} \cdot (y_k + 2.5)}{1 + y_k^2 + y_{k-1}^2} + u_k \quad (5.6)$$

Este sistema não linear possui dois pontos de equilíbrio (0, 0) e (2, 2), no espaço de estado. Segundo *Narendra & Parthasarathy (1990)*, para qualquer entrada $|u[k]| \leq 5$, a saída da planta é uniformemente limitada para condições iniciais (0, 0) e (2, 2) e satisfaz a inequação $|y[k]| \leq 13$.

Na estrutura da rede NFN utilizou-se 2 partições para a entrada que representa a ação de controle e 8 partições para as outras 5 entradas. Foram utilizadas entradas de ordem elevada porque com entradas simples o NFN não conseguiu modelar o sistema não linear. Segue os padrões de entrada utilizados: $y(k)$, $y(k-1)$, $y(k)*y(k)$, $y(k)*y(k-1)$, $y(k-1)*y(k-1)$ e $u(k)$. Os dados utilizados no treinamento off-line foram provenientes de uma distribuição uniforme no intervalo de $[-1,1]$. O treinamento da rede foi realizado com o método dos mínimos quadráticos.

A Figura 5.35 apresenta a validação do treinamento off-line com a rede NFN onde se utilizou os dados provenientes da equação (5.7). Nota-se que a rede conseguiu uma aproximação regular do sistema não linear.

$$u_k = 0.5 \cdot \sin\left(\frac{2\pi \cdot k}{25}\right) \quad k = 1..100 \quad (5.7)$$

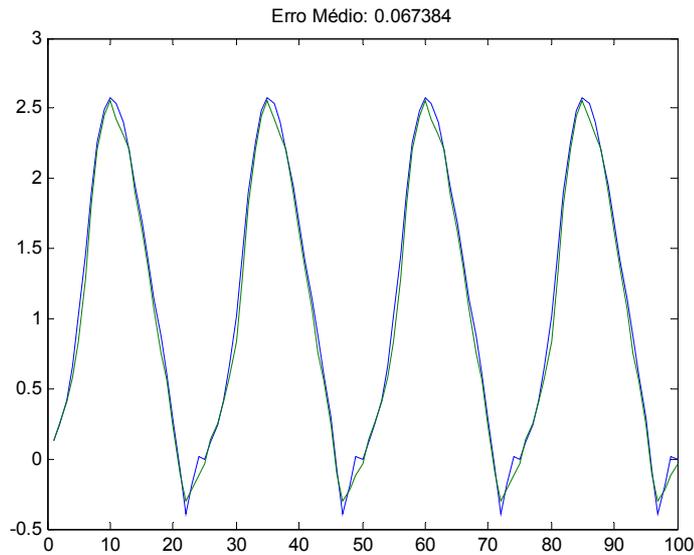


Figura 5.36 – Validação do treinamento off-line com a rede NFN.

A Figura 5.37 apresenta o controle do sistema não linear utilizando aproximação da inversa com a rede NFN treinada off-line. O controlador não consegue fazer com que a saída do sistema siga a referência muito bem. Utiliza-se um delta máximo da variação da ação de controle igual a 0,25.

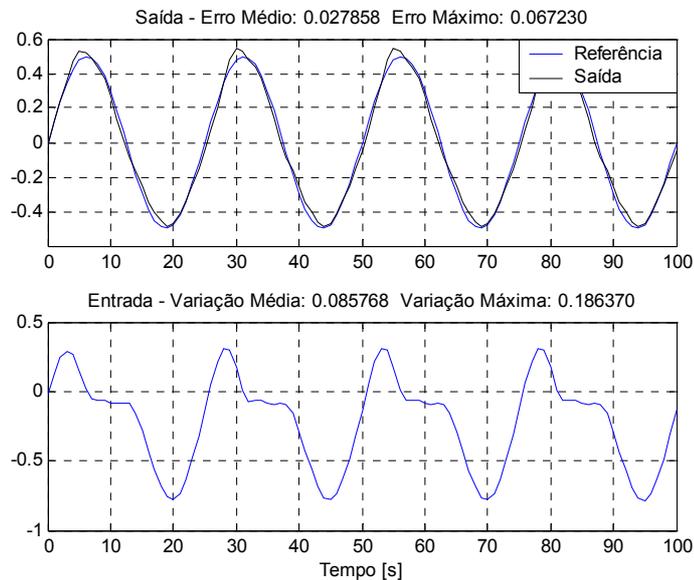


Figura 5.37 – Controle do sistema não linear utilizando aproximação da inversa com a rede NFN treinada off-line.

Os resultados apresentados pelo controlador utilizando a rede NFN para modelar o sistema não foram satisfatórios. Em virtude disso, foi feito o mesmo controle utilizando a rede MLP treinada off-line. A Figura 5.38 apresenta o controle empregando essa rede. Pode-se observar comparando a Figura 5.38 com a Figura 5.37 que os resultados utilizando a rede MLP foram bem melhores.

O sucesso da utilização da rede MLP nesse controlador foi possível graças aos pontos utilizados para obter a reta de escolha da ação de controle. Esses pontos eram próximos um dos outros que permitiram obter uma reta local que coincida com o sistema a ser controlado.

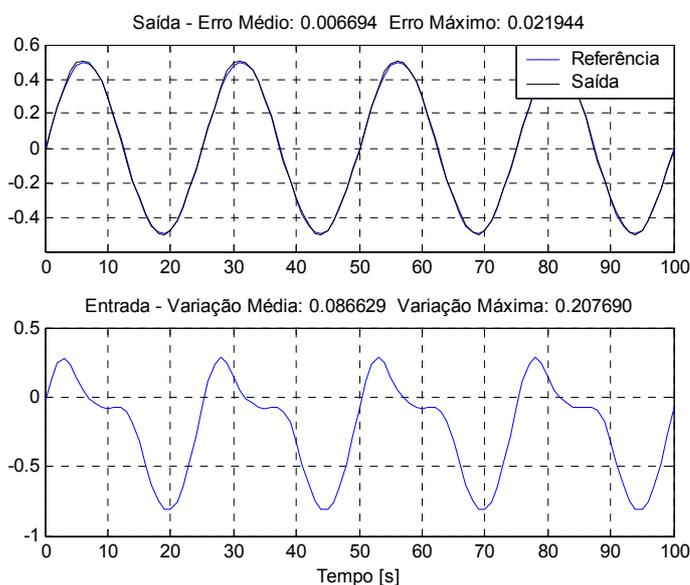


Figura 5.38 – Controle do sistema não linear utilizando aproximação da inversa com a rede MLP treinada off-line.

No treinamento off-line da rede MLP foram utilizados 10 neurônios na camada escondida e os seguintes padrões de entrada: $y(k)$, $y(k-1)$ e $u(k)$.

Os resultados obtidos no controle do sistema não linear com treinamentos on-line utilizando a rede NFN e a rede MLP não foram satisfatórios. Um dos principais motivos, além de ser um sistema não linear e utilizar sempre um dado no treinamento a cada amostragem, é a questão de se utilizar uma senóide como referência que faz com que sempre se tenha variações dos estados do sistema e com isso a rede não consegue modelar esses estados para tal sistema.

Embora fora obtido resultados melhores com a rede MLP deve-se ressaltar que a sua utilização deve ser compatível com o sistema a ser controlador. Isso porque a ação de controle é encontrada através de uma equação de reta em relação à entrada que representa a ação de controle. Por ser uma rede não linear, essa reta pode variar muito conforme os pontos escolhidos para se encontrá-la. Assim, é sempre recomendado que seja feito um treinamento off-line da rede e que tente sempre fazer uma aproximação mais local possível para que a representação do modelo de reta obtido através dessa rede para a escolha da ação de controle coincida com o sistema a ser controlado.

5.5 Conclusões

Neste capítulo foi apresentado simulações de controle de velocidade de um motor CC e de dois sistemas não lineares utilizando a aproximação da dinâmica inversa. Mostrou-se que a obtenção da função que aproxima a dinâmica inversa da planta é obtida de forma mais fácil do que função que mapeia a dinâmica inversa direta, principalmente quando se realiza treinamentos on-line no qual é de fundamental importância para se efetuar o controle em malha fechada.

Como era de se esperar, o controle utilizando a aproximação da dinâmica inversa apresentou características semelhantes ao controle utilizando a dinâmica inversa como, por exemplo, rejeitar bem as perturbações de carga, seguir trajetórias não estacionárias com erro pequeno ou nulo, variar bastante a ação de controle em situações que gerem erro (ruído na medição, torque de carga).

O controle feito somente com treinamento off-line apresentou resultados satisfatórios, principalmente quando se limitou à ação de controle, que além de ser necessário para a estabilidade física do processo, melhorou os índices de desempenho do processo. Diferentemente do controle utilizando a dinâmica inversa, o controle com treinamentos on-line com a aproximação da inversa apresentou resultados melhores e mostrou que tem maior capacidade de retenção de conhecimento sobre o processo no qual está controlando.

O controlador foi exposto a algumas situações adversas de forma a revelar possíveis problemas do controle. Alguns problemas foram observados onde o controlador não conseguiu se sair bem, mas algumas técnicas simples tais como: limitação da ação de controle, ajuste automático do ganho de treinamento, limitação da referência e utilizar rede não linear apropriada para o método foram propostas a fim de se contornar e evitar os mesmos.

Capítulo 6

Conclusões e Propostas de Continuidade

Neste capítulo são apresentadas as principais conclusões deste trabalho, assim como as propostas para trabalhos futuros.

6.1 Conclusões

A sintonia dos parâmetros de controladores convencionais não é uma das tarefas mais fáceis, mesmo sendo controladores simples e tendo sua base teórica toda consolidada. Normalmente, as técnicas de sintonia de parâmetros de controladores não são utilizadas na prática e quando utilizadas dificilmente apresentam resultados satisfatórios quando comparado com os resultados de projeto. Isso leva o projetista a aplicar métodos heurísticos para sintonizar os controladores. Assim, o esforço humano para sintonizá-los, normalmente, é grande e nem sempre satisfatório. Em virtude disso, diversos são os estudos para que o projeto de controladores seja realizado com menor esforço e resultados satisfatórios.

Nesse sentido, os controladores baseados em inteligência computacional tem se destacado por não ser necessário ter um conhecimento profundo do sistema a ser controlador. Isso porque as redes artificiais possuem a capacidade de aprendizado e possibilitam que o sistema controlável seja adaptável. Isso abstrai o projetista em ter que saber as técnicas de sintonia de parâmetros dos controladores convencionais que é um das principais razões pela má sintonia desses controladores.

O ONFC é um desses controladores baseados em inteligência computacional e uma alternativa interessante aos controladores do tipo PI, pois é simples, de baixo custo computacional, se adapta à dinâmica do processo numa ampla faixa de operação e de fácil sintonia de seus parâmetros. Por ser um controlador recentemente proposto, ainda não existem grandes estudos realizados sobre o mesmo. Foi feita uma contribuição nesse sentido onde é apresentado um estudo através do desenvolvimento matemático das equações do controlador. Através desse estudo, pôde-se sintetizar que este controlador possui as características de um controlador PI adaptativo. Essa contribuição possibilita ainda que sejam realizados os primeiros passos para se criar métodos de sintonia dos parâmetros do controlador. Por possuir as características de um PI adaptativo, as estratégias de sintonia dos parâmetros do controlador podem ser criadas utilizando os diversos estudos existentes sobre sintonia do PI.

Foi apresentado o análogo neural do ONFC, o On-Line Neuro Controller (ONC). Esse controlador é mais simples do que a versão neurofuzzy e com menor custo computacional. Embora isto não seja uma grande vantagem, já que o ONFC apresenta um custo computacional baixo. Nessa versão os pesos da rede têm uma ligação mais direta com os ganhos de um controlador PI.

Foi apresentada também a versão que utiliza Controle por Modos Deslizantes para realizar o treinamento da rede do controlador. Essa versão possibilita obter melhores resultados devido à superfície de deslizamento. Embora os limites do ganho de treinamento tenham sido estabelecidos na rede NFN, eles não são aplicáveis no treinamento da rede NFN do ONFC. Isso porque quando se utiliza a rede para atuar como um controlador utiliza-se o erro entre a saída da planta e a referência na superfície de deslizamento e não o erro entre a saída da rede e o padrão de saída para treinamento. Assim, a princípio, não é possível usar os limites da taxa de aprendizado para se determinar a taxa de aprendizado ótima que poderia ser utilizado em cada iteração. Para ser possível, deve-se incorporar o efeito do sistema a ser controlado nas equações de ajustes dos pesos. Mas o primeiro passo já está feito que é realizar o treinamento da rede NFN com o controle em modos deslizantes.

O controlador neurofuzzy baseado na dinâmica inversa do processo é um controlador interessante no sentido de se obter um controlador de fácil sintonia utilizando a inteligência

computacional. Basta, a princípio, utilizar dados históricos do processo para que se obtenha um modelo artificial da função inversa e utilizá-la para controlar um processo. Mas, nem todo processo possui uma função inversa e quando existe não é fácil obtê-la. Neste trabalho, o controle dinâmico inverso foi utilizado para controlar a velocidade de um motor de corrente contínua com treinamentos off-line e on-line da rede artificial que modela a função inversa.

Este controlador mostrou que é interessante a sua utilização quando se tem trajetórias não estacionárias, como a rampa e a senóide, porque o erro tenderá a ser menor, devido à capacidade de previsão do modelo, do que um PI que sempre apresentará erro para seguir trajetórias desse tipo. Em contrapartida, a ação de controle varia bem mais do que em um PI para que no instante seguinte tenha-se erro nulo, mas que facilmente pode ser limitada e continuar apresentando resultados satisfatórios. Mostrou-se que a função inversa não foi de fácil obtenção, principalmente quando há uma super amostragem do sinal e ou quando se utiliza somente treinamentos on-line, que é necessário para se fechar a malha de controle.

Devido às dificuldades de se obter o modelo artificial inverso do processo e por este controlador não exigir que o projetista tenha um grande conhecimento do processo, foi proposto o controlador dinâmico inverso aproximado. Este controlador utiliza o modelo artificial do processo para obter a função inversa através de uma equação de reta. Como a obtenção do modelo artificial do processo é mais fácil de se obter do que o modelo artificial inverso, esse controlador apresentou melhores resultados do que o controle dinâmico inverso, principalmente quando comparado com o controle com treinamentos on-line.

Perante as análises e os resultados apresentados neste trabalho, pode-se concluir que as estruturas de controladores utilizadas são interessantes por minimizar o esforço humano na sintonia de controladores, por obter resultados satisfatórios e por serem estruturas que não necessitam de grande conhecimento do processo para a sintonia dos parâmetros dos controladores.

6.2 Propostas de continuidade

Considerando os estudos realizados neste trabalho, as propostas de continuidade podem ser listadas a seguir:

- Desenvolver as equações de convergências do ONFC em modos deslizantes englobando o efeito da planta para que se possa fazer o ajuste automático da taxa de aprendizado.
- Propor estratégias de sintonia dos parâmetros do ONFC e do ONC aproveitando a análise matemática desenvolvida.
- Realizar o controle de plantas reais utilizando o ONFC em modos deslizantes, o ONC e o controle dinâmico inverso aproximado.
- Propor melhores meios de se realizar o treinamento on-line de uma rede artificial no controlador dinâmico inverso aproximado.
- Fazer análise de convergência do controlador dinâmico inverso aproximado.

Referências Bibliográficas

- Caminhas et al. (1998)* Caminhas, W.M.; Pereira, G. A. S.; Tavares, H.M.F. ; Gomide, F.A.C.; *Identificação de Sistemas Dinâmicos: Abordagem Baseada em Neurônio Nebuloso*. In Anais do V Simpósio Brasileiro de Redes Neurais (SBRN 98), Belo Horizonte – MG, pp. 105 – 110, Dezembro 1998.
- Carvalho et al. (2006)* Carvalho, Mateus de A; Gouvêa, Marlon R.; Silva, Renato G. S.; Caminhas, Waldir M.; *Uso de Controlador Neurofuzzy baseado em redes NFN para controle de pH*. XVI Congresso Brasileiro de Automática – 3 a 6 de outubro de 2006, Salvador.
- Chen et al. (2003)* Chen, Z.; Yue, Y. ; Zhao, G.; *Neural networks based an inverse dynamic model adaptive control Systems*, Man and Cybernetics, 2003. IEEE International Conference on Volume 2, 5-8 Oct. 2003 Page(s): 1892 – 1897 vol.2
- Gouvêa (2005)* Gouvêa, Marlon R.; *Controle Neurofuzzy de motor de indução com estimação de parâmetros e fluxo de estator*, Tese de Doutorado, Programa de Pós-graduação em Engenharia Elétrica, UFMG, Belo Horizonte, 2005.
- Gouvêa et al. (2005 a)* Gouvêa, Marlon R.; Menezes, Benjamin R.; Figueiredo, E. S.; Pires, Anderson V.; Caminhas, Waldir M. *A New Neurofuzzy Controller Applied to Vector Control of Induction Motors*. The IASTED International Conference on

Computational Intelligence. Calgary, Alberta – Canada – July 4-6, 2005.

Gouvêa et al. (2005 b)

Gouvêa, Marlon R.; Menezes, Benjamin R.; Figueiredo, E. S.; Parma, G. G.; Pires, Anderson V.; Caminhas, Waldir M. *A New Neurofuzzy Controller Based on NFN Networks*. VII Congresso Brasileiro de Redes Neurais – VII CBRN, 16 a 19 de outubro de 2005, Rio Grande do Norte.

Harris et al. (1981)

Harris, C. J.; Billings, S. A.; *Self-Tuning and Adaptive Control: Theory and Applications*. London: Peter Peregrinus ltd, 1981.

Hornik et al. (1989)

Hornik, K.; Stinchcombe, M.; White, H.; *Multilayer Feedforward Networks Are Universal Approximations*. *Neural Networks*, vol. 2, pp. 359-366, 1989.

Itkis (1976)

Itkis, U.; *Control Systems of variable structure*. John Wiley and Sons, New York, 1976.

Jang et al. (1997)

Jang, J. S. R.; Sun, C. T.; Mizutani, E.; *Neurofuzzy and Soft Computing*. Prentice Hall, 1997.

Justino (2004)

Justino, Júlio C. G.; *Redes Neurais Artificiais com Treinamento On-line Aplicadas ao Controle do Motor de Indução*. Dissertação de Mestrado, Programa de Pós-graduação em Engenharia Elétrica, UFMG, Belo Horizonte, 2004.

Lee (1990)

Lee, C.C.; *Fuzzy Logic In Control Systems: Fuzzy Logic Controller - Part I & II*. IEEE Transactions on Systems, Man and Cybernetics, 20(2), pp.404-435. 1990.

Lee & Ko (2002)

Lee, K.Y.; Hee-Sang Ko; *Power system stabilization using a free-model based inverse dynamic neuro controller*. Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International

- Liguni et al. (1991)* Liguni, Y.; Sakai, H. A.; *Nonlinear Regulator Design in the Presence of System Uncertainties Using Multilayered Neural Networks*, *IEEE Trans. on Neural Networks*, Vol. 3, No.4, pp. 410-417, July 1991.
- Mamdani (1974)* Mamdani, E. H.; *Applications of fuzzy algorithms for simple dynamic plant*. Proc. IEE, vol. 121. no.12. pp. 1585-1588. 1974.
- Mamdani (1976)* Mamdani, E. H.; *Advances in the linguistic synthesis of fuzzy controllers*, Int. J. Man Mach. Studies. vol. 8, no. 6, pp. 669-678, 1976.
- Narendra & Parthasarathy (1990)* Narendra, K. S.; Parthasarathy, K.; *Identification and Control of Dynamical Systems Using Neural Networks*. IEEE Trans. On Neural Networks., Vol 1, No.1, pp. 4-27, 1990.
- Nguyen et al. (1990)* Nguyen, D.; Widrow, B.; *The Truck Backer-Upper An Example of Self-Learning in Neural Networks*, IEEE Control System Magazine, pp. 18-23, 1990.
- Nunes et al. (1996)* Nunes, W.; Gomide, F. A. C.; *Previsão de Temperaturas para Alimentos em Congelamento*. Relatório Técnico do Projeto Rhae - CNPq/Multibras - Processo 16081-96, DCA/FEEC/Unicamp, 1996..
- Ogata (2003)* Ogata, K.; *Engenharia de Controle Moderno*. Rio de Janeiro: Prentice-Hall do Brasil Ltda, 2003.
- Parma (2000)* Parma, Gustavo G.; *Treinamento de Redes Neurais Artificiais Baseado em Sistemas de Estrutura Variável com Aplicações em*

- Acionamentos Elétricos*. Tese de Doutorado, Programa de Pós-graduação em Engenharia Elétrica, UFMG, Belo Horizonte, 2000.
- Rumelhart et al. (1986) Rumelhart , D. E.; Hilton , G. E.; Williams, R.J.; *Learning representations by back-propagation errors*. Nature, 323:pp. 533-536, 1986.
- Soloway & Haley (1996) Soloway, D.; Haley, P.J.; *Neural Generalized Predictive Control*, Proceedings of the 1996 IEEE International Symposium on Intelligent Control, 1996, pp. 277-281.
- Spath (1985) Spath, H.; *Cluster Dissection and Analysis: Theory, FORTRAN Programs, Examples*, translated by J. Goldschmidt, Halsted Press, New York, 1985, 226 pp.
- Sugeno (1985) Sugeno, M.; *An introductory survey of fuzzy control*, Inform. Sci., vol. 36, pp. 59-83, 1985.
- Takagi & Sugeno (1983) Takagi, T.; Sugeno, M.; *Derivation of fuzzy control rules from human operator's control actions*, in Proc. of the IFAC Symp. on Fuzzy Information, Knowledge Representation and Decision Analysis, Marseilles, France, July 1983, pp. 55-60.
- Utkin (1977) Utkin, V.; *Variable Structure Systems with Sliding Modes*. IEEE Trans. Autom. Cont. Vol. AC-22, n.2, pp 212-222, abr. 1977.
- Utkin (1978) Utkin, V.; *Sliding Modes and Their Application in Variable Structures Systems*. Moscow, MIR, 1978.
- Wang (1994) Wang, L.; *Adaptive Fuzzy Systems and Control*, Prentice Hall, 1994.

- Widrow (1962)* Widrow, B.; *Generalization and information storage in networks of adaline "neuros"*. In M. C. Yovits, G. T. Jacobi, & G. D. Goldstein (EDs.), *Self-Organization Systems 1962* (pp, 435-461). Washigton: (Chicago 1962) Spartan.
- Yamakawa et al. (1992)* Yamakwa, T.; Uchino, E.; Miki, T.; Kusanagi; *A Neo Fuzzy Neuron and its Applications to System Identification and Predictions to System Behavior*. Proc. Of the 2nd IIZUKA, Iizuka-Japan , pp. 477-483, 1992.
- Zadeh (1965)* Zadeh, L. A.; *Fuzzy Sets*. Information and Control - 1965.

Apêndice A

Parâmetros do motor CC

<i>Tensão nominal de armadura (v_{an})</i>	<i>750 V</i>
<i>Tensão nominal de campo (v_{fd})</i>	<i>750 V</i>
<i>Corrente de armadura nominal (I_{an})</i>	<i>17098 A</i>
<i>Corrente de campo nominal (I_{fdn})</i>	<i>61,54 A</i>
<i>Resistência de armadura (r_a)</i>	<i>0,00467 Ω</i>
<i>Resistência de campo (r_{fd})</i>	<i>12,1875 Ω</i>
<i>Indutância da armadura (L_a)</i>	<i>0,000238 H</i>
<i>Indutância do campo (L_{fd})</i>	<i>8,750 H</i>
<i>Indutância comum (L_{afd})</i>	<i>0,23 H</i>
<i>Momento de Inércia (J_m)</i>	<i>2580 kgm^2</i>
<i>Coefficiente de Atrito (B)</i>	<i>80,7</i>
<i>Tempo de Amostragem</i>	<i>0,002 s</i>