

UNIVERSIDADE FEDERAL DO PARANÁ

SETOR DE TECNOLOGIA

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

**CONTROLADOR PID SINTONIZADO POR
REDES NEURAIS ARTIFICIAIS**

CURITIBA

2011

ALLAN CHRISTIAN KRAINSKI FERRARI

**CONTROLADOR PID SINTONIZADO POR
REDES NEURAIS ARTIFICIAIS**

**Trabalho de Conclusão de Curso apresentado
para obtenção do título de Engenheiro
Eletricista no curso de graduação em
Engenharia Elétrica pela UFPR – Universidade
Federal do Paraná.**

Orientador: Prof. Dr. Gideon Villar Leandro

CURITIBA

2010

ALLAN CHRISTIAN KRAINSKI FERRARI

**CONTROLADOR PID SINTONIZADO POR
REDES NEURAIS ARTIFICIAIS**

MONOGRAFIA APRESENTADA AO CURSO DE ENGENHARIA ELÉTRICA, DA
UNIVERSIDADE FEDERAL DO PARANÁ, COMO REQUISITO À OBTENÇÃO DO
TÍTULO DE ENGENHEIRO ELETRICISTA.

COMISSÃO EXAMINADORA

PROF. DR. EDUARDO PARENTE RIBEIRO

PROF. DR. GIDEON VILLAR LEANDRO

PROF. DR. GUSTAVO OLIVEIRA

CURITIBA, AGOSTO DE 2011

AGRADECIMENTOS

Primeiramente á Deus pela saúde e iluminação nos meus trabalhos.

Aos meus familiares por todo apoio, amor e carinho recebido.

Ao meu orientador Prof. Gideon Villar Leandro pela orientação, motivação e assistência durante a execução deste trabalho.

Aos meus amigos que sempre tiveram ao meu lado.

A Casa do Estudante Luterano Universitário (CELU), um lugar que além de ter me acolhido, tive a oportunidade de conhecer muitas pessoas de diferentes lugares e culturas e fazer grandes amizades.

Aos meus colegas de turma pelo convívio.

RESUMO

Este trabalho teve como objetivo principal fazer o controle de um sistema de nível líquido utilizando como estratégia o controle PID sintonizado através das regras de Ziegler-Nichols e do ajuste automático por Redes Neurais Artificiais (RNA). Primeiramente, foi construída a estrutura física do sistema no LAR que é composta por uma planta de terceira ordem, um atuador, três sensores e um reservatório. A planta é constituída por um sistema de interação com três tanques acoplados. O atuador que alimenta o sistema é uma motobomba que está conectado diretamente ao reservatório. Os sensores medem a altura do nível dos tanques e são constituídos por um potenciômetro multivoltas, uma haste e uma bóia. O reservatório coleta a água que sai da planta, garantindo a realimentação do sistema. Em seguida, foi feita a modelagem matemática do sistema, determinando os parâmetros da planta, do atuador e dos sensores. Depois, foram projetados e simulados os controladores PID sintonizados pelas regras Ziegler-Nichols e por RNA. Na simulação, o controlador PID sintonizado pela RNA apresentou resultado satisfatório, mas a sintonia pelas regras Ziegler-Nichols fez o controle ficar instável, tanto para a primeira regra quanto para a segunda. Com esses problemas, foi feita a opção de projetar um controlador do tipo PI pelo segundo método de Ziegler-Nichols, mas teve desempenho inferior em comparação ao controlador neural. A ferramenta matemática utilizada para a modelagem do sistema e a simulação dos controladores se fez pelo uso do software MATLAB®. Finalmente os controladores foram embarcados de forma prática.

Palavras Chaves: Sistema de Tanques Acoplados, Controlador PID e Redes Neurais Artificiais.

ABSTRACT

This work have had as principal objective to do the control of the liquid level system utilizing as strategy the control PID Ziegler-Nichols tuning and adaptive PID control using neural networks. Initially, the physical structure composed by an actuator, three sensors and a reservoir was built. The plant is composed by an interaction system with three tanks connected. The motor-pump is the actuator that feeds the system and it is directly connected with the reservoir. The sensors measure the level of the tanks and they are composed by a multi-turn potentiometer, a rod and a buoy. A reservoir collects the water which flows out of the plant, guarantying the feedback of the system. The mathematic modeling of system was carried out, and parameters of the plant, actuator and sensors were determined. PID controllers were designed by rules of Ziegler-Nichols and Neural Networks and simulated. Based on the simulation, it was verified that Neural Networks tuning showed satisfactory results. Ziegler Nichols's rules tuning led control to stay unstable either using first rule or second rule. Then a PI controller was designed using Ziegler-Nichols second rule, but its performance was inferior in comparison with neural PID controller. The software MATLAB® was the mathematic tool utilized for the modeling of the system and for the simulation of the controllers. Finally, the controllers were embedded in a microcontroller.

Keywords: Coupled tanks system, PID controller and Artificial Neural Networks.

LISTA DE FIGURAS

Figura 1 - Representação esquemática do sistema de nível de líquido com interação	17
Figura 2 - Curva de descarga dos tanques do processo de calculo das resistências	20
Figura 3 - Motobomba e sua curva de calibração	22
Figura 4 - Circuito ganho de corrente	22
Figura 5 - a) potenciômetro multivoltas e b) modelo do sensor de nível	23
Figura 6 - Representação da rede neural multicamadas (MLP)	26
Figura 7 - Modelo não linear de um neurônio	27
Figura 8 - Função de ativação limiar	28
Figura 9 - Função ativação linear por partes.....	28
Figura 10 - Função de ativação logística	29
Figura 11 - Função ativação tangente hiperbólica	29
Figura 12 - Tipos de redes neurais	30
Figura 13 - Grafo de fluxo da saída do neurônio j	32
Figura 14 - Curva de resposta em forma de S	37
Figura 15 - Oscilação sustentada com período P_{cr} (P_{cr} é medido em segundos)	39
Figura 16 - Estrutura do controlador PID neural.....	42
Figura 17 - Malha de controle.....	44
Figura 18 - Resposta do modelo de referência quando aplicado o degrau unitário	47
Figura 19 - Diagrama de blocos do sistema.....	50
Figura 20 - Curva de resposta em S da planta quando aplicada um sinal degrau unitário na entrada e reta tangente ao ponto de inflexão.....	51
Figura 21 - Resposta da saída da planta com o controlador PID sintonizado pelo primeiro método Ziegler-Nichols	52
Figura 22 - Curva de resposta obtida com o ganho crítico K_{cr} para determinar o período crítico P_{cr}	52
Figura 23 - Resposta do controlador PI sintonizado pelo segundo método de Ziegler-Nichols na simulação.....	54
Figura 24 - Sinal de Controle do controlador PI na simulação	54
Figura 25 - Resposta simulada do controlador PIDNN na simulação	55
Figura 26 - Sinal de controle PIDNN na simulação.....	56
Figura 27 - Variação dos ganhos K_p , K_i e K_d na simulação	56

Figura 28 - Controle do processo de nível líquido	57
Figura 29 - Resultado do Controlador PI no experimento prático	58
Figura 30 - Sinal de controle do controlador PI no experimento prático	58
Figura 31 - Resultado do Controlador PIDNN no experimento prático.....	59
Figura 32 - Sinal de controle do controlador PIDNN no experimento prático	59
Figura 33 - Variação dos ganhos K_p , K_i e K_d no experimento prático	60

LISTA DE TABELAS

Tabela 1 - Parâmetros do modelo de identificação dos tanques	24
Tabela 2 - Regra de sintonia de Ziegler-Nichols baseada na resposta ao degrau da planta.....	38
Tabela 3 - Regra de sintonia de Ziegler-Nichols baseada no ganho crítico K_{cr} e no período crítico P_{cr}	39
Tabela 4 - Parâmetros iniciais para simulação do PIDNN	55

LISTA DE SIGLAS

RNA – Rede Neural Artificial

PI – Proporcional-Integral (Proporcional Integral)

PID – Proporcional-Integral-Derivative (Proporcional Intergral Derivativo)

PIDNN - Proporcional-Integral-Derivative-Neural-Networks

LAR – Laboratório de Automação e Robótica (UFPR)

SUMÁRIO

INTRODUÇÃO	13
1 MODELAGEM MATEMÁTICA DA PLANTA.....	16
1.1 INTRODUÇÃO.....	16
1.2 PLANTA	16
1.3 MODELO MATEMÁTICO DOS SENSORES E ATUADORES	21
1.3.1 MOTOBOMBA	21
1.3.2 SENSOR DE NÍVEL	22
2 REDES NEURAIS ARTIFICIAIS	25
2.1 INTRODUÇÃO.....	25
2.2 REDES NEURAIS ARTIFICIAIS	25
2.2.1 MODELO DE UM NEURÔNIO	27
2.2.2 FUNÇÃO ATIVAÇÃO DO NEURÔNIO	28
2.2.3 ARQUITETURA DE REDE.....	30
2.2.4 APRENDIZAGEM	30
2.2.5 TREINAMENTO.	31
2.2.6 ALGORITMO DE RETROPROPAGAÇÃO	31
3 CONTROLADORES PID	36
3.1 INTRODUÇÃO.....	36
3.2 REGRAS DE SINTONIA DE ZIEGLER-NICHOLS	36
3.2.1 PRIMEIRO MÉTODO	37
3.2.2 SEGUNDO MÉTODO	38
3.3 CONTROLADOR PIDNN	40
3.3.1 CONTROLADOR PID EM TEMPO DSCRETO	40
3.3.2 PROJETO DO CONTROLADOR RNN	41
3.3.3 EQUAÇÕES DE SITONIZAÇÃO DO PIDNN	42
3.3.4 CONTROLADOR PID ADAPTATIVO	44
3.3.5 EQUAÇÃO DE ATUALIZAÇÃO DOS GANHOS ONLINE	44
3.3.6 SISTEMA JACOBIANO	45
3.3.7 MODELO DE REFERÊNCIA	46
4 ANÁLISES E RESULTADOS OBTIDOS	49

4.1 INTRODUÇÃO	49
4.2 SIMULAÇÃO	50
4.2.1 REGRAS DE SINTONIA DE ZIEGLER-NICHOLS	50
4.2.1.1 RESULTADO DO PRIMEIRO MÉTODO	50
4.2.1.2 RESULTADO DO SEGUNDO MÉTODO	52
4.2.2 RESULTADO DO CONTROLADOR PIDNN	55
4.3 RESULTADO EXPERIMENTAL	57
4.3.1 RESULTADO DO CONTROLADOR PI	58
4.3.2 RESULTADO DO CONTROLADOR PIDNN	59
CONCLUSÃO	61
REFERÊNCIAS.....	62

INTRODUÇÃO

O processo de controle de tanques acoplados é de grande interesse tanto na área industrial quanto acadêmica, é principalmente aplicado nos ramos químicos, petroquímicos, celulose, alimentos e bebidas. Ao longo dos últimos anos, diversas técnicas têm sido utilizadas com o objetivo de melhorar o desempenho do controle do nível de líquido. Podem ser encontrados na literatura modelos de controladores utilizando desde técnicas mais clássicas como o PID até as mais avançadas como controladores nebulosos e modelos preditivos neurais (SOUTO, GINANI E BAUCHSPIESS, 2007).

Na área dos sistemas de controle de processos, os controladores PID são utilizados em grande escala pela indústria por proporcionar controle satisfatório em diversos sistemas de controle e por sua facilidade de compreensão e otimização. A maioria dos controladores PID é ajustada em campo, utilizando diferentes técnicas de ajuste. Como a maioria dos processos físicos é de natureza não linear, em muitas situações as regras de sintonia clássicas podem não proporcionar um controle eficaz do processo. As técnicas de ajuste clássicas têm suas limitações quando aplicadas em sistemas que possuem não linearidades e ruídos derivativos.

Para compensar essas deficiências foram desenvolvidas técnicas de controle adaptativo, com atenção especial aos neuro-controladores que tem sido aplicado ao controle de processos não lineares.

As redes neurais artificiais (RNA) têm despertado muito interesse da comunidade de controle por possuir a habilidade de resolver muitos problemas que não podem ser resolvidos por técnicas analíticas clássicas (HAN and LEE, 1999). Elas são utilizadas em diversos problemas que envolvem modelagem, controle e identificação de sistemas. Sua capacidade de adaptação permite que controladores sintonizados por RNAs se ajustem a eventuais mudanças na dinâmica do sistema.

O recente e considerável interesse em controladores neurais tem resultado em um número de diferentes técnicas usando redes neurais artificiais em projeto de sistemas de controle (SAMAD and SU, 1994).

EYNG (2008) propôs e empregou o uso de um controlador *feedforward-feedback*, baseado no modelo inverso de redes neurais, no controle de colunas de absorção do processo de produção de etanol por fermentação. O controlador neural proposto apresentou resultados

superiores em comparação ao controlador PID convencional comprovado pelos índices ITAE (integral do erro absoluto ponderada pelo tempo), IAE (integral do erro absoluto) e ISE (integral do quadrado do erro). O controlador neural proposto se mostrou robusto em relação a influência de incertezas nos sensores sobre o desempenho do sistema de controle. Para incertezas de 5 a 10% o controlador neural apresentou desempenho satisfatório enquanto o PID não conseguiu manter o sistema estável para incertezas superiores a 5%.

OLIVEIRA (2009) faz a avaliação do controlador neural NARMA-L2 e compara com um controlador PI clássico em um processo complexo de quatro tanques acoplados, a resposta característica deste tipo de processo não permite um bom desempenho em ampla faixa de operação com a utilização de controladores convencionais.

FILHO *et al* (2008) desenvolvem um controlador híbrido indireto neural, usando o modelo de Rede Neural Autoregressiva com Entradas Exógenas para o controle de nível de uma planta industrial existente na Universidade Potiguar. O controlador inteligente proposto se mostrou eficiente e em comparação com um controlador PID convencional apresentou melhor desempenho em relação ao tempo morto (“*delay*”), tempo de atraso e tempo de estabilidade (“*settling time*”) da planta. O controlador neural comportou-se também melhor na presença de perturbações devido à capacidade de aprendizagem da RNA, levando a uma resposta mais rápida.

De acordo com (CARRARA, 1997) que investiga o uso de redes neurais em controles de satélites artificiais, obtendo o modelo dinâmico direto e o inverso do sistema através da rede neural, pode-se comprovar que a rede neural possui a capacidade de compensação, capaz de assegurar robustez ao controle proposto.

RIBEIRO (2010) desenvolveu um controlador adaptativo do tipo regulador auto-ajustável (*STR – Self Tuning Regulator*), fez sua implementação e comparou seu desempenho com um controlador PID (proporcional-integrativo-derivativo) e dois controladores preditivos: um baseado em redes neurais artificiais e um controlador DMC (*Dynamic Matrix Control*). O controlador adaptativo mostrou bons resultados em relação às outras técnicas utilizadas no trabalho. A capacidade de ajuste dos parâmetros do controlador torna-se uma estratégia promissora para sistemas que sofrem com alterações contínuas em suas variáveis de processo ou mudanças de *setpoint*.

De acordo com OLIVEIRA *et al* (1997) que fazem aplicação experimental da técnica *gain scheduling* para o controle de altura de nível de tanques interativos descrito por modelo

dinâmico não linear, o controlador neural global pode substituir os diversos controladores sintonizados para os diferentes pontos de operação da planta.

Este trabalho teve como objetivos a construção de uma planta composta por um sistema de três tanques acoplados; a modelagem matemática da planta; a identificação dos parâmetros do modelo; o projeto de controladores PID sintonizado por redes neurais artificiais e pelos métodos de Ziegler-Nichols embarcando-os de forma prática.

Com as duas técnicas de sintonia dos controladores PID foi realizado uma comparação do desempenho deles no processo de controle do nível de líquido dos três tanques acoplados.

Esta monografia está estruturada em cinco capítulos. No primeiro capítulo é apresentado o sistema a ser controlado, assim como é feito a sua modelagem matemática e dos demais dispositivos que interferem em sua dinâmica. O segundo capítulo apresenta as redes neurais artificiais, bem como suas aplicações, sua estrutura e base teórica. O terceiro capítulo apresenta as técnicas utilizadas neste trabalho para sintonia do controlador PID. O quarto capítulo apresenta e faz análise dos resultados obtidos para o controle da planta e finalmente o quinto capítulo faz as conclusões referentes ao trabalho desenvolvido.

CAPÍTULO 1

MODELAGEM MATEMÁTICA DA PLANTA

1.1 INTRODUÇÃO

A modelagem matemática de um sistema real é uma forma de representar um fenômeno físico por equações matemáticas. As técnicas de modelagem podem ser classificadas em três categorias denominadas *modelagem caixa branca*, *modelagem caixa preta* e *modelagem caixa cinza* (AGUIRRE, 2007).

Modelagem caixa branca: é necessário conhecer bem o sistema em estudo, bem como as leis físicas que descrevem o sistema a ser modelado.

Modelagem caixa preta: utiliza técnica de identificação de sistemas e uma de suas características é utilizar pouco ou nenhum conhecimento prévio do sistema.

Modelagem caixa cinza: é uma categoria intermediária entre modelagem caixa branca e caixa preta. As técnicas desse grupo se caracterizam por usar informação auxiliar que não se encontra no conjunto de dados utilizados durante a identificação.

A técnica de modelagem utilizada neste trabalho é a modelagem caixa cinza.

1.2 PLANTA

A dinâmica do sistema pode ser controlada através da vazão de entrada no tanque1 e das resistências dos canos de comunicação entre os tanques.

A modelagem deste sistema foi obtida utilizando o princípio de conservação: a vazão volumétrica de entrada q_e menos a vazão volumétrica de saída q_s é igual à variação de volume acumulada no tanque.

$$\frac{dV}{dt} = q_e - q_s \quad (1.1)$$

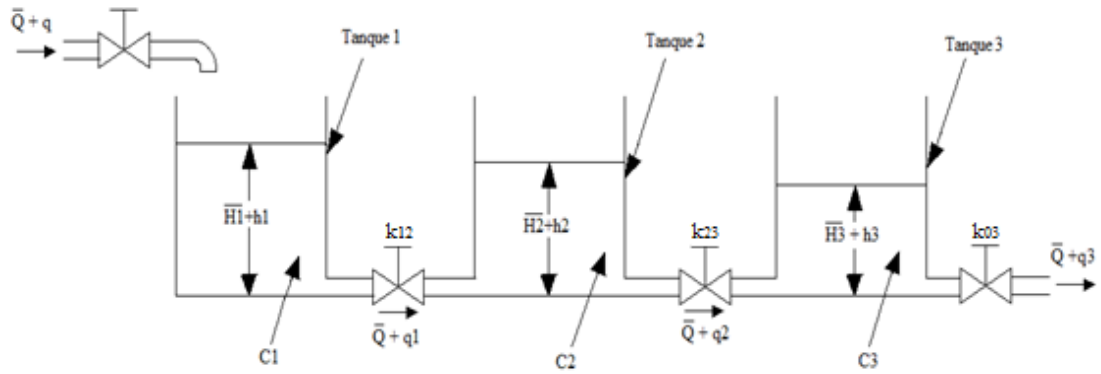


Figura 1: Representação esquemática do sistema de nível de líquido com interação.

Na figura 1 as alturas e as vazões são definidas:

\bar{Q} = vazão em volume em regime permanente em $[m^3/s]$.

$\bar{H}1$, $\bar{H}2$ e $\bar{H}3$ = nível de líquido dos tanques 1, 2 e 3 em regime permanente em $[m]$.

q = entrada no tanque 1 em $[m^3/s]$.

q_1 e q_2 = vazões entre os tanques 1-2 e 2-3 em $[m^3/s]$.

q_3 = vazão de saída do tanque 3 em $[m^3/s]$.

$h1$, $h2$ e $h3$ = altura do nível de água nos tanques: 1, 2 e 3.

$k12$, $k23$ e $k03$ = parâmetros das válvulas $[m^2/s]$.

Para determinar os parâmetros do sistema, inicialmente foi determinada a natureza do escoamento do sistema se é laminar ou turbulento através do coeficiente de Reynolds. O escoamento é laminar se tiver o coeficiente de Reynolds menor que 2000 e turbulento com número de Reynolds maior que 2400. Foi verificado que o escoamento entre os três tanques é laminar. Logo, a equação de vazão instantânea de saída de líquido para o processo de um tanque é dada por:

$$q = k \cdot h \quad (1.2)$$

Sendo q a vazão de saída do tanque em $[m^3/s]$, k o parâmetro da válvula em $[m^2/s]$ e h a altura do nível em relação à válvula em $[m]$. Logo, a resistência para escoamento laminar pode ser obtida como:

$$R = \frac{dH}{dQ} = \frac{H}{Q} \quad (1.3)$$

Para a formulação da equação da planta, para o sistema pode-se obter as seguintes equações:

$$\frac{h_1 - h_2}{R_1} = q_1 \quad (1.4)$$

$$C_1 \frac{dh_1}{dt} = q - q_1 \quad (1.5)$$

$$\frac{h_2 - h_3}{R_2} = q_2 \quad (1.6)$$

$$C_2 \frac{dh_2}{dt} = q_1 - q_2 \quad (1.7)$$

$$\frac{h_3}{R_3} = q_3 \quad (1.8)$$

$$C_3 \frac{dh_3}{dt} = q_2 - q_3 \quad (1.9)$$

Combinando as equações de 1.4 a 1.9 e como o nível do terceiro tanque é controlado em função da alimentação de entrada do primeiro tanque, a função transferência do sistema fica:

$$\frac{H_3(s)}{Q(s)} = \quad (1.10)$$

$$\frac{R_3}{C_1 R_1 C_2 R_2 C_3 R_3 s^3 + (C_1 R_1 C_2 R_2 + C_1 R_1 C_3 R_3 + C_2 R_2 C_3 R_3 + C_1 R_1 C_2 R_3 + C_1 R_2 C_3 R_3) s^2 + (C_1 R_1 + C_2 R_2 + C_3 R_3 + C_2 R_3 + C_1 R_2 + C_1 R_3) s + 1}$$

Através de experimentos e medição, as resistências e as capacitâncias do modelo do sistema foram determinadas. Inicialmente foram determinadas as capacitâncias C_1 , C_2 e C_3 que correspondem às áreas de secção de cada tanque.

Logo:

$$C_1 = 0.01128 \text{ m}^2$$

$$C_2 = 0.0083 \text{ m}^2$$

$$C_3 = 0.0058 \text{ m}^2$$

Para determinar o valor das resistências foram feitos experimentos. O processo consiste em encher de água até o nível máximo os tanques e em seguida esvaziar-los obtendo valores de altura de nível e vazão, permitindo calcular sua resistência.

Primeiramente foi calculada a resistência R_3 responsável pelo controle do nível do tanque 3, a válvula k_{23} foi fechada permitindo o esvaziamento do tanque através da válvula k_{03} . A resistência R_2 responsável pelo controle dos níveis dos tanques 2 e 3 foi determinada com a válvula k_{12} fechada e as válvulas k_{23} e k_{03} abertas permitindo o esvaziamento dos tanques 1 e 2 sobre as mesmas. A resistência R_1 responsável pelo controle dos tanques 1 e 2 foi determinada com todas as válvulas abertas permitindo o esvaziamento dos tanques 1 e 2.

As equações que medem as resistências R_1 , R_2 e R_3 :

$$R_1 = \frac{h_1 - h_2}{q_1} \quad (1.11)$$

$$R_2 = \frac{h_2 - h_3}{q_2} \quad (1.12)$$

e

$$R_3 = \frac{h_3}{q_3} \quad (1.13)$$

Os valores medidos no experimento dos tanques foram coletados por sensores de nível e transmitidos via serial para um computador. Para o cálculo médio das resistências dos tanques foi considerada uma faixa de valores intermediários com o sensor em movimento evitando pegar valores próximos aos limites da faixa de medição dos sensores devido à presença de atrito.

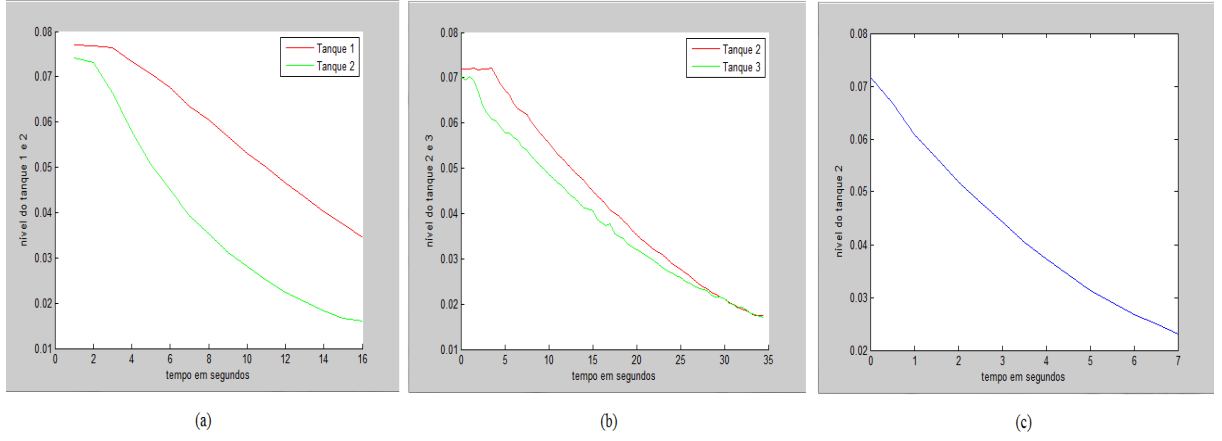


Figura 2: Curva de descarga dos tanques do processo de cálculo das resistências. a) Curvas de descarga do tanque 1 e 2 para obtenção de R_1 , b) Curvas de descarga do tanque 2 e 3 para obtenção de R_2 e c) Curvas de descarga do tanque 3 para obtenção de R_3 .

Logo os valores médios das resistências valem:

$$R_1 \approx 300 [\text{m}^{-1}\text{s}^{-1}], R_2 \approx 150 [\text{m}^{-1}\text{s}^{-1}] \text{ e } R_3 \approx 1050 [\text{m}^{-1}\text{s}^{-1}]$$

Logo a equação da planta fica:

$$H(s) = \frac{1050}{25.7s^3 + 72.31s^2 + 24.28s + 1} \quad (1.14)$$

A forma recursiva da planta é utilizada para a simulação do processo de controle. Para encontrar a equação a diferenças no domínio do tempo discreto é utilizada a representação por segurador de ordem zero para que os pontos da função de tempo discreto sejam equivalentes à função de tempo contínuo nos instantes de amostragens.

A representação discreta da planta por segurador de ordem zero é obtida pelo seguinte procedimento:

i.

$$H_a(s) = \frac{1}{s} H(s) \quad (1.15)$$

ii.

$$h_a(t) = \mathcal{L}^{-1}\{H_a(s)\} \quad (1.16)$$

iii.

$$h_a(k\Delta t) = h_a(t)|_{t=k\Delta t} \quad (1.17)$$

iv.

$$H_a(z) = Z\{h_a(k)\} \quad (1.18)$$

v.

$$H(z) = \frac{z-1}{z} H_a(z) \quad (1.19)$$

Com $\Delta t = 1s$, a planta no domínio z fica:

$$H(z) = \frac{y(z)}{u(z)} = \frac{4.228z^2 + 8.639z + 0.8461}{z^3 - 1.718z^2 + 0.7674z - 0.03681} \quad (1.20)$$

E sua forma recursiva:

$$\begin{aligned} y(n) = & 1.718y(n-1) - 0.7674y(n-2) + 0.03681y(n-3) + 4.228u(n-1) \\ & + 8.639u(n-2) + 0.8461u(n-3) \end{aligned} \quad (1.21)$$

1.3 MODELO MATEMÁTICO DOS SENSORES E ATUADORES

1.3.1 MOTOBOMBA

A motobomba é o atuador do sistema, ela funciona com tensão de 0 a 12 Volts, seu acionamento é feito através de um circuito de potência e comandado pela saída analógica do microcontrolador. O software converte o sinal de controle de vazão para um valor em tensão aplicado sobre a bomba que por sua vez gera a vazão correspondente ao sinal controle do microcontrolador.

Foi realizada experimentalmente a identificação da motobomba para a geração do sinal de controle. A curva obtida foi aproximada por uma equação de segundo grau. Para melhor aproximação foram excluídos os valores de tensão inferiores a 2 Volts e valores na região de saturação superiores a 10 Volts.

A equação que relaciona a tensão aplicada sobre a bomba em função de sua vazão:

$$V = 0.0002726q^2 + 0.0493832q + 0.9243 \quad (1.22)$$

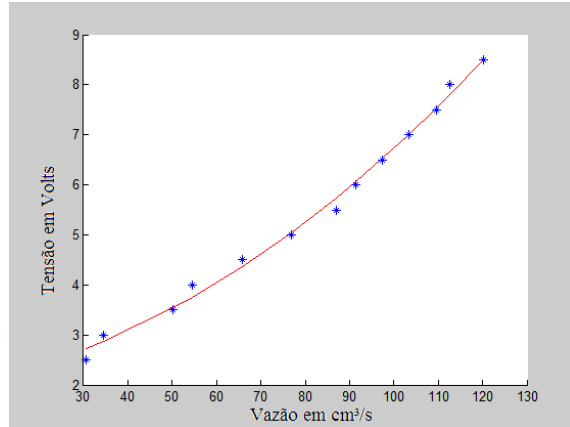


Figura 3: Motobomba e sua curva de calibração.

O controle do atuador do sistema foi realizado através de um circuito de potência comandado através da saída analógica do microcontrolador representado na figura 4.

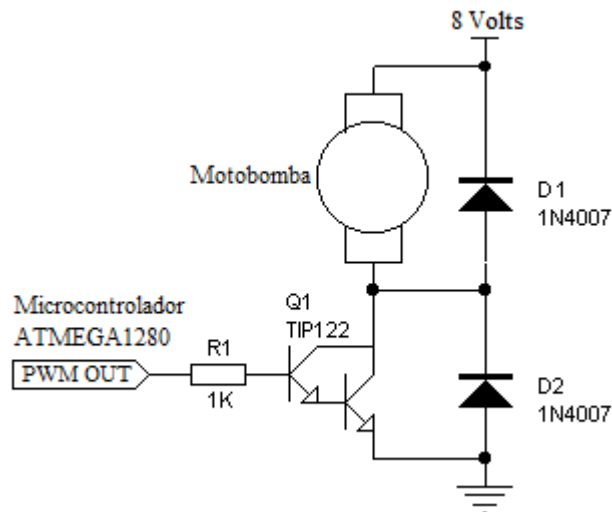


Figura 4: Circuito ganho de corrente.

1.3.2 SENSOR DE NÍVEL

Os sensores de nível são instalados nos três tanques para o processo de modelagem da planta, mas no processo de controle é utilizado somente o sensor do terceiro tanque. Eles são constituídos por um potenciômetro multivoltas de 100 k Ω , uma haste de comprimento h e uma bóia como mostrado na figura n. A medida do nível de cada tanque se fez em função do ângulo de giro do potenciômetro θ .

O θ foi determinado através da queda de tensão no potenciômetro V_p a partir de um divisor de tensão (eq. 1.23), alimentado por uma fonte de tensão de 5 Volts, com o potenciômetro R_p em série com um resistor 1 k Ω .

$$V_p = \frac{5R_p}{1000 + R_p} \quad (1.23)$$

Como cada volta no potenciômetro corresponde a uma variação de $10k\Omega$, pode ser obter a relação de θ em função de R_p .

$$\theta = \frac{360^\circ R_p}{1000} \quad (1.24)$$

Com as equações 1.23 e 1.24, tem-se θ em função de V_p :

$$\theta = \frac{360^\circ V_p}{5 - V_p} \quad (1.25)$$

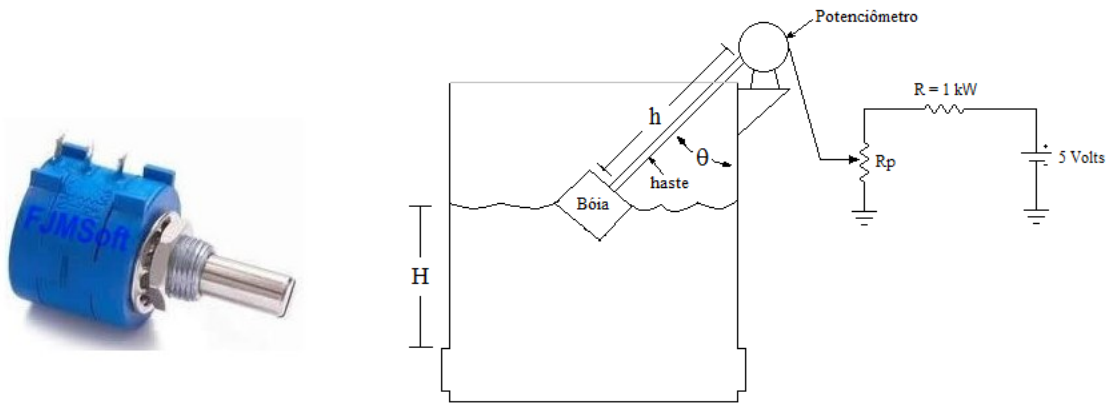


Figura 5: a) potenciômetro multivoltas e b) modelo do sensor de nível.

Como cada tanque possui dimensões diferentes, os sensores de nível foram calibrados pela seguinte modelo de identificação:

$$H = C + h. (1 - \cos(\theta' + \theta)) \quad (1.26)$$

Onde H é altura do nível no tanque medido, θ' corresponde ao ângulo formado pela haste e a parede do tanque quando o sensor indica o nível mínimo e C é uma constante que corresponde ao valor do offset. Com os valores de H e θ obtidos experimentalmente foram identificados os valores C , h e θ' . Os valores de θ' e C não foram medidos diretamente porque dependem do centro massa conjunto haste-bóia.

A tabela 1 mostra os valores da altura do nível no tanque (H), o ângulo formado pela haste e a parede do tanque quando o sensor indica o nível mínimo θ' e a constante que corresponde ao valor do offset (C).

Tabela 1: Parâmetros do modelo de identificação dos tanques.

	C	H	θ'
Tanque 1	- 0.1878 cm	10.63 cm	28.07°
Tanque 2	0.73987 cm	10.226 cm	23.8°
Tanque 3	1.37 cm	10.88 cm	19.17°

CAPÍTULO 2

REDES NEURAIIS ARTIFICIAIS

2.1 INTRODUÇÃO

A grande capacidade de processamento de informações do cérebro humano tem motivado pesquisas no sentido de encontrar modelos que reproduzam suas características computacionais, possibilitando, desta forma, que se realizem certas tarefas de uma maneira semelhante ao cérebro humano.

As redes neurais artificiais (RNA) foram desenvolvidas tomando-se como base o cérebro humano, ou seja, são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através da experiência (ROCHA, 2006).

HAYKIN (2001) menciona que uma rede neural é uma “máquina” projetada para modelar, de forma simplificada, a maneira como o cérebro realiza uma tarefa particular ou função de interesse; a rede é normalmente implementada utilizando-se componentes eletrônicos ou é simulada por programação computacional.

Assim, o intuito deste capítulo é apresentar, de forma sucinta, as características principais das redes neurais artificiais. Para tanto, inicia-se com uma breve introdução as redes neurais; em seguida é descrito o modelo de um neurônio artificial. O capítulo continua com a relação das arquiteturas e o aprendizado/treinamento de uma rede neural artificial.

2.2 REDES NEURAIIS ARTIFICIAIS (RNA)

As redes neurais artificiais são sistemas computacionais de processamento paralelo e distribuído baseado no modelo biológico do neurônio ou sistema nervoso, têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso. O poder computacional da rede neural depende de sua estrutura paralelamente distribuída e sua habilidade de aprendizagem. Essas capacidades tornam as redes neurais capazes de resolver problemas complexos.

Ela se assemelha ao cérebro em dois aspectos:

- Conhecimento adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem.

- Forças de conexão entre neurônios, conhecidas como pesos sinápticos, são utilizadas para armazenar conhecimento adquirido.

Uma representação de rede neural pode ser vista na figura 6.

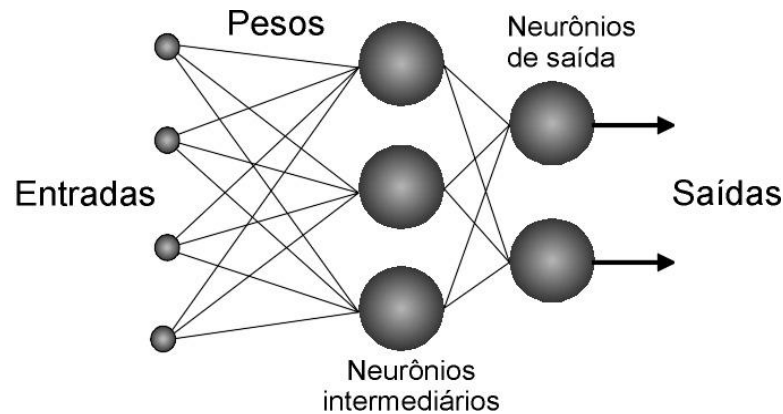


Figura 6: Representação da rede neural multicamadas (MLP).

A partir de um algoritmo de aprendizado é efetuado a aprendizagem da rede neural, onde os pesos sinápticos são atualizados durante a execução da mesma com a finalidade de se alcançar um resultado desejado (HAYKIN, 2003).

Dentre as principais propriedades das redes neurais pode se destacar:

- **Aprendizado:** a rede é treinada a partir de exemplos.
- **Não linearidade:** um neurônio artificial pode ser linear ou não-linear, sua aplicação é adequada para sistemas físicos que geralmente são não lineares.
- **Capacidade de adaptação:** apresenta a propriedade de se adaptar a nova informação.
- **Capacidade de generalização:** uma rede neural possui uma capacidade de generalização do conhecimento adquirido a partir da existência de dados incompletos ou imprecisos, uma vez que pode preencher lacunas sem sofrer degradação, em uma referência muito próxima à interpolação e extrapolação no campo da estatística, mas com uma execução muito diferente.
- **Informação Contextual:** O conhecimento adquirido é representado pela estrutura neural e o estado de ativação dos neurônios da rede. Como cada neurônio é afetado de todos os neurônios da rede a informação contextual é tratada de forma natural.

- Tolerância a Falhas: é capaz de realizar computação robusta sob condições de operação adversas, onde seu desempenho degrada suavemente por estar relacionado ao processamento paralelo e distribuído da rede.

2.2.1 MODELO DE UM NEURÔNIO

Um neurônio é uma unidade de processamento de informação que é fundamental para a operação de uma rede neural (HAYKIN, 2003).

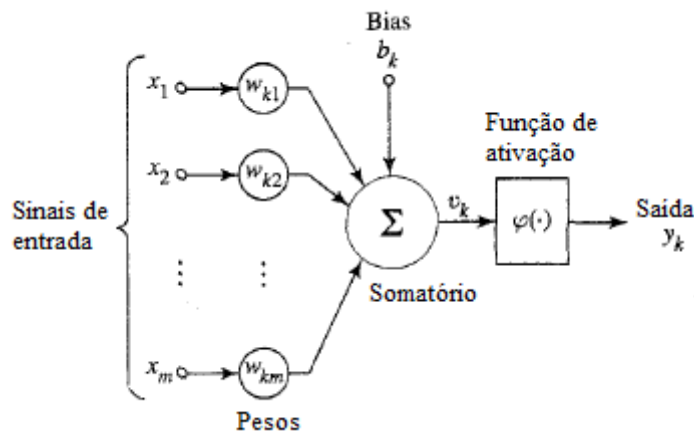


Figura 7: Modelo não linear de um neurônio.

Este modelo de neurônio pode ser expressa de forma matemática:

$$u_k = \sum_{j=1}^m w_{kj} x_j \quad (2.1)$$

$$v_k = u_k + b_k \quad (2.2)$$

$$y_k = \varphi(v_k) \quad (2.3)$$

Onde x_1, x_2, \dots, x_m são os sinais de entrada, $w_{k1}, w_{k2}, \dots, w_{km}$ são os pesos sinápticos dos neurônios, u_k é a resposta da multiplicação dos pesos com os sinais de entrada, b_k é o *bias* que é um parâmetro externo do neurônio, $\varphi(\cdot)$ é a função de ativação, v_k potencial de ativação do neurônio ou campo local induzido e y_k é a resposta de saída do neurônio.

2.2.2 FUNÇÃO ATIVAÇÃO DO NEURÔNIO

A função de ativação é a resposta de saída de cada neurônio. Existem três tipos de funções de ativação básicas (HAYKIN, 2001): *Função de Limiar*, *Função linear por Partes* e *Função Sigmóide*.

Função de Limiar: Este tipo de função de ativação é descrito como:

$$\varphi(v) = \begin{cases} 1 & \text{se } v \geq 0 \\ 0 & \text{se } v < 0 \end{cases} \quad (2.4)$$

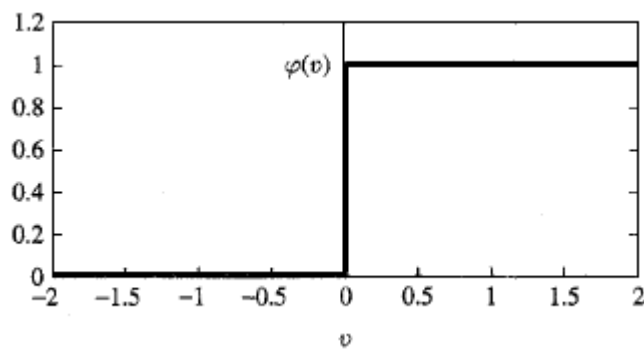


Figura 8: Função de ativação limiar.

Função Linear por Partes: A função linear por partes pode ser descrita como na figura 9.

$$\varphi(v) = \begin{cases} 1, & v \geq \frac{1}{2} \\ v, & -\frac{1}{2} < v < \frac{1}{2} \\ 0, & v \leq -\frac{1}{2} \end{cases} \quad (2.5)$$

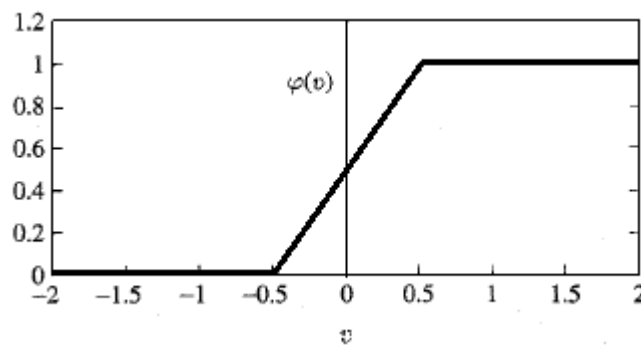


Figura 9: Função ativação linear por partes.

O fator de amplificação é assumido igual a 1 dentro da região linear. Esta função é utilizada neste trabalho na estrutura do controlador PIDNN, mas as regiões de saturação são limitadas com os valores de -1 e 1 .

Função Sigmóide: Apresenta um comportamento em formato de S em seu gráfico, é forma mais utilizada na construção de redes neurais artificiais. Ela apresenta balanceamento adequado entre o comportamento linear e não linear e é uma função diferencial.

A função logística e a função tangente hiperbólica são exemplos de funções sigmóides.

Função logística:

$$\varphi(n) = \frac{1}{1 + e^{(-av)}} \quad (2.6)$$

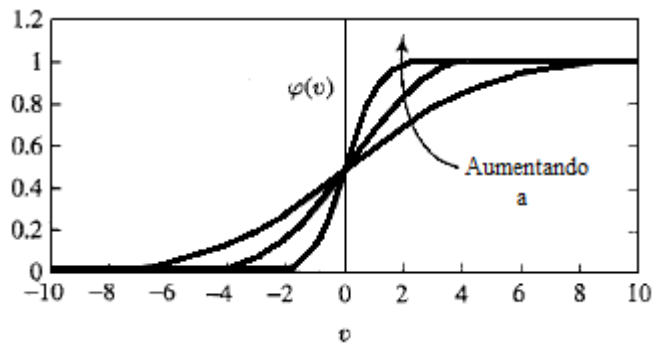


Figura 10: Função de ativação logística.

Função tangente hiperbólica:

$$\varphi(n) = \frac{2}{1 + e^{(-2av)}} - 1 \quad (2.7)$$

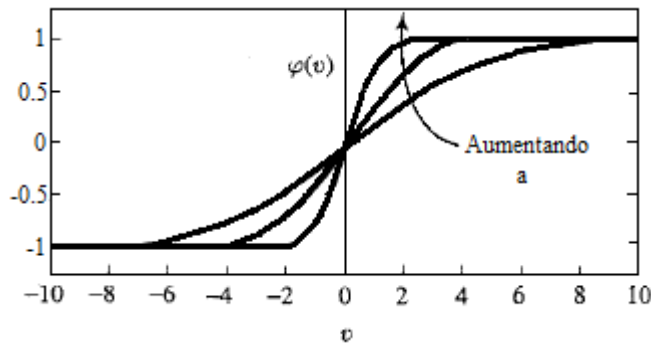


Figura 11: Função ativação tangente hiperbólica.

2.2.3 ARQUITETURA DE REDE

A forma pela qual os neurônios de uma rede neural estão estruturados está relacionada com a forma de aprendizagem usada no treinamento da rede. As redes neurais podem ser classificadas basicamente em três classes de arquiteturas diferentes: *feedforward*, *feedback* e *auto-organizáveis*.

Redes neurais *feedforward*: São redes que propagam o sinal somente no sentido positivo, ou seja, a propagação do sinal ocorre da entrada para a saída. Para redes pertencentes nesta classe, as saídas de seus neurônios podem somente se conectar com neurônios pertencentes da próxima camada. São exemplos de redes dessa arquitetura: perceptron de uma camada, perceptron multicamadas e RN função de base radial.

Redes neurais *feedback*: São redes recorrentes que possuem pelo menos um laço de realimentação em sua estrutura. Exemplo de rede: redes hopfield. Neste trabalho o controlador PID neural utiliza este tipo de configuração.

Redes neurais *auto-organizáveis*: São redes que possuem a capacidade de utilizar experiências passadas para adaptar-se às mudanças imprevisíveis. Exemplo: mapas auto-organizáveis de Kohonen.

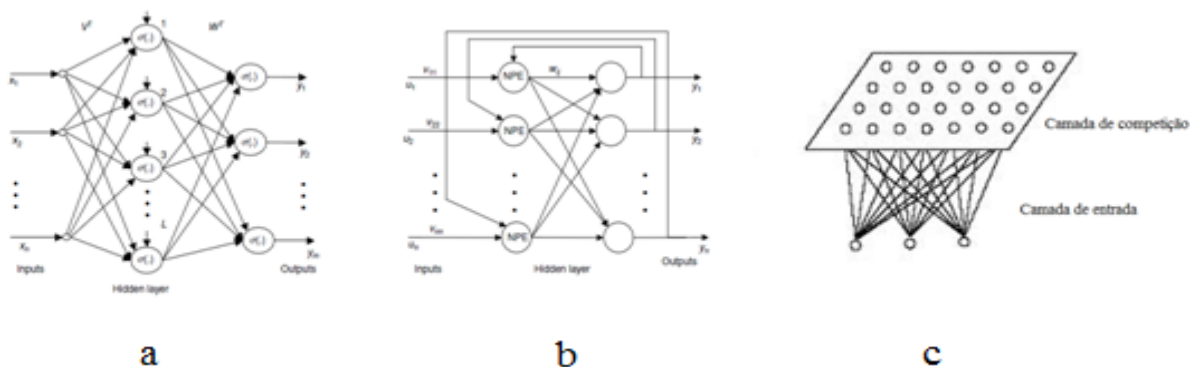


Figura 12: Tipos de redes neurais. a) multicamadas (*feedforward*), b) hopfield (*feedback*) e c) mapa de Kohonen (*auto-organizável*).

2.2.4 APRENDIZAGEM

A aprendizagem é um processo pelo qual os parâmetros livres de uma rede neural são adaptados através de um processo de estimulação pelo ambiente no qual está inserida. O tipo de aprendizagem é determinado pela maneira pela qual a modificação dos parâmetros ocorre (HAYKIN, 2001).

Aprendizagem supervisionada: o ajuste dos pesos sinápticos ocorre com a utilização de um professor ou um exemplo a ser seguido. A rede neural recebe um conjunto de entradas

e saídas padronizadas onde o ajuste dos pesos ocorre em função da minimização do erro entre o sinal desejado (o conjunto de saídas padronizadas) e a resposta de saída da rede neural (quando aplicada o conjunto de entradas padronizadas).

Aprendizagem não supervisionada: não há um professor externo, o aprendizado é constituído por propriedades determinadas no conjunto de dados pela rede neural.

Aprendizagem Híbrida: neste tipo de configuração ocorre uma mistura dos aprendizados supervisionados e não supervisionados.

A estrutura neural do controlador PID deste trabalho utiliza aprendizagem supervisionada pela regra por correção do erro.

2.2.5 TREINAMENTO

O treinamento é o processo de ajuste dos parâmetros da rede. Existem dois modos de treinamentos: sequencial e por lote.

Modo sequencial: também é conhecido como modo on-line, onde a atualização dos pesos é realizada após a apresentação de cada exemplo de treinamento, é o modo de treinamento utilizado neste trabalho.

Modo por Lote: o ajuste dos pesos é realizado após a apresentação de todos os exemplos de treinamento que constituem uma interação.

2.2.6 ALGORITMO DE RETROPROPAGAÇÃO

Este algoritmo de treinamento consiste em dois passos através das diferentes camadas da rede: um *passo a frente*, a propagação, e um *passo para trás*, a retropropagação. No *passo a frente* um sinal de entrada é aplicado na rede que se propaga de camada por camada e um sinal real na saída da rede é gerado, na fase de propagação os valores dos pesos são mantidos fixo. No passo para trás, os pesos sinápticos da rede são ajustados de acordo com uma regra de aprendizagem por correção do erro.

A partir da regra de correção de erro, o sinal de erro na saída do neurônio j (Figura 13) é definido por:

$$e_j(n) = d_j(n) - y_j(n) \quad (2.8)$$

Onde $y_j(n)$ é a resposta se saída do neurônio e $d_j(n)$ a resposta desejada, na interação n . Os neurônios da camada de saída são os únicos que podem ter o erro calculado diretamente.

Se a energia do erro para o neurônio j é definido como $e_j^2(n)/2$, o valor instantâneo $\mathcal{E}(n)$ da energia total do erro que é a soma de todos os termos $e_j^2(n)/2$ dos neurônios da camada de saída. Logo pode ser obter a seguinte expressão:

$$\mathcal{E}(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n) \quad (2.9)$$

Onde o conjunto C inclui todos os neurônios da camada de saída da rede. Com o número total de exemplos N contido C , pode ser obter a energia média do erro quadrado \mathcal{E}_{med} em relação ao tamanho do conjunto N .

$$\mathcal{E}_{med} = \frac{1}{N} \sum_{n=1}^N \mathcal{E}(n) \quad (2.10)$$

A energia média do erro quadrado representa a função de custo que pode ser uma medida de desempenho de aprendizagem, logo o ajuste dos pesos resulta na minimização de \mathcal{E}_{med} .

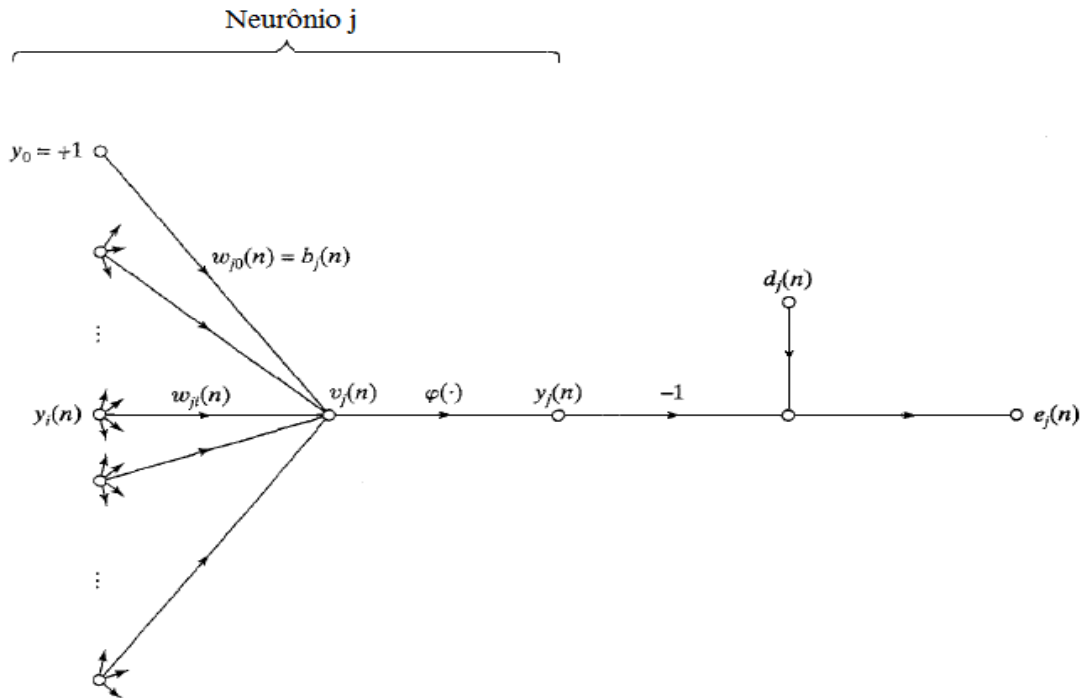


Figura 13: Grafo de fluxo da saída do neurônio j .

O campo local induzido $v_j(n)$ é a entrada da função de ativação, ele é obtido por:

$$v_j(n) = \sum_{i=0}^m w_{ji}(n) y_i(n) \quad (2.11)$$

Onde m é o número total de entradas (sem bias) aplicadas no neurônio j . Logo o sinal na saída $y_j(n)$ do neurônio j na interação n é:

$$y_j(n) = \varphi_j(v_j(n)) \quad (2.12)$$

A correção do peso $\Delta w_{ji}(n)$ pode ser obtida através da derivada parcial $\partial \mathcal{E}(n)/\partial w_{ji}(n)$, aplicando a regra da cadeia do cálculo:

$$\frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} = \frac{\partial \mathcal{E}(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial w_{ji}(n)} \quad (2.13)$$

A direção de busca no espaço dos pesos é determinado pela derivada parcial $\partial \mathcal{E}(n)/\partial w_{ji}(n)$ que representa um fator de sensibilidade.

Diferenciando a equação 2.9 em ambos os lados em relação à $e_j(n)$ obtém:

$$\frac{\partial \mathcal{E}(n)}{\partial e_j(n)} = e_j(n) \quad (2.14)$$

Diferenciando a equação 2.8 em ambos os lados em relação à $y_j(n)$ obtém:

$$\frac{\partial e_j(n)}{\partial y_j(n)} = -1 \quad (2.15)$$

Diferenciando a equação 2.12 em ambos os lados em relação à $v_j(n)$ obtém:

$$\frac{\partial y_j(n)}{\partial v_j(n)} = \varphi'(v_j(n)) \quad (2.16)$$

Onde o uso do apostrofe significar diferenciar em relação ao argumento. Finalmente diferenciando a equação 2.11 em relação à $w_{ji}(n)$ produz:

$$\frac{\partial v_i(n)}{\partial w_{ji}} = y_i(n) \quad (2.17)$$

Com uso das equações 2.14 a 2.17 em 2.13 obtém:

$$\frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} = -e_j(n) \varphi' \left(v_j(n) \right) y_i(n) \quad (2.18)$$

A partir da regra do delta define a correção Δw_{ji} aplicada $w_{ji}(n)$:

$$\Delta w_{ji}(n) = -\eta \frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} \quad (2.19)$$

Onde η é a taxa de aprendizagem do algoritmo de retropropagação. O sinal negativo da equação indica a descida do gradiente no espaço de pesos. O uso da equação 2.18 em 2.19 resulta:

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n) \quad (2.20)$$

Onde $\delta_j(n)$ o gradiente local é definido por:

$$\delta_j(n) = e_j(n) \varphi' \left(v_j(n) \right) \quad (2.21)$$

Há dois casos distintos para a determinação do gradiente local de um neurônio: quando o neurônio j é um nó de saída ou um nó oculto. Quando o neurônio j é um nó de saída, o gradiente local é determinado diretamente pelo erro e pode ser calculado utilizando a equação 2.21. Quando o neurônio j é um nó oculto, o sinal do erro para um neurônio oculto deve ser determinado de forma recursiva, o algoritmo de retropropagação torna-se complexo pois leva em conta os sinais de erro de todos os neurônios que estão diretamente conectado com o neurônio oculto. A equação do gradiente local para o caso do nó oculto não é deduzida

neste trabalho porque os pesos sinápticos da camada oculta da estrutura neural do controlador são contates que correspondem aos parâmetros do controlador PID digital.

CAPÍTULO 3

CONTROLADORES PID

3.1 INTRODUÇÃO

Em 95% dos processos industriais o controlador PID é utilizado, onde a maioria dos Controladores PID é ajustada em campo. As regras de sintonia proposta na literatura como Ziegler-Nichols e Cohen-Coon podem permitir ajustes fino em campo. Há também técnicas de ajuste automático que estão sendo desenvolvidas e permite fazer sintonia online como os controladores PID adaptativos. Estruturas modificadas como I-PD e controle PID com vários graus de liberdade estão em uso na indústria.

O controlador PID pode ser representado pela seguinte equação:

$$G_c(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (3.1)$$

Onde K_p é o ganho proporcional, T_i é tempo integral e T_d é o tempo derivativo.

Apesar do controlador PID ser de fácil entendimento e aplicação, pode apresentar algumas limitações relacionadas com problemas de sintonia ou não linearidades no sistema a ser controlado.

3.2 REGRAS DE SINTONIA DE ZIEGLER-NICHOLS

Pode se determinar parâmetros de um controlador que atendam às especificações do comportamento do sistema a malha fechada, aplicando várias técnicas de sintonia quando se tem um modelo matemático do processo.

O processo de selecionar parâmetros do controlador que garantam dada especificação de desempenho é conhecido como sintonia de controlador (OGATA, 2011).

Ziegler e Nichols propuseram regras para a sintonia de controladores PID (ajuste dos valores K_p , T_i e T_d) baseadas em uma ação proporcional no valor K_p que resulta em uma estabilidade marginal ou na resposta experimental ao degrau.

Existem duas regras de sintonia de Ziegler-Nichols:

3.2.1 PRIMEIRO MÉTODO

A resposta da planta é obtida experimentalmente aplicando um sinal de degrau unitário na entrada. Para que este método seja aplicável a curva de resposta ao degrau deve possuir um aspecto de um S , ou seja, a planta não deve possuir integradores ou pólos complexos conjugados dominantes.

Duas constantes, o atraso L e a constante de tempo T podem caracterizar a curva de formato S . Essas constantes são determinadas desenhando-se uma linha tangente no ponto de inflexão da curva de formato S e em seguida determinado a intersecção da linha tangente com o eixo dos tempos e a linha K referente ao ganho da resposta da planta, de acordo com a figura 14.

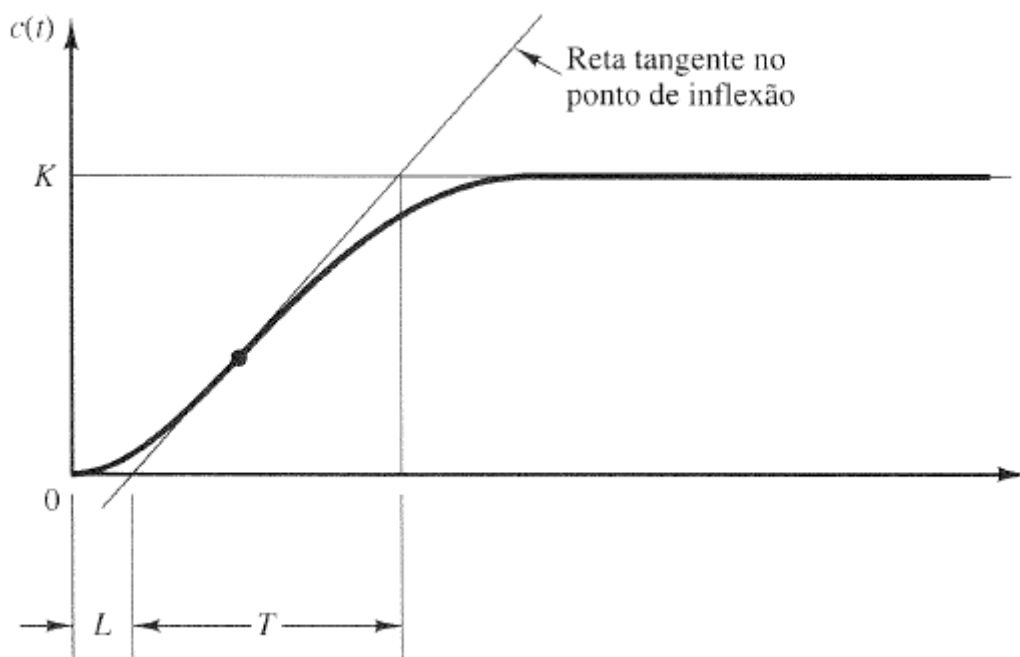


Figura 14: Curva de resposta em forma de S .

Para determinar os ganhos (K_p , T_i e T_d) dos controladores do tipo P, PI e PID, Ziegler e Nichols sugeriram utilizar a tabela 2 para a sintonia.

Tabela 2: Regra de sintonia de Ziegler-Nichols baseada na resposta ao degrau da planta.

Tipo de controlador	K_p	T_i	T_d
P	T/L	∞	0
PI	$0,9(T/L)$	$L/0,3$	0
PID	$1,2(T/L)$	$2L$	$0,5L$

Com o primeiro método de Ziegler-Nichols a formula controlador PID pode ser simplificado por:

$$= 1.2 \frac{T}{L} \left(1 + \frac{1}{2L} s + 0.5Ls \right) \quad (3.2)$$

$$= 0.6T \frac{\left(s + \frac{1}{L} \right)^2}{s} \quad (3.3)$$

Este controlador possui pólo na origem e zero duplo em $s = -1/L$.

3.2.2 SEGUNDO MÉTODO

Primeiro, define que a constante integral (T_i) tende ao infinito e a constante derivativa (T_d) é igual a zero. Considerando somente a ação de controle proporcional, fazendo o valor de K_p aumentar de zero até o valor crítico (K_{cr}) que faça a resposta de saída apresentar uma oscilação sustentada pela primeira vez com período P_{cr} .

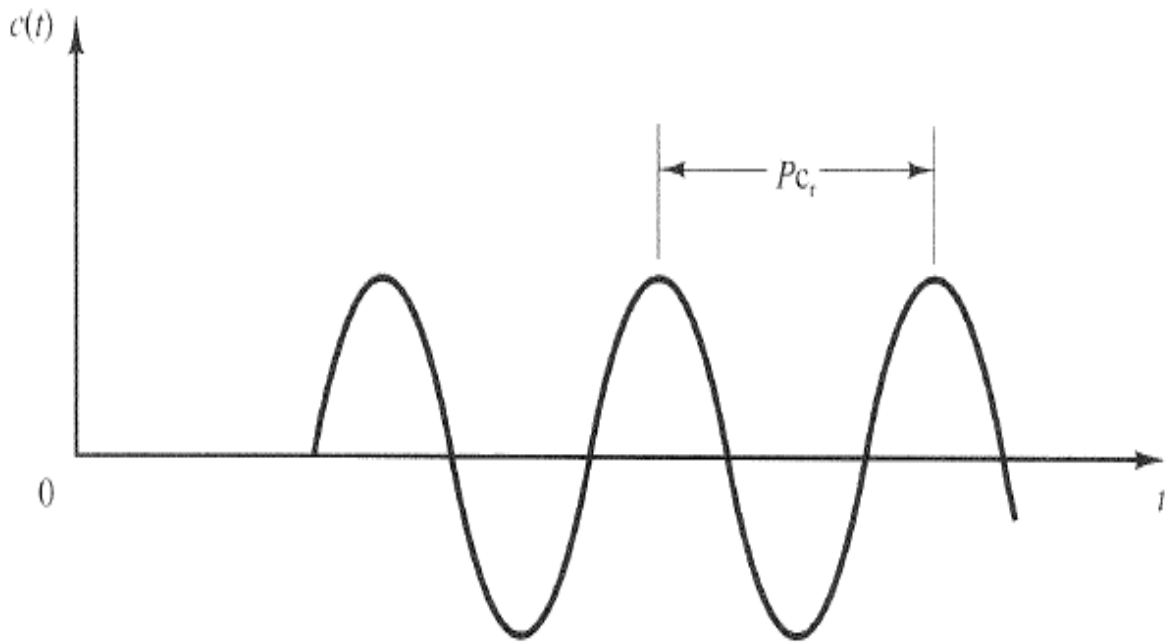


Figura 15: Oscilação sustentada com período P_{cr} (P_{cr} é medido em segundos).

Com os valores K_{cr} e P_{cr} , faz-se a regra de sintonia de Ziegler-Nichols através da tabela 3.

Tabela 3: Regra de sintonia de Ziegler-Nichols baseada no ganho crítico K_{cr} e no período crítico P_{cr} .

Tipo de controlador	K_p	T_i	T_d
P	$0,5K_{cr}$	∞	0
PI	$0,45K_{cr}$	$(1/1,2)P_{cr}$	0
PID	$0,6K_{cr}$	$0,5P_{cr}$	$0.125P_{cr}$

Com o segundo método de Ziegler-Nichols a formula controlador PID pode ser simplificado por:

$$G_c(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (3.4)$$

$$= 0.6K_{cr} \left(1 + \frac{1}{0.5P_{cr}s} + 0.125P_{cr}s \right) \quad (3.5)$$

$$0.075K_{cr}P_{cr}\frac{\left(s + \frac{4}{P_{cr}}\right)^2}{s} \quad (3.6)$$

Este controlador possui pólo na origem e zero duplo em $s = -4/P_{cr}$.

Capítulo 3

3.3 CONTROLADOR PIDNN

3.3.1 CONTROLADOR PID EM TEMPO DISCRETO

O controlador PIDNN é baseado no modelo da equação discreta da função de transferência do controlador PID. A equação genérica da função do controlador PID no domínio s é dada:

$$\frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d s \quad (3.7)$$

Onde K_d e K_i correspondem respectivamente aos ganhos derivativo e integral.

Em seguida a representação da função do controlador PID no domínio do tempo discreto é obtida por mapeamento da função transferência do domínio s para o domínio z usando aproximações discretas.

Método forward:

$$s = \frac{z - 1}{\Delta t} \quad (3.8)$$

Método backward:

$$s = \frac{z - 1}{z\Delta t} \quad (3.9)$$

A parte integral é aproximada usando o método forward e a parte derivativa com backward que resulta em:

$$C(z) = K_p + K_i \left(\frac{\Delta t}{z - 1} \right) + K_d \left(\frac{z - 1}{z\Delta t} \right) \quad (3.10)$$

O sinal de controle de tempo discreto $u(n)$ é derivado do sinal de erro $e(n)$:

$$u(n) = K_p e(n) + K_i v(n) + K_d w(n) \quad (3.11)$$

Onde $v(n)$ é o termo integral e $w(n)$ é o termo derivativo.

$$\frac{V(s)}{E(s)} = \frac{1}{s}, \quad (3.12)$$

$$\frac{W(s)}{E(s)} = s, \quad (3.13)$$

Para o domínio z :

$$\frac{V(z)}{E(z)} = \frac{\Delta t}{z - 1}, \quad (3.14)$$

$$\frac{W(z)}{E(z)} = \frac{z - 1}{z \Delta t} \quad (3.15)$$

A equação em tempo discreto para o termo integral $v(n)$ e o termo derivativo $w(n)$ fica:

$$v(n) = v(n - 1) + \Delta t \cdot e(n) \quad (3.16)$$

$$w(n) = \frac{1}{\Delta t} (e(n) - e(n - 1)) \quad (3.17)$$

3.3.2 PROJETO DO CONTROLADOR RNN

A partir da combinação das equações do controlador PID de tempo discreto obtém-se uma rede neural recorrente (figura 16).

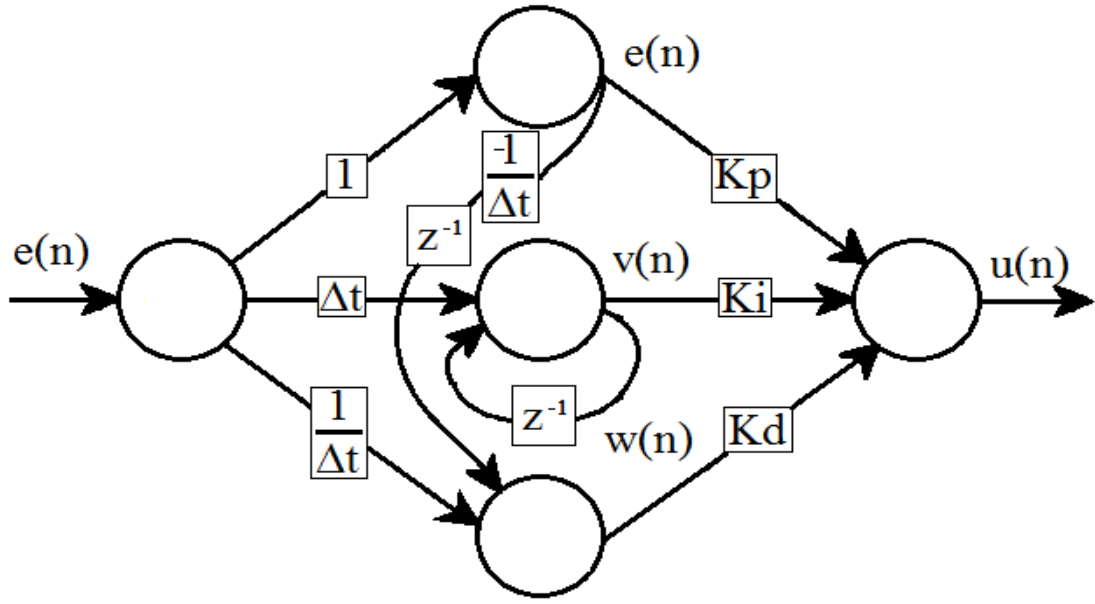


Figura 16: Estrutura do controlador PID neural.

A entrada da rede é o sinal de erro $e(n)$ e a saída é o sinal de controle $u(n)$. A função de ativação dos neurônios é linear.

3.3.3 EQUAÇÕES DE SINTONIZAÇÃO DO PIDNN

Os ganhos K_p , K_i e K_d da estrutura neural do controlador PID (Figura 16) serão os pesos que serão sintonizados e o restante dos pesos possuem valores fixos. A saída da rede pode ser computada seguindo as seguintes equações:

$$e(n) = O_1(n) \quad (3.18)$$

$$v(n) = O_2(n) = O_2(n-1) + \Delta t \cdot e(n) \quad (3.19)$$

$$w(n) = O_3(n) + \frac{1}{\Delta t} (e(n) - e(n-1)) \quad (3.20)$$

O algoritmo de treinamento do controlador PIDNN é baseado no método do gradiente descendente e utiliza o sinal de erro $e(n)$ que é a diferença entre o sinal desejado ($d(n)$) e resposta de saída da rede neural.

$$\varepsilon(n) = d(n) - u(n) \quad (3.21)$$

Para o treinamento off-line, os dados são gerados de um sistema PID simulado e usado para treinar a rede neural. A performance desejada é minimizada pela soma dos erros E_{sq} do treinamento dos dados.

$$E_{sq}(n) = \frac{1}{2} \sum_{n=1}^N \varepsilon(n)^2 \quad (3.22)$$

De acordo com método do gradiente descendente, a atualização dos pesos segue as seguintes derivações.

$$K_p(n+1) = K_p(n) - \eta \frac{\partial E_{sq}(n)}{\partial K_p} \quad (3.23)$$

$$K_i(n+1) = K_i(n) - \eta \frac{\partial E_{sq}(n)}{\partial K_i} \quad (3.24)$$

$$K_d(n+1) = K_d(n) - \eta \frac{\partial E_{sq}(n)}{\partial K_d} \quad (3.25)$$

Onde η é a taxa de aprendizagem. A regra da cadeia da derivada é usada na retro-propagação do erro para o ajuste dos pesos e na minimização da performance do treinamento. A derivada da soma dos erros quadráticos em relação aos seus respectivos parâmetros pode ser escrita como:

$$\frac{\partial E_{sq}(n)}{\partial K_p} = -\varepsilon(n) \frac{\partial u(n)}{\partial K_p} = -\varepsilon(n) O_1(n) \quad (3.26)$$

$$\frac{\partial E_{sq}(n)}{\partial K_i} = -\varepsilon(n) \frac{\partial u(n)}{\partial K_i} = -\varepsilon(n) O_2(n) \quad (3.27)$$

$$\frac{\partial E_{sq}(n)}{\partial K_d} = -\varepsilon(n) \frac{\partial u(n)}{\partial K_d} = -\varepsilon(n) O_3(n) \quad (3.28)$$

Ao mesmo tempo que os pesos são atualizados, as equações de 3.23 a 3.25 estão sendo computadas.

3.3.4 CONTROLADOR PID ADAPTATIVO

O controlador PIDNN pode ser mostrado no sistema na figura 17. O modelo de referência é escolhido de acordo com as características desejadas para o sistema malha fechada. Primeiramente o PIDNN é treinado off-line e depois é colocado em série com o sistema para sintonia online. A diferença da resposta de saída da planta em relação ao sinal de referência gera o erro e_1 que será o sinal de entrada do controlador. O erro e_2 é gerado pela diferença da resposta de saída da planta em relação à resposta do modelo de referência, este sinal está diretamente relacionado com a atualização dos pesos online.

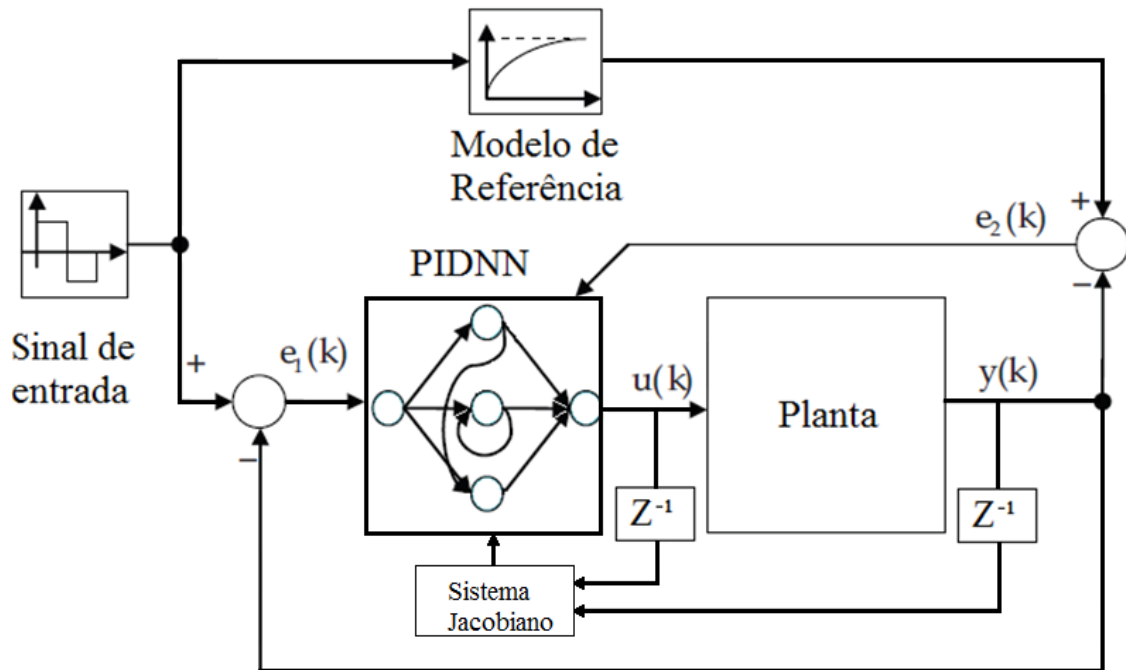


Figura 17: Malha de controle.

3.3.5 EQUAÇÕES DE ATUALIZAÇÃO DOS GANHOS ONLINE

A adaptação dos pesos feita de forma diferente que o treinamento off-line e a performance é minimizada.

$$I(n) = \frac{1}{2} e_2(n)^2 \quad (3.29)$$

Aplicando a regra da cadeia:

$$\frac{\partial I(n)}{\partial K_p} = \frac{\partial I(n)}{\partial y(n)} \times \frac{\partial y(n)}{\partial u(n)} \times \frac{\partial u(n)}{\partial K_p} = -e_2(n) \cdot J_p(n) \frac{\partial u(n)}{\partial K_p} \quad (3.30)$$

$$\frac{\partial I(n)}{\partial K_i} = \frac{\partial I(n)}{\partial y(n)} \times \frac{\partial y(n)}{\partial u(n)} \times \frac{\partial u(n)}{\partial K_i} = -e_2(n) \cdot J_p(n) \frac{\partial u(n)}{\partial K_i} \quad (3.31)$$

$$\frac{\partial I(n)}{\partial K_d} = \frac{\partial I(n)}{\partial y(n)} \times \frac{\partial y(n)}{\partial u(n)} \times \frac{\partial u(n)}{\partial K_d} = -e_2(n) \cdot J_p(n) \frac{\partial u(n)}{\partial K_d} \quad (3.32)$$

Onde $J_p(n)$ é o sistema jacobiano. Os termos $\partial u(n)/\partial K_p$, $\partial u(n)/\partial K_i$ e $\partial u(n)/\partial K_d$ podem ser calculados de forma similar pelas equações 3.26, 3.27 e 3.28.

3.3.6 SISTEMA JACOBIANO

Para um sistema MIMO com n entradas e m saídas, o sistema jacobiano é definido pela matriz de equação 3.33:

$$J_p = \frac{\partial y}{\partial u} = \begin{bmatrix} \frac{\partial y_1}{\partial u_1} & \dots & \frac{\partial y_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_m}{\partial u_1} & \dots & \frac{\partial y_m}{\partial u_n} \end{bmatrix} \quad (3.33)$$

Onde $y = [y_1 \ y_2 \ \dots y_n]^T$ é o vetor de saída e $u = [u_1 \ u_2 \ \dots u_n]^T$ é o vetor de entrada para o sistema. O cálculo do jacobiano da planta pode ser aproximado pela aproximação da derivada:

$$J_p(n) = \frac{\partial y(n)}{\partial u(n)} \approx \frac{y(n) - y(n-1)}{u(n) - u(n-1)} \quad (3.34)$$

Esta aproximação pode ser considerada precisa para um tempo de amostragem suficientemente pequeno.

3.3.7 MODELO DE REFERÊNCIA

O modelo de referência é gerado online com o algoritmo de controle no sistema em simulação. O modelo desejado em malha fechada é de terceira ordem.

$$G_{rm}(s) = \frac{\omega_n^2 p}{(s^2 + 2\zeta\omega_n + \omega_n^2) * (s + p)} \quad (3.35)$$

Onde ω_n é a frequência natural, ζ coeficiente de amortecimento e p é o parâmetro que determina o valor do terceiro pólo.

O modelo de referência desejado é considerado um sistema criticamente amortecido ($\zeta = 1$), possuindo dois pólos com o mesmo valor. O valor do parâmetro p é escolhido de tal forma que seu efeito determine o tempo de acomodação t_s (OGATA, 1982).

$$t_s = \frac{p}{\zeta\omega_n} \quad (3.36)$$

Para $t_s = 100$ s e $p = 8$, ω_n é igual a:

$$\omega_n = 0.08 \text{ rads/s}$$

Logo, a o modelo de referência fica:

$$G_{rm}(s) = \frac{0.0512}{s^3 + 8.16s^2 + 1.286s + 0.0512} \quad (3.37)$$

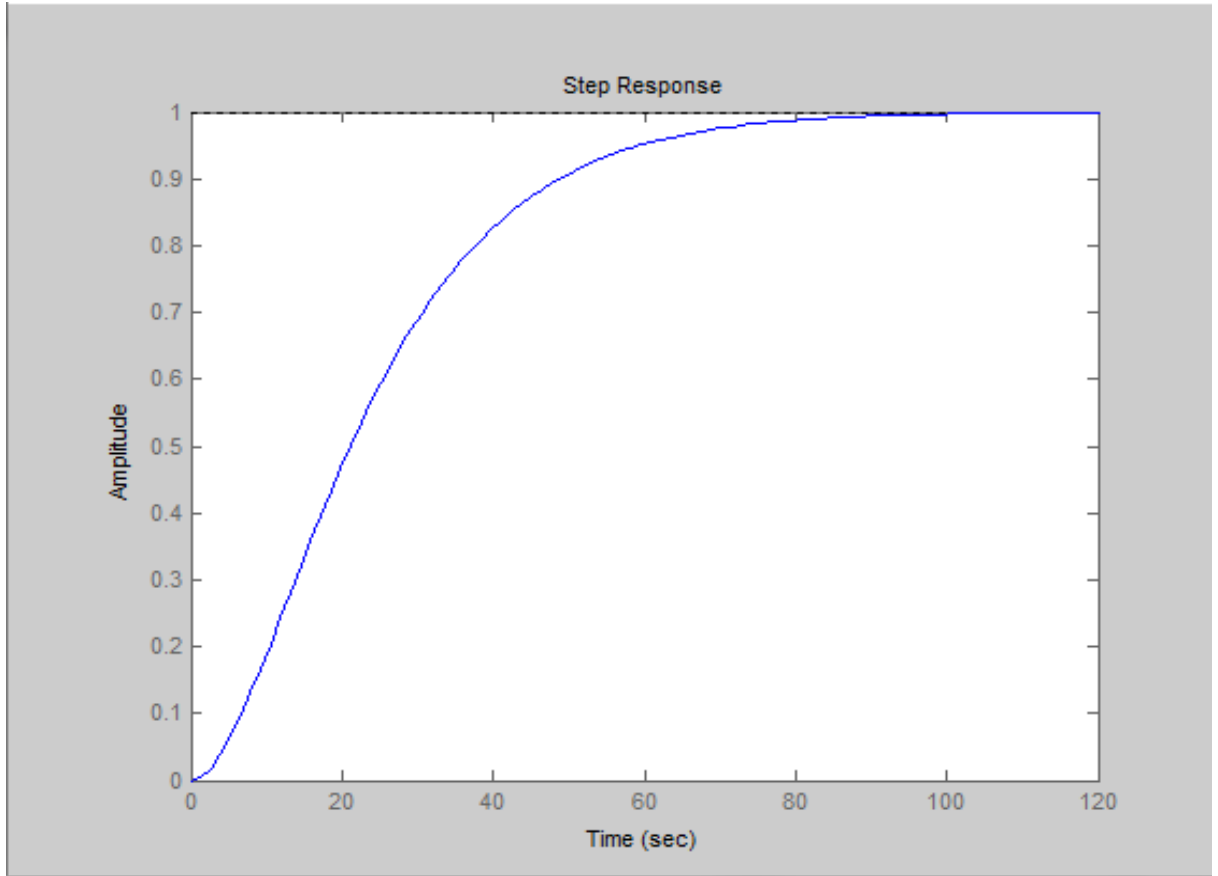


Figura 18: Resposta do modelo de referência quando aplicado o degrau unitário.

O tempo de amostragem Δt foi determinado a partir do tempo de subida que corresponde ao intervalo de tempo em que a curva do modelo de referência (figura 18) assume valores entre 10 % e 90 %. O tempo de subida da curva do modelo de referência é igual a 40 segundos e considerado que ocorram 40 amostras durante o tempo de subida, logo:

$$\Delta t = 1s$$

A representação discreta do modelo de referência por segurador de ordem zero, para Δt igual a 1 segundo é:

$$G_{rm}(z) = \frac{y_{rm}(z)}{r_m(z)} = \frac{0.002383z^2 + 0.003441z + 0.0000857}{z^3 - 1.847z^2 + 0.8528z - 0.0002859} \quad (3.38)$$

E sua forma recursiva:

$$\begin{aligned}y_{rm}(n) &= 1.847y_{rm}(n-1) - 0.8528y_{rm}(n-2) + 0.0002859y_{rm}(n-3) \\&+ 0.002383r_m(n-1) + 0.003441r_m(n-2) \\&+ 0.0000857r_m(n-3)\end{aligned}\tag{3.39}$$

CAPÍTULO 4

ANÁLISE E RESULTADOS OBTIDOS

4.1 INTRODUÇÃO

O processo a ser controlado é um sistema de nível de líquido com interação que constitui de três tanques de água acoplados, representando um sistema não linear de terceira ordem.

O primeiro tanque possui dimensão de $0.105 \times 0.12 \times 0.094 \text{m}^3$, o segundo possui dimensão de $0.105 \times 0.0941 \times 0.088 \text{m}^3$ e o terceiro tanque possui dimensão de $0.105 \times 0.095 \times 0.061 \text{m}^3$.

O primeiro tanque possui entrada para água que é fornecida por meio de uma moto-bomba que bombeia água de um reservatório para este tanque. Para medição da altura da coluna de líquido de cada tanque são utilizados potenciômetros multivoltas como sensores de nível que garante maior precisão na medida do que potenciômetros comuns. As válvulas de controle k_{12} e k_{23} fornecem água para os tanques 2 e 3 e possuem 2 cm de diâmetro e a válvula de saída k_{03} do terceiro tanque possui 1 cm de diâmetro.

Toda estrutura física do sistema de controle foi montado no LAR (Laboratório de Automação e Robótica). A modelagem do sistema e o projeto do controlador se fez pela utilização do software MATLAB® e a implementação prática se fez pelo uso de um microcontrolador ATMEGA1280.

A entrada analógica do microcontrolador tem uma resolução 10 bits e a saída analógica possui uma resolução de 8 bits. Os programas desenvolvidos no microcontrolador foram usando linguagem C.

O circuito de potência, responsável pelo ganho de corrente das portas analógicas do microcontrolador, foi alimentado por uma fonte que fornece uma tensão 8 Volts e uma corrente de 1,5 Ampére. Logo, o transistor TIP122 está operando na região do triodo (linear) para o controle da motobomba que está sendo limitada por uma tensão de no máximo 5 Volts.

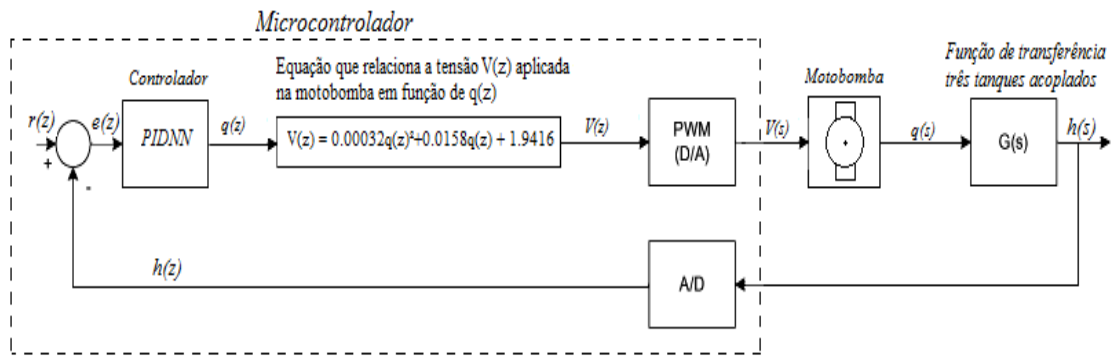


Figura 19: Diagrama de blocos do sistema.

Para a simulação e o experimento prático, o sinal de referência que corresponde à altura desejável do nível do terceiro tanque teve seu valor variando entre 3 cm a 4 cm a cada 600 segundos.

Para critério de comparação do desempenho de cada controlador utilizado neste trabalho, foram utilizados os valores característicos da curva de resposta do sistema de malha fechada. Esses valores são o tempo de subida da curva t_r (tempo em que a resposta passe de 10% a 90% do valor final) e o máximo sobressinal M_p (valor máximo de pico da curva de resposta, medido a partir da unidade em porcentagem).

4.2 SIMULAÇÃO

4.2.1. REGRAS DE SINTONIA DE ZIEGLER-NICHOLS

4.2.1.1 RESULTADO DO PRIMEIRO MÉTODO

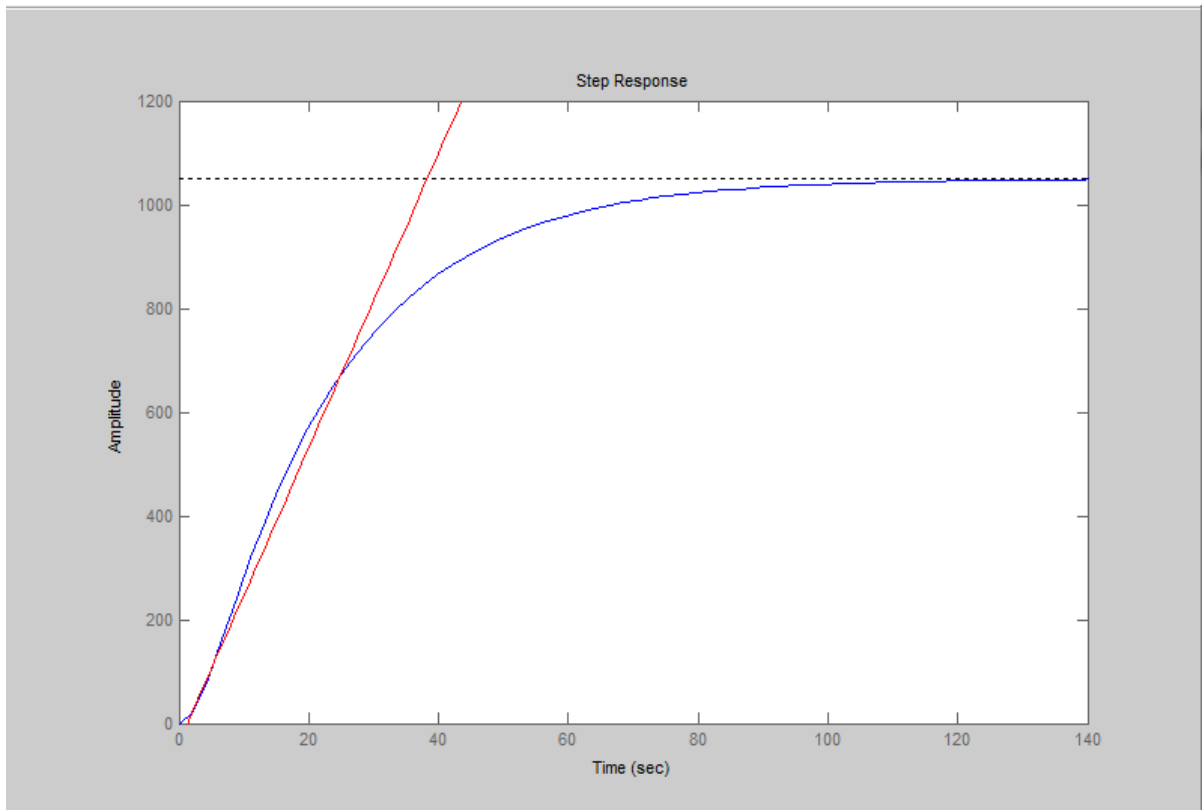


Figura 20: Curva de resposta em S da planta quando aplicada um sinal degrau unitário na entrada e reta tangente ao ponto de inflexão.

Ponto de inflexão: $X = 5$.

Equação da reta tangente ao ponto de inflexão:

$$Y = 28.4092X - 37.046 \quad (4.1)$$

Através da reta Y que tangencia o ponto de inflexão X da curva de resposta ao degrau unitário (figura 20), obtém os valores de T e L .

$$T = 36.96 \text{ e } L = 1.304$$

De acordo com a tabela 2, para o controlador PID os valores de K_p , T_i e T_d valem:

$$K_p = 34.0123, T_i = 2.608 \text{ e } T_d = 0.6520$$

O primeiro método de Ziegler-Nichols para controlador PID apresentou um resultado instável na resposta de saída da planta e também para os controladores dos tipos P e PI.

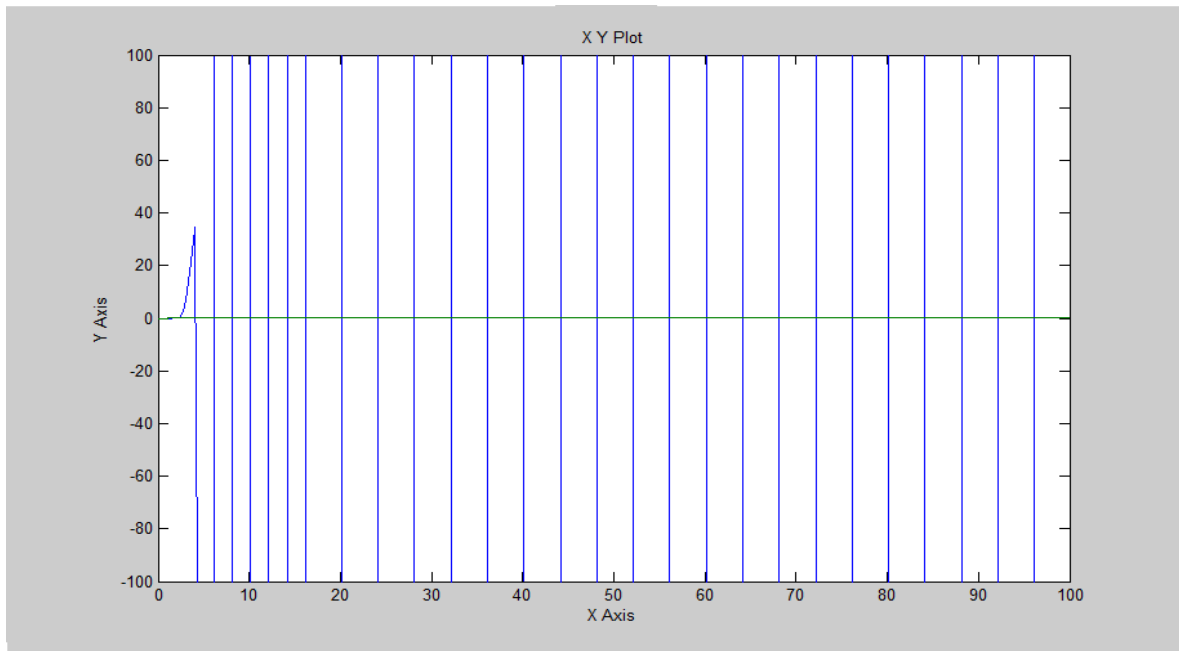


Figura 21: Resposta da saída da planta com o controlador PID sintonizado pelo primeiro método Ziegler-Nichols.

4.2.1.2 RESULTADO DO SEGUNDO MÉTODO

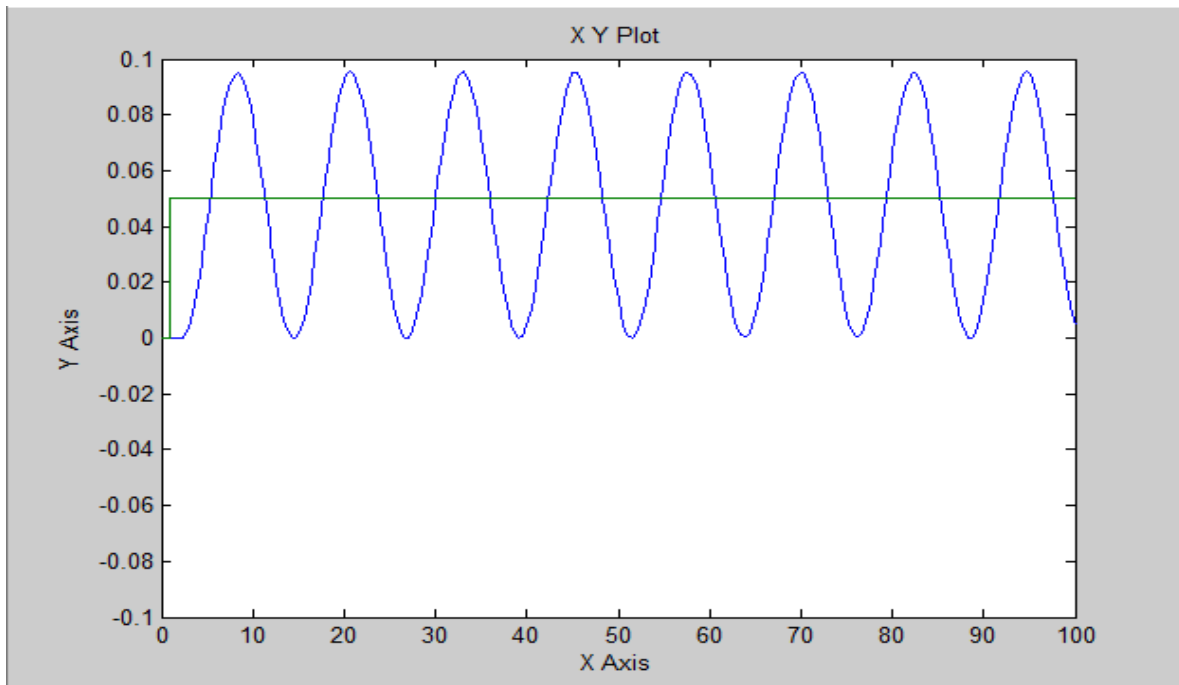


Figura 22: Curva de resposta obtida com o ganho crítico K_{cr} para determinar o período crítico P_{cr} .

Os valores do ganho crítico K_{cr} e período crítico P_{cr} obtidos são:

$$K_{cr} = 0.0191 \text{ e } P_{cr} = 12.4$$

De acordo com a tabela 3, para o controlador PID, os valores de K_p , T_i e T_d valem:

$$K_p = 0.0115, T_i = 6.40 \text{ e } T_d = 1.55$$

O controlador PID sintonizado pelo segundo método de Ziegler-Nichols também fez o sistema ficar instável. Logo foi proposto um controlador do tipo PI e de acordo com a tabela 3 os valores de K_p e T_i valem:

$$K_p = 0.008595 \text{ e } T_i = 10.33$$

Para o controlador PI no domínio tempo discreto usando a aproximação discreta forward e com um período de amostragem igual a 1 segundo, temos:

$$C(z) = \frac{u(z)}{e(z)} = \frac{0.008595z - 0.0078}{z - 1} \quad (4.2)$$

Em sua forma recursiva:

$$u(n) = u(n - 1) + 0.008595e(n) - 0.0078e(n - 1) \quad (4.3)$$

A reposta do sistema malha fechada com o controlador PI fica:

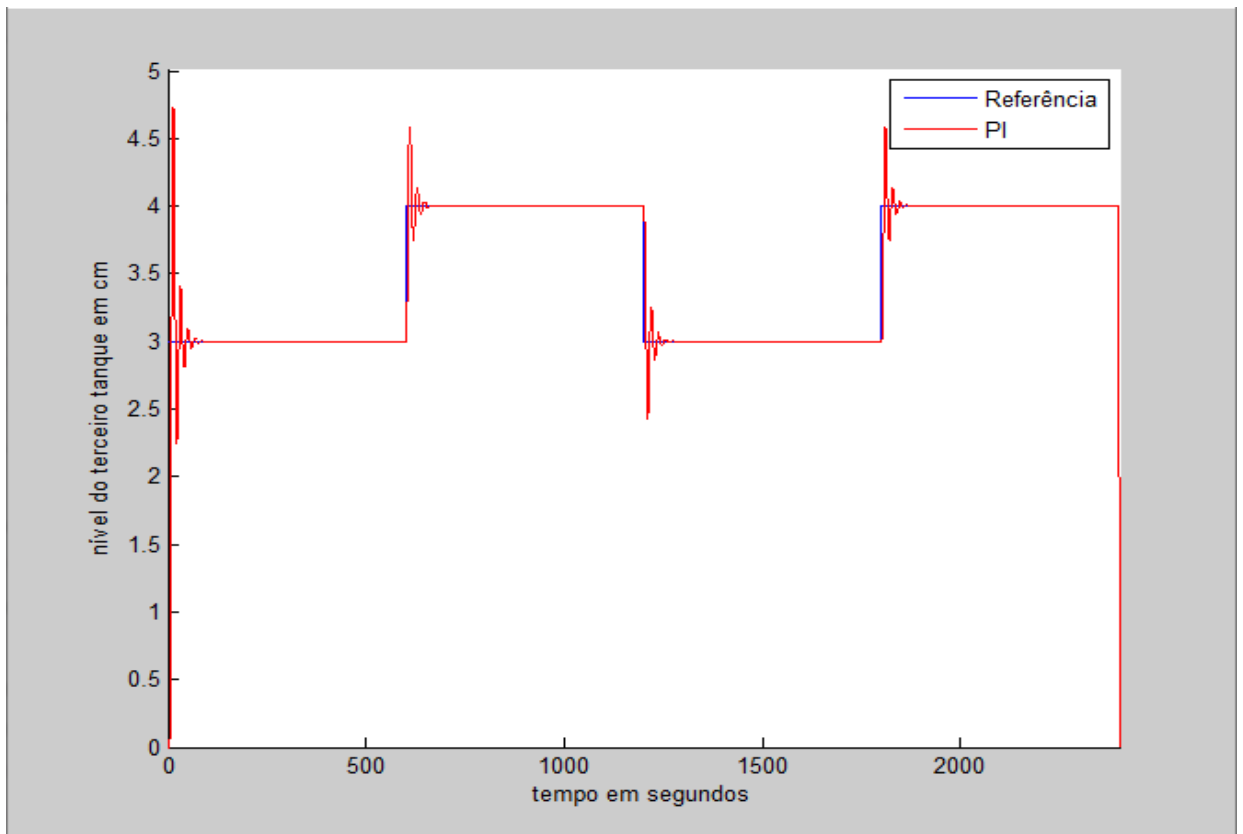


Figura 23: Resposta do controlador PI sintonizado pelo segundo método de Ziegler-Nichols na simulação.

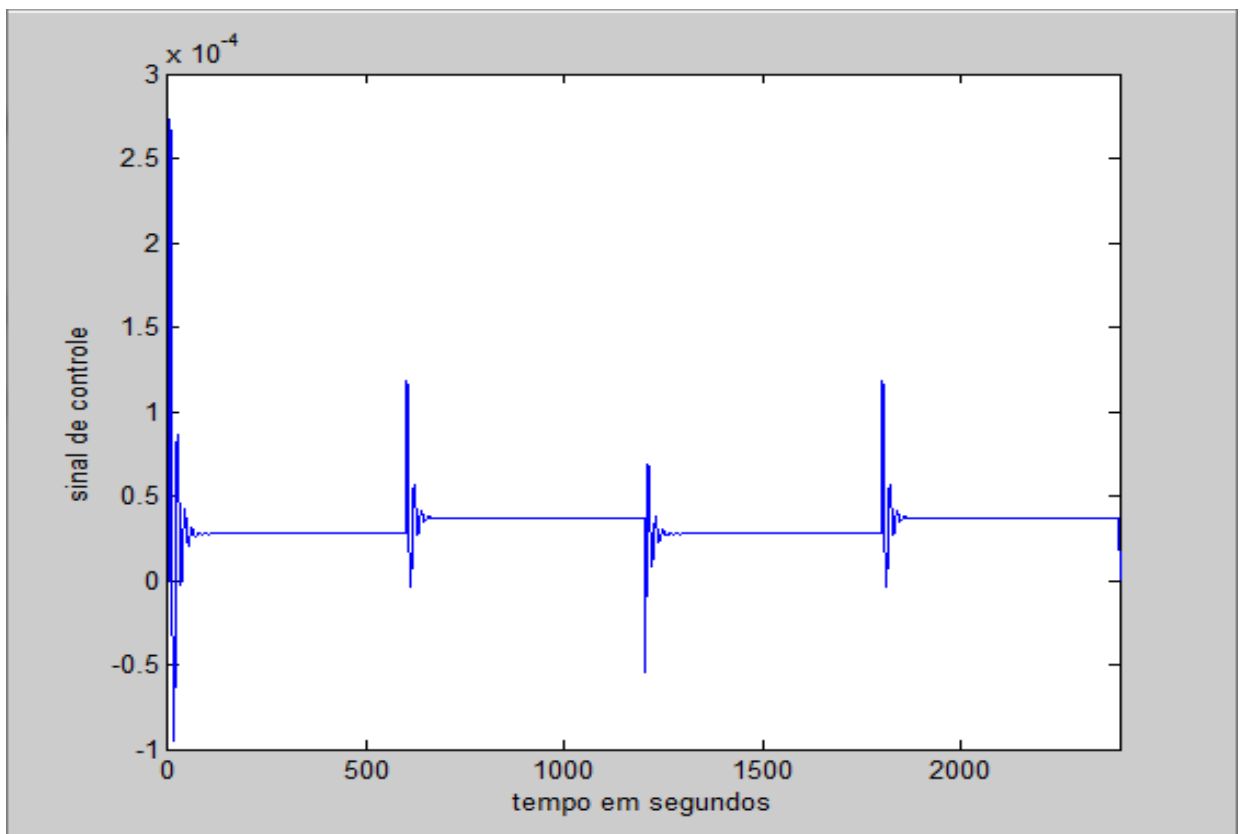


Figura 24: Sinal de Controle do controlador PI na simulação.

A partir da curva de resposta de malha fechada com o controlador PI, foram obtidos os seguintes valores:

$$t_r = 3.40 \text{ segundos.}$$

$$M_p = 57,77 \text{ \%}.$$

4.2.2 RESULTADO DO CONTROLADOR PIDNN

Tabela 4: Parâmetros iniciais para simulação do PIDNN.

Parâmetros iniciais de Simulação do PIDNN	
Tempo de Amostragem	1 segundo
Taxa de aprendizado da rede neural	1×10^{-5}
Coefficiente de Amortecimento (ζ)	1
Ganho K_p Inicial	4.169×10^{-5}
Ganho K_i Inicial	4.3×10^{-5}
Ganho K_d Inicial	1.795×10^{-5}

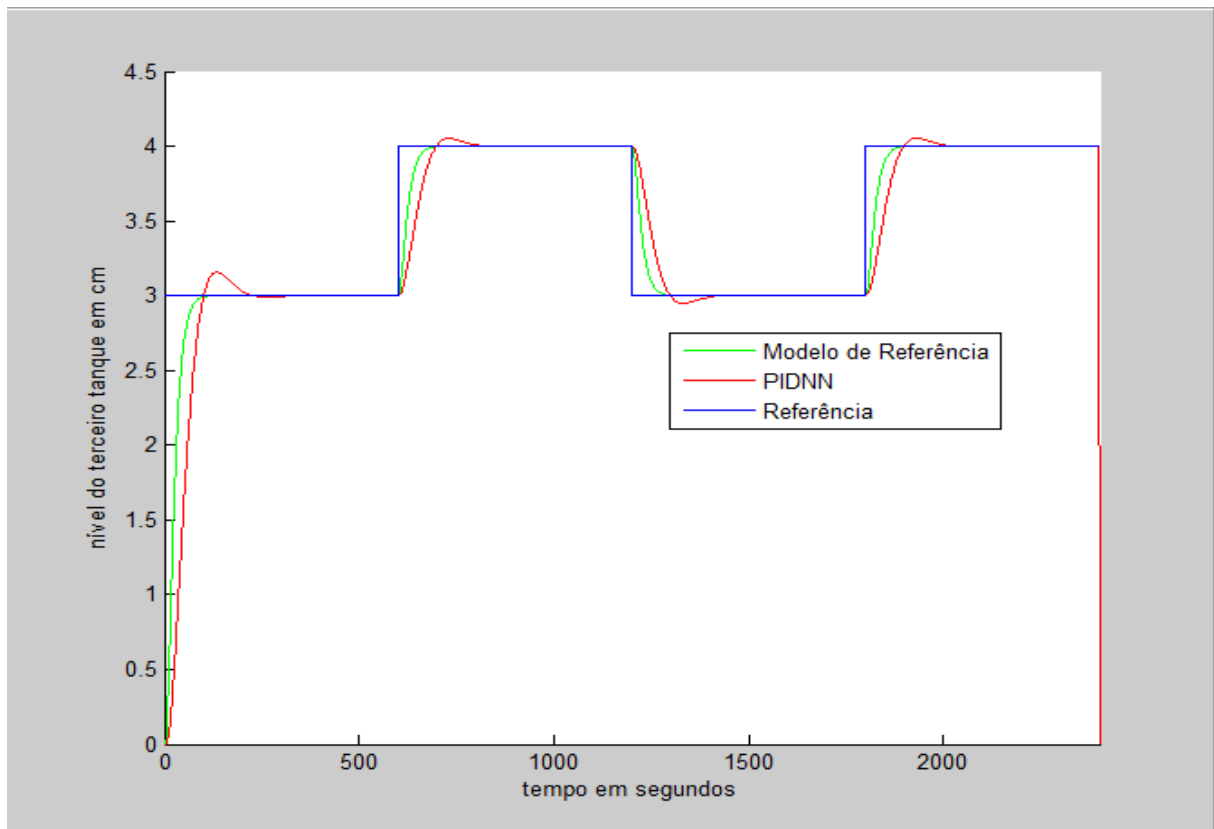


Figura 25: Resposta simulada do controlador PIDNN na simulação.

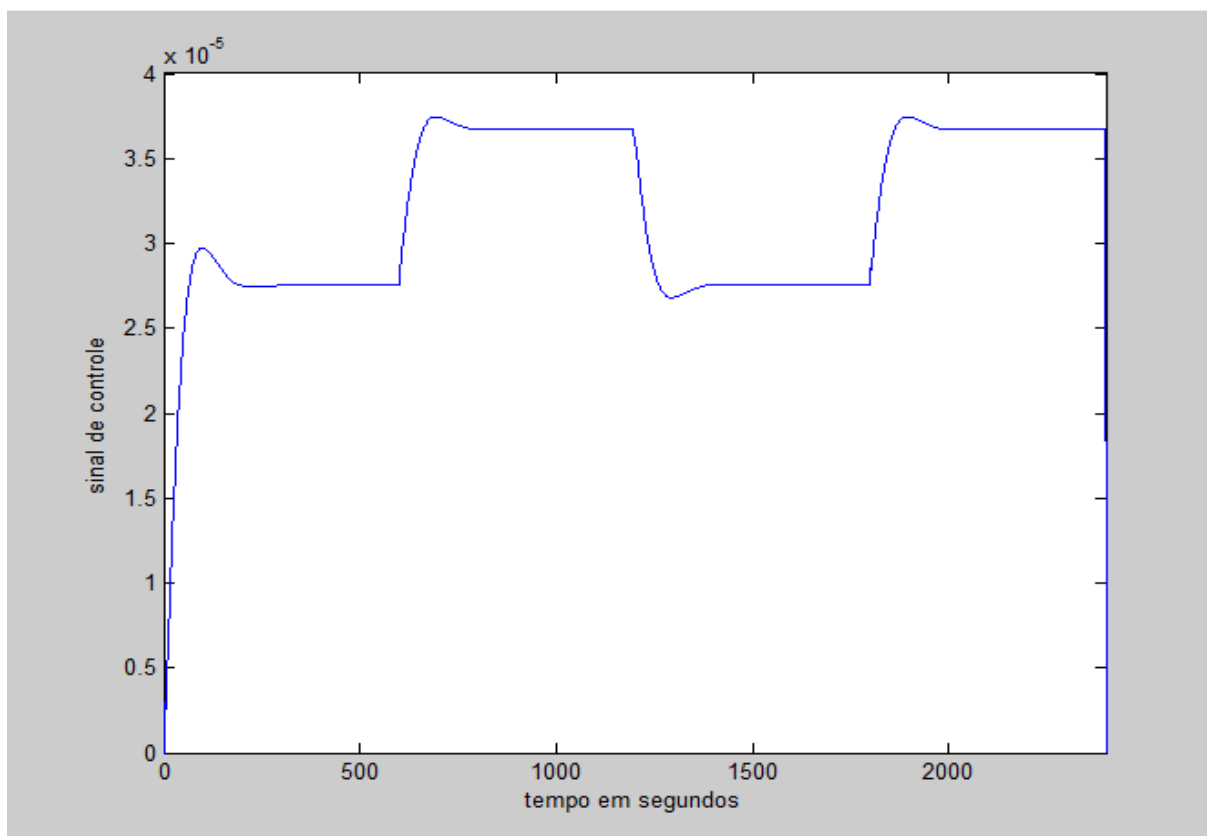


Figura 26: Sinal de controle PIDNN na simulação.

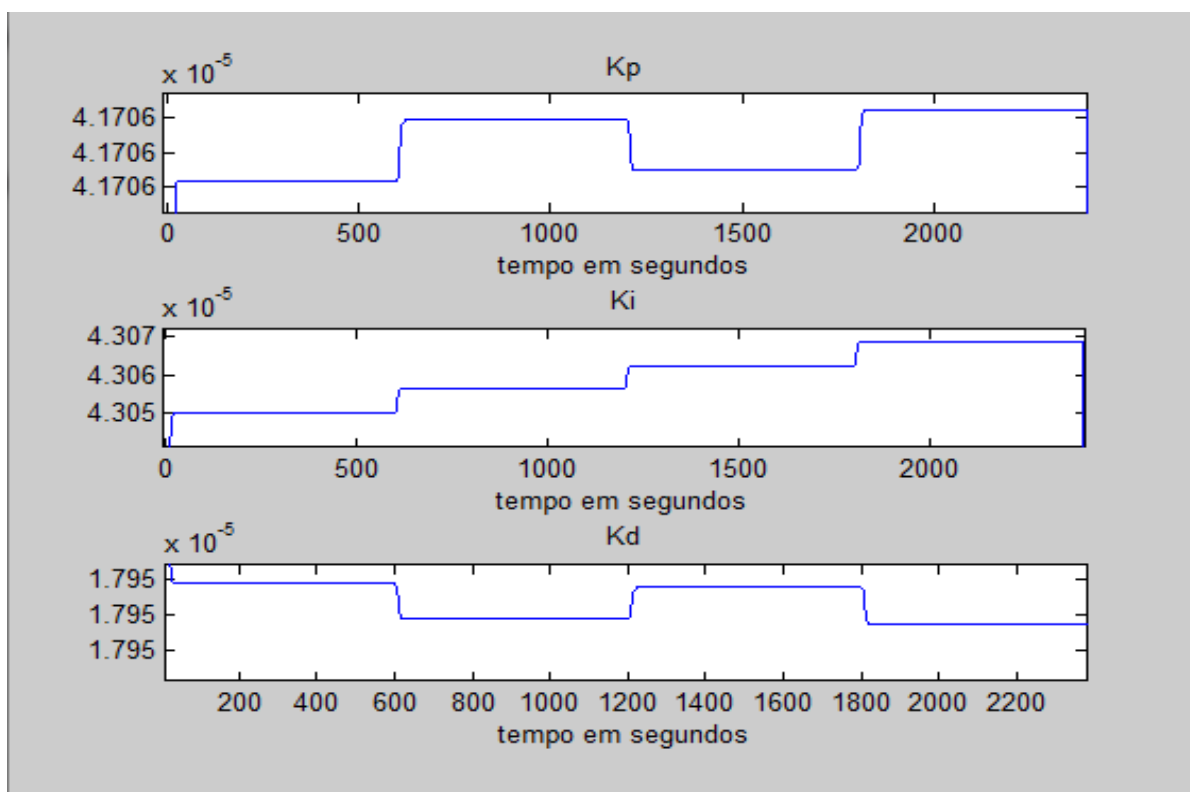


Figura 27: Variação dos ganhos K_p , K_i e K_d na simulação.

A partir da curva de resposta de malha fechada com o controlador PIDNN, foram obtidos os seguintes valores:

$$t_r = 63 \text{ segundos.}$$

$$M_p = 5,23 \text{ \%}.$$

4.3 RESULTADO EXPERIMENTAL

Para a obtenção dos dados e a aplicação das estratégias de controle o sistema físico real é apresentado na figura a seguir.



Figura 28: Controle do processo de nível líquido.

4.3.1 RESULTADO DO CONTROLADOR PI

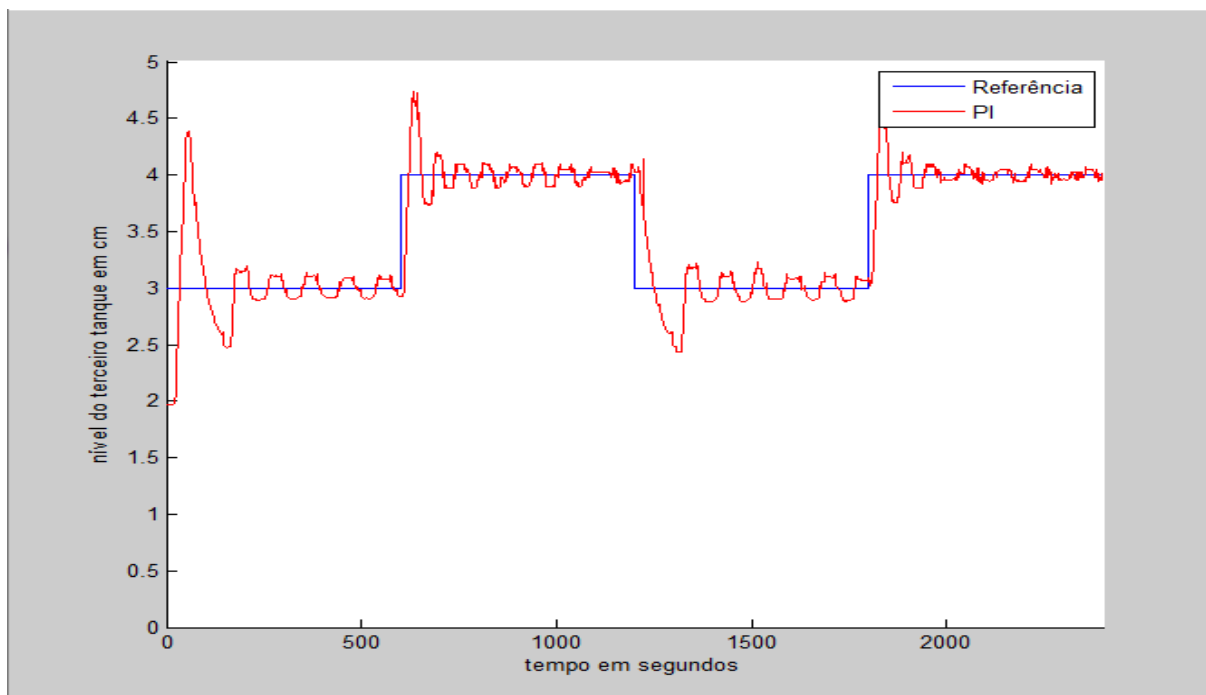


Figura 29: Resultado do Controlador PI no experimento prático.

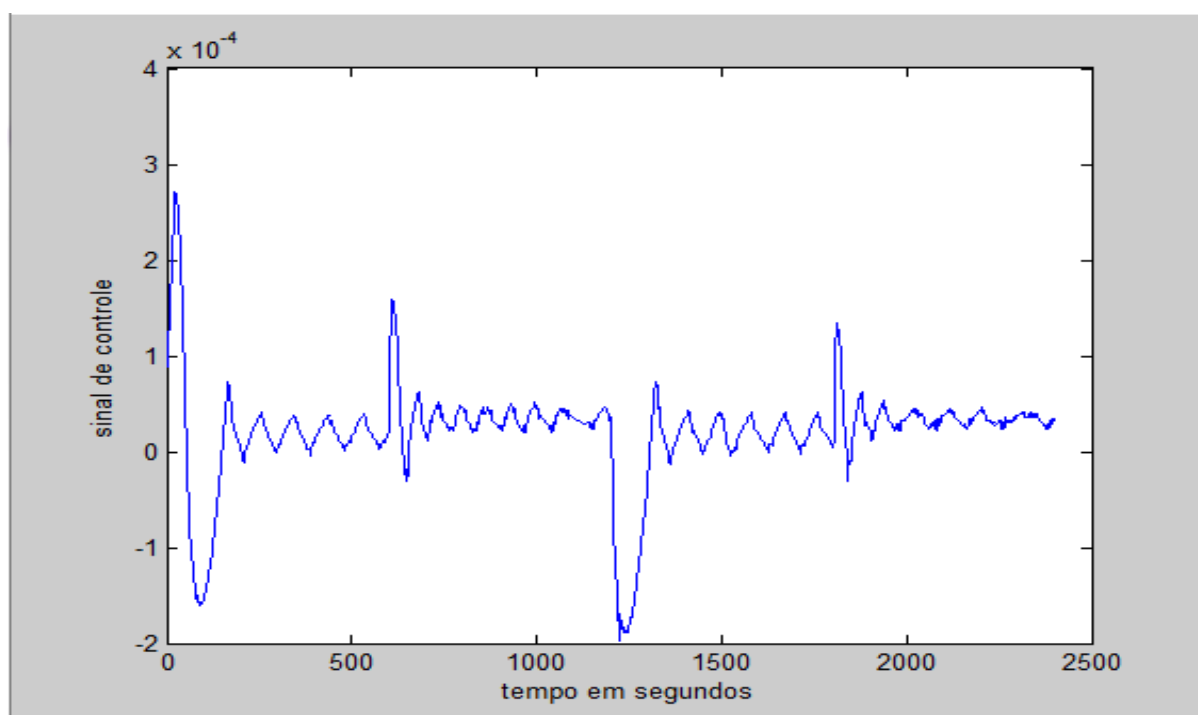


Figura 30: Sinal de controle do controlador PI no experimento prático.

A partir do resultado prático do controlador PI, foram obtidos os seguintes valores:

$$t_r = 25 \text{ segundos.}$$

$$M_p = 43 \text{ \%}.$$

4.3.2 RESULTADO DO CONTROLADOR PIDNN

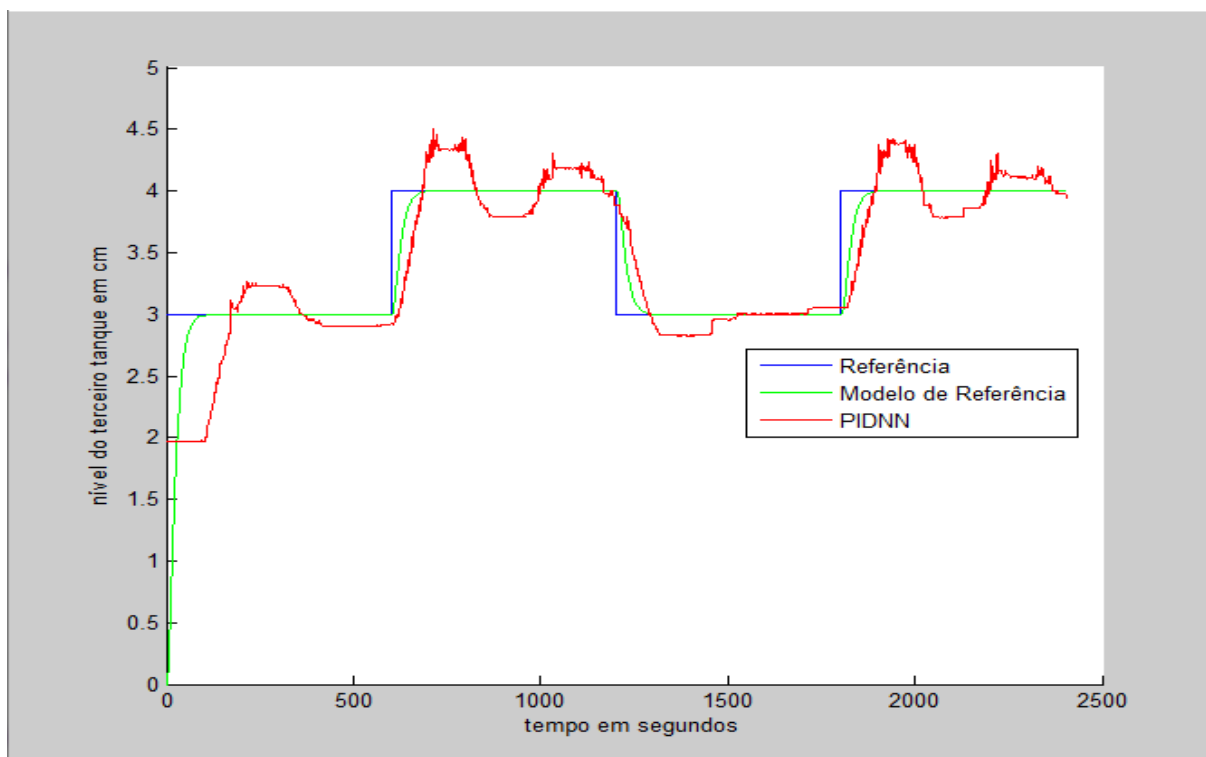


Figura 31: Resultado do Controlador PIDNN no experimento prático.

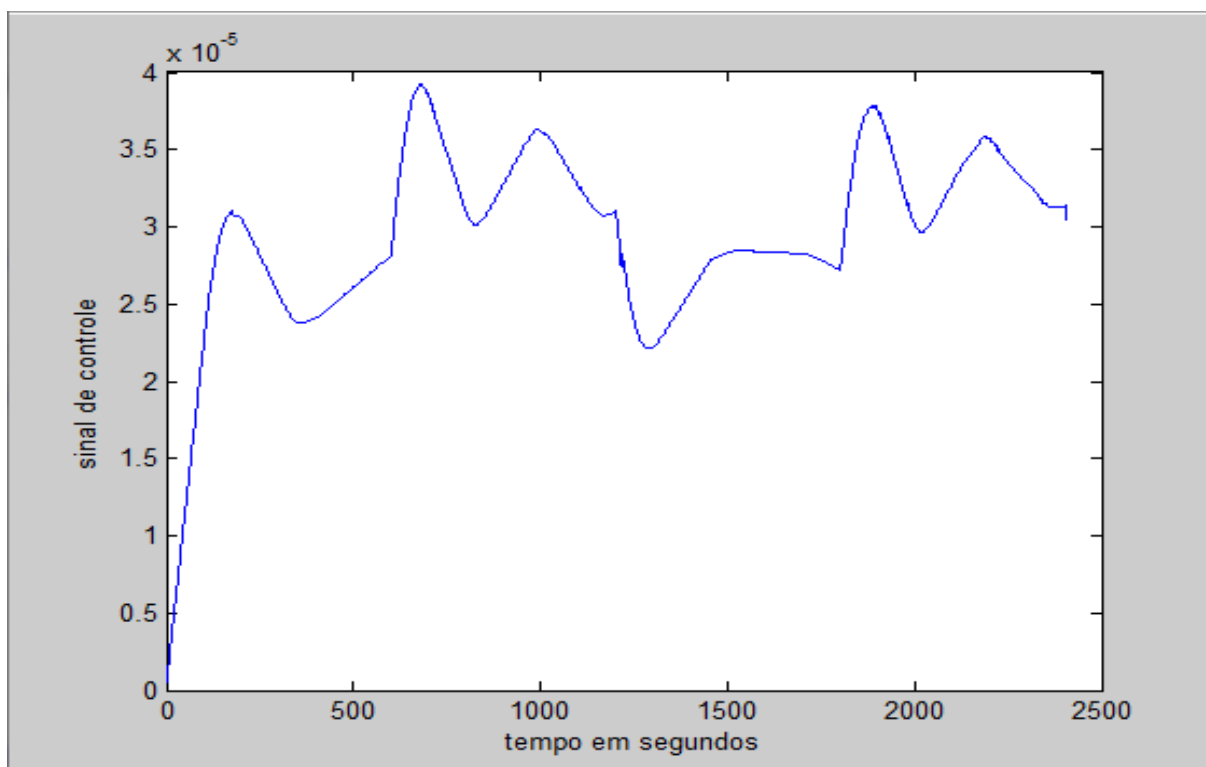


Figura 32: Sinal de controle do controlador PIDNN no experimento prático.

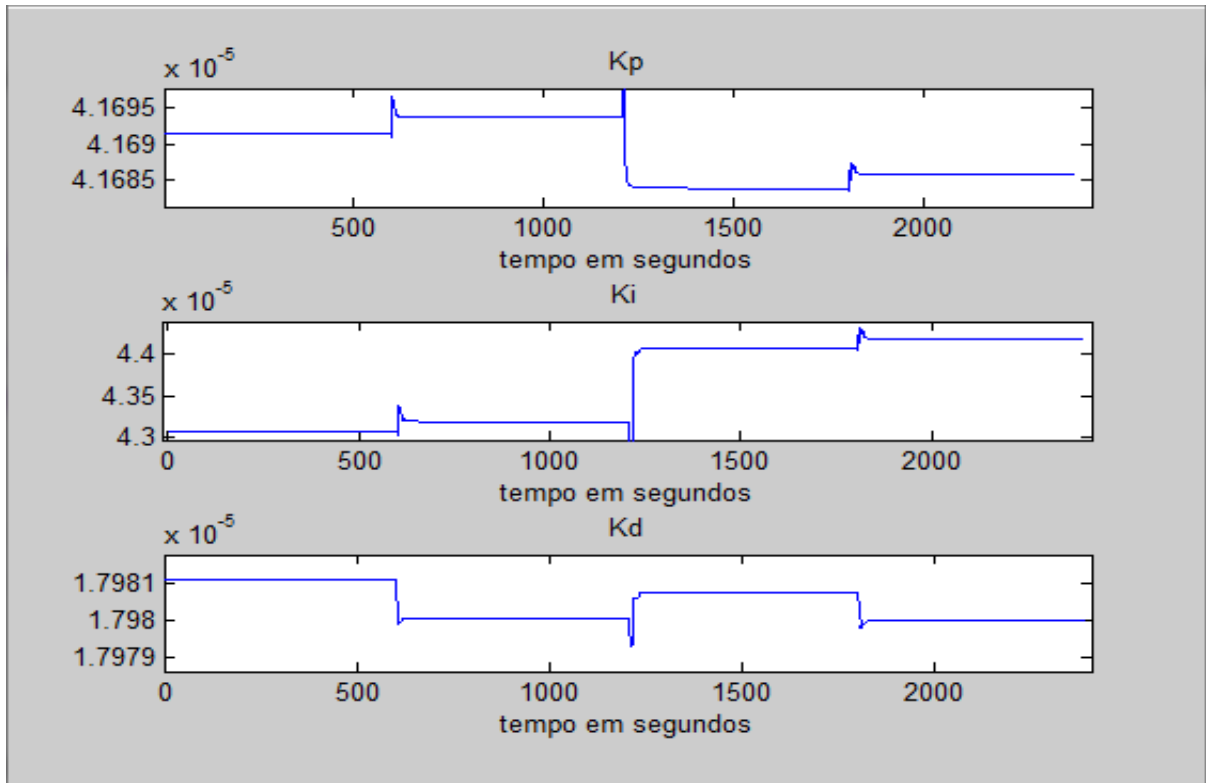


Figura 33: Variação dos ganhos K_p , K_i e K_d no experimento prático.

A partir do resultado prático do controlador PIDNN, foram obtidos os seguintes valores:

$$t_r = 48 \text{ segundos.}$$

$$M_p = 9.25 \text{ \%}.$$

CONCLUSÃO

Observando os resultados da simulação e experimentais, o controlador PIDNN apresentou desempenho superior que o controlador PI sintonizado pelo segundo método de Ziegler-Nichols. Mesmo tendo uma resposta mais lenta, o controlador PIDNN apresentou máximo sobre-sinal muito menor.

O uso da primeira regra de Ziegler-Nichols se mostrou inadequado em relação à complexidade do sistema, para a sintonia dos controladores do tipo PID, PI e P.

Através dos resultados experimentais pode se observar o que sensor do terceiro tanque possui imprecisão na medida devido ao atrito presente no potenciômetro que afetou o desempenho dos controladores.

As limitações práticas como máxima tensão aplicada ao motor e offset causado pela mínima leitura do sensor de nível influíram muito nos resultados práticos. Na figura 31 o sinal de resposta de saída sofre atraso devido ao offset causado pela mínima leitura do sensor de nível.

Este trabalho de conclusão de curso além de ajudar no desenvolvimento do Laboratório de Automação e Robótica certamente servirá como referência para novos estudos e futuros trabalhos.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] HAYKIN, S. *Redes Neurais - Princípios e Prática*, Bookman, 2001.
- [2] BRAGA, A. P., CARVALHO, A. C. P. L. F. and LUDEMIR, T. B. *Redes Neurais Artificiais*. Rio de Janeiro, 2000, Vol. 1, LTC.
- [3] FRACASSO, P. T. e SILVA, B. C. *Calibração de Medidores de Vazão Utilizando um Controlador PID Neural*. XVIII Congresso Brasileiro de Automática (Bonito-MS).-USP, 2010.
- [4] FERREIRA, A. P. L. *Uma Proposta de Controle Neural Adaptativo para Posicionamento de um Braço Mecânico*. Dissertação de Mestrado. UFRGS, 1996.
- [5] RIBEIRO, P. T. *Desenvolvimento, Implementação e Avaliação de Desempenho de um Controlador Adaptativo do Tipo Self-Tuning Regulator Aplicado a um processo FCC*. Tese de Doutorado. UNICAMP. 2010.
- [6] NARENDRA, K. S. and PARTHASARATHY, K. *Neural Networks and Dynamical Systems*. Part I: A gradient approach to Hopfield networks. Technical Reports 8820, Center for Systems Science, Department of Electrical Engineering, Yale University. 1988.
- [7] OLIVEIRA, P. J. C. *Avaliação de Controle Neural a um Processo de Quatro Tanques Acoplados*. Tese de doutorado. UnB. 2009.
- [8] CARRARA, V. *Redes Neurais Aplicadas ao Controle de Altitude de Satélites com Geometria Variável*. Tese de Doutorado. INPE. 1997.
- [9] EYNG, E. *Controle Feedforward-Feedback Aplicado às Colunas de Absorção do Processo de Produção de Etanol por Fermentação*. Tese de doutorado. UNICAMP. 2008.
- [10] SOUTO, R.F, GINANI, L. S. e BAUCHSPIESS, A. *Estudo Comparativo de Técnicas de Controle em um Processo de Nível*. Artigo Científico. Simpósio de Excelência em Gestão e Tecnologia (SEGET). 2007.
- [11] OLIVEIRA, J. C, RESENDE, P. e MAIA C. A. *Controle Neural Gain Scheduling de Tanques Interativos*. Artigo Científico. UFMG. 3o SBAi.Vitória, ES, 1997.

- [12] FILHO O. G, NETO A. D. D, REGO J. B.A e DAMASCENO N. C. *Controlador Híbrido Neural no Controle de Nível de uma Planta Industrial*. Congresso Brasileiro de Automática, 2008, Juiz de Fora. XVII Congresso Brasileiro de Automática.
- [13] ROCHA, Fabiano L. Identificação de sistemas não lineares multivariáveis usando redes neurais perceptron multicamadas e função de base radial. Dissertação de Mestrado. Curitiba: PUC/PR, 2006.
- [14] AGUIRRE, L. A. *Introdução à Identificação de Sistemas – Técnicas Lineares e Não-lineares Aplicadas a Sistemas Reais*. 3 ed. Belo Horizonte, editora UFMG, 2007.
- [15] HAN, W, HAN J and LEE C. *Development of a Self-tuning PID Controller based on Neural Network for Nonlinear Systems*. Proceedings of the 7th Mediterranean Conference on Control and Automation (MED99) Haifa, Israel – 1999.
- [16] OGATA, K, *Engenharia de Controle Moderno*. Person Education. 5º Edição. 2011.
- [17] OGATA, K, *Engenharia de Controle Moderno*. Prentice-Hall. 2º Edição. 1982.
- [18] MILLER, W. T, SUTTON, R. S and WERBOS, P. J. *Neural Networks for Control*. Massachusetts Institute of Technology. 1990.
- [19] SAMAD, T and SU, H. T. *Neural Networks as Process Controllers-Optimization aspects*. In Proceedings of the American Control Conference, volume 3, pages 2486-2490. IEEE Press, Piscataway, New Jersey, 1994.
- [20] OMIDVAR, O and Elliot, D. L. *Neural Systems for Control*. Academic Press. 1997.
- [21] Sarangapani, J. *Neural Network Control of Nonlinear Discrete-Time Systems*. TLFBOOK, 2006.
- [22] ABNT. Normas para Apresentação de Documentos Científicos, Vol.2 – Teses, dissertações, monografias e trabalhos acadêmicos, UFPR, 2002.