



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
PRÓ-REITORIA DE GRADUAÇÃO
DEPARTAMENTO DE ENGENHARIA E TECNOLOGIA
BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

LUMA SLLARY FERNANDES OLIVEIRA

**INTELIGÊNCIA COMPUTACIONAL APLICADA NO CONTROLE DE
POSIÇÃO DE UM MOTOR DE CORRENTE CONTÍNUA**

PAU DOS FERROS

2021

LUMA SLLARY FERNANDES OLIVEIRA

**INTELIGÊNCIA COMPUTACIONAL APLICADA NO CONTROLE DE
POSIÇÃO DE UM MOTOR DE CORRENTE CONTÍNUA**

Monografia apresentada a Universidade
Federal Rural do Semi-Árido como
requisito para obtenção do título de
Bacharel em Engenharia de Computação.

Orientador: Ádller de Oliveira
Guimarães, Prof. Dr.

PAU DOS FERROS

2021

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

048i Oliveira, Luma Sllary Fernandes.
 Inteligência computacional aplicada no
 controle de posição de um motor de corrente
 contínua / Luma Sllary Fernandes Oliveira. - 2021.
 43 f. : il.

 Orientador: Ádller de Oliveira Guimarães.
 Monografia (graduação) - Universidade Federal
 Rural do Semi-árido, Curso de Engenharia de
 Computação, 2021.

 1. inteligência computacional. 2. algoritmo
 genético. 3. redes neurais. 4. motor de corrente
 contínua. 5. controlador PID. I. Guimarães,
 Ádller de Oliveira, orient. II. Título.

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

LUMA SLLARY FERNANDES OLIVEIRA

**INTELIGÊNCIA COMPUTACIONAL APLICADA NO CONTROLE DE
POSIÇÃO DE UM MOTOR DE CORRENTE CONTÍNUA**

Trabalho de Conclusão de Curso
apresentado, a Universidade Federal
Rural do Semi-Árido - UFERSA,
Campus Pau dos Ferros, como requisito
para a obtenção do título de Bacharel em
Engenharia de Computação.

DATA DE APROVAÇÃO 01/06/2021.

BANCA EXAMINADORA

ADLLER DE OLIVEIRA

GUIMARAES:07614216458

Assinado de forma digital por ADLLER
DE OLIVEIRA
GUIMARAES:07614216458
Dados: 2021.06.02 22:46:39 -03'00'

Prof. Dr. Ádller de Oliveira Guimarães (UFERSA)

Presidente

CECILIO MARTINS DE

SOUSA NETO:06286737448

Assinado de forma digital por CECILIO
MARTINS DE SOUSA NETO:06286737448
Dados: 2021.06.03 14:11:47 -03'00'

Prof. Dr. Cecílio Martins de Sousa Neto (UFERSA)

1º Examinador

TYCIANNE JANYNNE DE

OLIVEIRA CABRAL:09381932441

Assinado de forma digital por TYCIANNE
JANYNNE DE OLIVEIRA
CABRAL:09381932441
Dados: 2021.06.02 22:47:15 -03'00'

Prof.^a Dr.^a Tycianne Janyenne de Oliveira Cabral (UFERSA)

2º Examinador

AGRADECIMENTOS

Agradeço aos meus pais, José Lurandir e Maria Fernandes, e ao meu irmão Lennon Smith, por apoiarem minha caminhada na educação e por me incentivarem a buscar o melhor que a vida tem para oferecer.

Agradeço ao meu orientador, Prof. Dr. Ádller de Oliveira Guimarães, pela paciência, disponibilidade e pelos ensinamentos.

Agradeço a banca examinadora, pelas valiosas contribuições que vieram a enriquecer o trabalho.

Agradeço aos demais professores, muitos de vocês ensinaram mais que conteúdos didáticos. Em especial aos professores Segundo Gurgel, Cecílio Martins, Pedro Thiago, Marco Diego e Ernano Arrais.

Agradeço aos meus amigos Leila Freitas, Bianca Moraes, Italo Deison, Rafael Abrantes e Bruno Fontes. Agradeço por todo o incentivo, ajuda e força que me deram nos momentos difíceis e por sempre acreditarem que eu seria capaz.

RESUMO

Os motores de corrente contínua (motor CC) são amplamente utilizados por sua versatilidade e sua facilidade de controle de velocidade e posição. O controle pode ser feito por meio do controlador PID, um dos mais utilizados na indústria, e para tal é necessária aplicação de técnicas de sintonização dos parâmetros do controlador. Dessa forma, o presente trabalho tem como objetivo aplicar técnicas clássicas e técnicas alternativas, como os Algoritmos Genéticos, para realizar o ajuste dos parâmetros de um controlador PID aplicado em um motor CC de modo a conseguir uma melhor resposta. É também proposta a aplicação da técnica de Redes Neurais Artificiais (RNA) com base nos dados obtidos por meio do Algoritmo Genético (AG) para alcançar a resposta desejada. Os resultados obtidos pela otimização por AG mostraram uma melhor performance quando comparados com os fornecidos pelos métodos clássicos, assim como em relação aos trabalhos encontrados na literatura. Por fim, os resultados obtidos pela RNA demonstraram uma boa capacidade de previsão, e pode ser usada em conjunto com a AG para a otimização do sistema de controle proposto.

Palavras-chave: inteligência computacional; algoritmo genético; redes neurais; motor de corrente contínua; controlador PID

ABSTRACT

Continuous current motors (DC motor) are widely used for their versatility and their ease of speed and position control. The control can be done through the PID controller, one of the most used in the industry, and for this it is necessary to apply tuning techniques of the parameters of the controller. Thus, the present work aims to apply classical techniques and alternative techniques, such as Genetic Algorithms, to perform the adjustment of the parameters of a PID controller applied in a DC engine in order to achieve a better response. It is also proposed the application of the Artificial Neural Networks (RNA) technique based on the data obtained through the Genetic Algorithm (GA) to achieve the desired response. The results obtained by optimization by GA showed a better performance when compared to those provided by the classical methods, as well as in relation to the studies found in the literature. Finally, the results obtained by RNA demonstrated a good predictability, and can be used in conjunction with GA for the optimization of the proposed control system.

Keywords: computational intelligence; genetic algorithm; neural network; direct current motor; PID controller.

LISTA DE FIGURAS

Figura 1 – Modelo do circuito elétrico do Motor CC.....	14
Figura 2 – Resposta do sistema em malha aberta.....	18
Figura 3 – Modelo do diagrama de blocos para o 2º método de ZN.....	19
Figura 4 – Exemplo de aplicação do crossover de 1 ponto	22
Figura 5 – Estrutura básica do neurônio artificial	24
Figura 6 – Rede de alimentação direta com camada única.....	24
Figura 7 – Rede com múltiplas camadas	25
Figura 8 – Rede recorrente	25
Figura 9 – Representação genérica do sistema de controle a ser projetado	27
Figura 10 – Resposta do sistema à entrada ao degrau sem controlador	28
Figura 11 – Diagrama de blocos do sistema sintonizado pelo método de Ziegler-Nichols	29
Figura 12 – Resposta do sistema sintonizado pelo método de Ziegler-Nichols.....	30
Figura 13 – Diagrama de blocos do sistema sintonizado pelo ajuste fino.....	31
Figura 14 – Resposta do sistema sintonizado pelo ajuste fino	32
Figura 15 – Resposta do sistema sintonizado por Algoritmo Genético	35
Figura 16 – Comparação entre ajustes de controladores	35
Figura 17- Estrutura RNA proposta	37
Figura 18 – Desempenho da rede neural durante o treinamento	37
Figura 19 – Curvas de regressão	38
Figura 20 - Comparativo entre respostas da amostra 3	40

LISTA DE TABELAS

Tabela 1 – Parâmetros do controlador PID - 1º Método de Ziegler-Nichols	18
Tabela 2 – Parâmetros do controlador PID - 2º Método de Ziegler-Nichols	19
Tabela 3 – Parâmetros do Motor CC	26
Tabela 4 – Parâmetros do Algoritmo Genético	34
Tabela 5 – Dados de validação da RNA.....	39

SUMÁRIO

1 INTRODUÇÃO	12
1.1 Objetivo Geral.....	12
1.2 Objetivos Específicos	12
1.2 Estrutura do Trabalho	13
2 REVISÃO BIBLIOGRÁFICA	14
2.1 Modelagem do Motor de Corrente Contínua.....	14
2.2 Controladores PID	16
2.2.1 Ações Proporcional-Integral-Derivativa	16
2.3 Métodos de Sintonia de Ziegler-Nichols	17
2.3.1 Primeiro Método	18
2.3.1 Segundo Método	19
2.4 Algoritmos Genéticos.....	19
2.4.1 Implementação dos algoritmos genéticos	19
2.4.2 Inicialização e tamanho das populações.....	20
2.4.3 Função de avaliação	20
2.4.4 Operadores Genéticos de Seleção	21
2.4.5 Operadores de Cruzamento e Operadores de Mutação	22
2.5 Redes Neurais Artificiais.....	23
2.5.1 Princípios Básicos de Rede Neurais.....	23
2.5.2 Arquiteturas de Redes Neurais.....	24
2.5.3 Paradigmas de Aprendizagem.....	26
3 RESULTADOS E DISCUSSÕES	26
3.1 Caracterização do Problema.....	26
3.2 Sintonia pelos métodos de Ziegler – Nichols	27
3.3 Sintonia fina com abordagem de otimização computacional.....	31
3.4 Sintonia por Algoritmo Genético.....	32

3.5 Sintonia por Redes Neurais Artificiais.....	36
4 CONSIDERAÇÕES FINAIS.....	41
REFERÊNCIAS	42

1 INTRODUÇÃO

Os motores de corrente contínua (Motor CC) têm uma vasta aplicação na indústria, isso se deve a sua versatilidade e a relativa facilidade de acionamento. Outros pontos de destaque são: torque elevado e a vasta faixa de velocidade ou a precisão da posição da saída do motor que pode ser obtida por meio do controle desse tipo de máquina (FITZGERALD et al, 2014).

Dentre as várias técnicas de controle para motores CC, destaca-se o controlador Proporcional Integral Derivativo (PID). Segundo Ogata (2010), a aplicabilidade do PID na maioria dos sistemas de controle e a possibilidade de utilização quando o modelo matemático da planta não é conhecido o tornam bastante útil nas aplicações industriais, sendo empregado em mais da metade dos controladores industriais em uso atualmente.

A sintonia correta dos parâmetros do controlador PID é necessária para produzir uma resposta que satisfaça as especificações desejadas, o que faz com que vários trabalhos de pesquisa ainda abordem o estudo sobre esse tipo de controlador e sintonia de seus parâmetros. Entre os métodos clássicos destacam-se: diagrama de resposta em frequência, lugar das raízes e os dois métodos de Ziegler-Nichols (ZN). Esses são baseados em técnicas heurísticas e podem não atingir níveis desejáveis, por dependerem do conhecimento do processo por parte do projetista para a sua análise e posterior ajuste dos ganhos (BINDU et al, 2012; NETO e GOMES, 2010 apud GUIMARÃES, 2013).

Buscando formas mais modernas de sintonizar esses parâmetros, modelos de aprendizagem de máquina têm sido utilizados nesses sistemas, tornando recorrentes estudos que utilizam do poder de predição de inteligência computacional, como os AGs e as RNAs, para investigar a eficiência desses para área de controle industrial.

1.1 Objetivo Geral

Esse trabalho visa aplicar as técnicas de AGs para otimizar a sintonia dos parâmetros de um controlador PID, assim como utilizar RNAs para treinar um modelo que forneça os parâmetros do PID que atenderão às especificações desejadas de tempo de acomodação e máximo sobressinal para um motor CC.

1.2 Objetivos Específicos

- Utilizar técnicas clássicas de controle para obter um comportamento preliminar do sistema;
- Desenvolver um AG que otimize a resposta do sistema de controle do motor CC;

- Utilizar uma RNA para, a partir dos parâmetros fornecidos pelo AG, conseguir prever o comportamento do sistema.

1.2 Estrutura do Trabalho

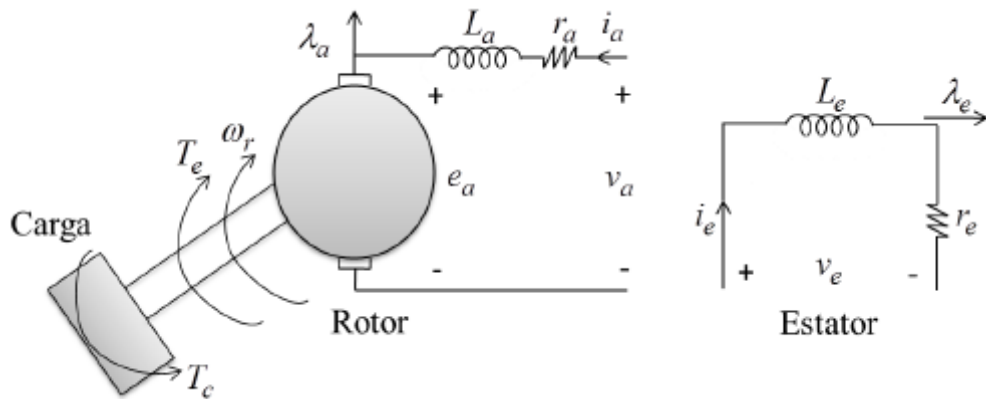
- A primeira seção contém informações introdutórias sobre o tema abordado, sua relevância e os principais objetivos do estudo.
- A segunda seção trata do embasamento teórico necessário para compreender o problema e desenvolver soluções. Nesse ponto serão abordadas questões sobre a modelagem do motor de corrente contínua, o comportamento dos controladores PID, os métodos clássicos de Ziegler-Nichols para sintonia desses controladores e os conceitos básicos das técnicas computacionais de AGs e RNAs.
- A terceira seção traz os processos utilizados durante a pesquisa, os resultados obtidos com os diferentes métodos de sintonia e a comparação entre as eficiências desses.
- Por fim, na quarta seção são apresentadas as considerações finais sobre a pesquisa desenvolvida.

2 REVISÃO BIBLIOGRÁFICA

2.1 Modelagem do Motor de Corrente Contínua

O motor de corrente contínua (Motor CC) é um componente capaz de produzir uma saída mecânica, a partir de uma entrada elétrica. Basicamente, a corrente contínua fornecida é convertida em energia mecânica rotativa, na qual uma fração significativa do torque gerado no rotor está disponível para adicionar carga extra (DORF e BISHOP, 2001). Uma modelagem comum para esse sistema é representada na Figura 1 que consideram as leis de Kirchoff e Newton. De acordo com a Figura 1, na constituição em paralelo o Motor CC é constituído por: circuito de estator (Campo) e rotor (Armadura).

Figura 1 – Modelo do circuito elétrico do Motor CC



Fonte: Guimarães (2013).

Aplicando a Lei de Kirchoff na malha do circuito rotor, temos que:

$$v_a = r_a i_a + L_a \frac{di_a}{dt} + e_a \quad (1)$$

Sabendo que a relação eletromecânica estabelecida para o motor CC é dada pelas seguintes equação para o torque sobre o eixo do rotor:

$$T = K_1 i_e \quad (2)$$

E que a relação entre a tensão aplicada ao motor e a velocidade angular do rotor é dada por:

$$e_a = K_2 \omega \quad (3)$$

A equação para o movimento do motor pode ser assim descrita como:

$$J_m \frac{dw}{dt} + bw = K_1 i_a \quad (4)$$

Usando as equações (3) para reescrever (1):

$$L_a \frac{di_a}{dt} + r_a i_a = v_a - K_2 w \quad (5)$$

Fazendo $w = \frac{d\theta}{dt}$ e $K_1 = K_2 = K_b$. Em (4) e (5) no domínio da frequência tem-se:

$$J_m s^2 \theta(s) + Bs \theta(s) = K_b I(s) \quad (6)$$

$$L_a s I(s) + R_a I(s) = V(s) - K_b s \theta(s) \quad (7)$$

Substituindo (6) em (7) e usando algumas simplificações algébricas:

$$\left[\frac{J_m s^2 \theta(s) + Bs \theta(s)}{K_b} \right] (L_a s + R_a) = V(s) - K_b s \theta(s) \quad (8)$$

$$\left[\frac{(J_m s^2 + Bs)(L_a s + R_a)}{K_b} + K_b s \right] \theta(s) = V(s) \quad (9)$$

$$\frac{\theta(s)}{V(s)} = \frac{K_b}{[J_m L_a s^3 + (J_m R_a + B L_a) s^2 + (K_b^2 + R_a B) s]} \quad (10)$$

‘Que será a função de transferência usada para o motor CC nesse estudo. Onde:

- θ é posição do eixo do rotor;
- V é a tensão;
- J_m é o momento de inércia do rotor;
- B é o coeficiente de amortecimento;
- K_b é a constante da força eletromotriz;
- L_a é a indutância de armadura;
- R_a é a resistência de armadura .

2.2 Controladores PID

O controlador Proporcional Integral Derivativo (PID) são amplamente usados na indústria devido sua praticidade e notável eficácia. Ogata (2010) estima que em mais da metade dos controladores industriais em uso atualmente se empregue os esquemas de controle PID ou PID modificado. A grande utilidade do PID está no fato de ele poder ser aplicado a maioria dos sistemas e a possibilidade de ser aplicado em plantas nas quais o modelo matemático não é conhecido. A sintonização pode ser realizada em campo e seu controle se deve a três ações que modificam a resposta para que a pretendida seja alcançada. A seguir serão detalhes os efeitos das ações Proporcional, Integral e Derivativa sobre as respostas de sistemas, assim como a formulação matemática correspondente a representação dessas no domínio do tempo e no domínio da frequência.

2.2.1 Ações Proporcional-Integral-Derivativa

O controlador PID é composto por três ações de controle. A ação proporcional caracteriza-se pelo aumento do sobressinal, diminuição do tempo de subida e erro em regime estacionário, além de pequenas variações no tempo de acomodação (CANIATO, 2006), sendo representada pela equação:

$$u(t) = K_p e(t) \quad (11)$$

Aplicando a transformada de Laplace:

$$\frac{U(s)}{E(s)} = K_p \quad (12)$$

A ação integral é utilizada em conjunto com a ação proporcional, caracteriza-se pela diminuição do tempo de subida, aumento do tempo de acomodação, eliminação do erro de regime estacionário e aumento do sobressinal (CANIATO, 2006).

É representada pelas equações a seguir, que correspondem o domínio do tempo e a transformada de Laplace, respectivamente:

$$u(t) = K_i \int e(t) dt \quad (13)$$

$$\frac{U(s)}{E(s)} = \frac{K_i}{s} \quad (14)$$

A ação derivativa também é utilizada em conjunto com a ação proporcional e caracteriza-se por uma pequena variação no tempo de subida, diminuição do tempo de acomodação, pequena variação do erro de regime estacionário e diminuição do sobressinal (CANIATO, 2006). É representada pela equação:

$$u(t) = K_d \frac{de(t)}{dt} \quad (15)$$

E tem como transformada de Laplace:

$$\frac{U(s)}{E(s)} = K_d s \quad (16)$$

A junção de todas essas ações resulta no controlador PID, que é descrito pela equação:

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt} \quad (17)$$

No domínio da frequência, tem-se:

$$\frac{U(s)}{E(s)} = \frac{K_d s^2 + K_p s + K_i}{s} \quad (18)$$

Na qual, K_p , K_i e K_d são os parâmetros do controlador ajustáveis e capaz de modificar o comportamento do sistema.

Pode ser descrito ainda em função do ganho proporcional K_p , de tempo integral T_i e de tempo derivativo T_d obtendo a equação (19).

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (19)$$

O método de Ziegler-Nichols, usado para realizar esse ajuste, será descrito a seguir.

2.3 Métodos de Sintonia de Ziegler-Nichols

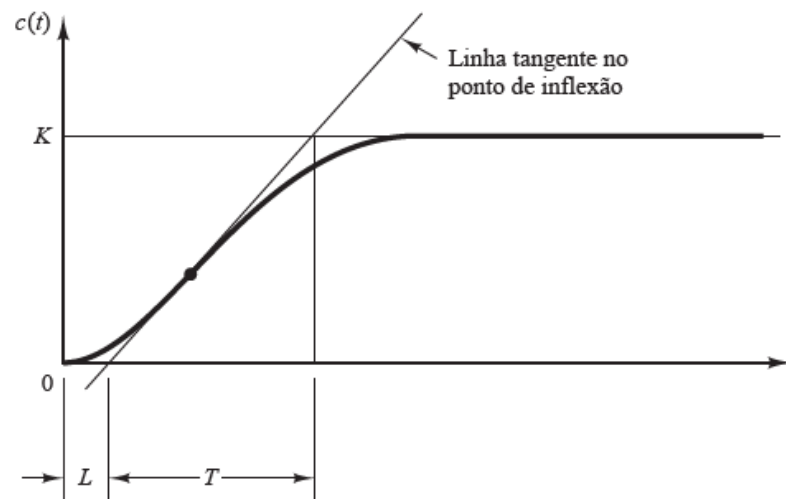
Ogata (2010) destaca as regras de sintonia de Ziegler-Nichols (ZN) como uma importante ferramenta para sintonizar os parâmetros de um controlador PID em casos que o modelo matemático da planta não pode ser obtido facilmente, nela os valores de K_p , K_i e K_d são ajustados baseados na resposta experimental ao degrau ou no valor do ganho K_p , que resulta em uma estabilidade marginal quando se usa apenas uma ação proporcional. Os parâmetros sugeridos vão gerar uma operação estável no sistema, porém podem apresentar um máximo

sobressinal muito elevado, tornando necessário um ajuste fino dos parâmetros. Desse modo, as regras de ZN fornecem um ponto de partida para essa sintonia e podem ser divididas em dois métodos descritos seguir.

2.3.1 Primeiro Método

Ogata (2010) descreve que no primeiro método é necessário inicialmente obter a resposta da planta a um degrau unitário e verificar se a resposta gerada tem um aspecto de “S”, observado na Figura 2, o que indicaria e que o método é aplicável.

Figura 2 – Resposta do sistema em malha aberta



Fonte: Ogata (2010)

A curva em formato de “S” pode ser descrita pelas constantes L , o atraso, e a constante de tempo T . Essas constantes são encontradas desenhando uma linha que tangencie o ponto de inflexão da curva, o atraso será a intersecção dessa linha com o eixo do tempo e o T a intersecção com a linha $c(t) = K$. Os valores de K_p , T_i e T_d devem ser calculados seguindo a Tabela 1.

Tabela 1 – Parâmetros do controlador PID - 1º Método de Ziegler-Nichols

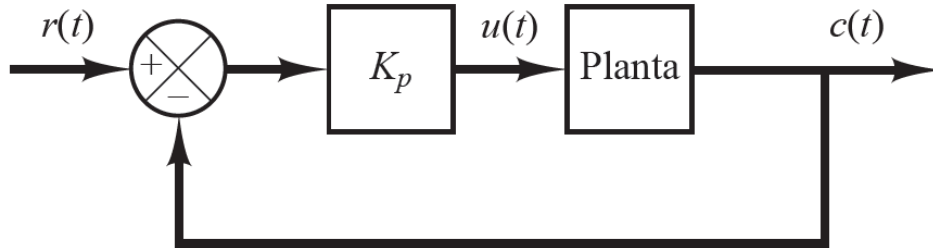
Tipo de Controlador	K_p	T_i	T_d
P	T/L	∞	0
PI	$0,9 T/L$	$T/0,3$	0
PID	$1,2 T/L$	$2L$	$0,5L$

Fonte: Adaptado de Ogata (2010).

2.3.1 Segundo Método

No segundo método define-se inicialmente $T_i = \infty$ e $T_d = 0$, de forma que apenas a ação proporcional seja aplicada na planta como na Figura 3.

Figura 3 – Modelo do diagrama de blocos para o 2º método de ZN



Fonte: Ogata (2010).

O valor de K_p deve ser aumentado até que a saída exiba uma oscilação sustentada, o que indica que a resposta do sistema se tornou marginalmente estável. O ganho nesse ponto é chamado de ganho crítico K_{cr} , ele e o período da oscilação P_{cr} serão utilizados para determinar os parâmetros de acordo com a Tabela 2.

Tabela 2 – Parâmetros do controlador PID - 2º Método de Ziegler-Nichols

Tipo de Controlador	K_p	T_i	T_d
P	$0,5K_{cr}$	∞	0
PI	$0,45K_{cr}$	$\frac{1}{1,2}P_{cr}$	0
PID	$0,6K_{cr}$	$0,5P_{cr}$	$0,125P_{cr}$

Fonte: Adaptado de Ogata (2010).

2.4 Algoritmos Genéticos

2.4.1 Implementação dos algoritmos genéticos

Os AGs são técnicas computacionais, que começaram a ser desenvolvidas por John Henry Holland's (1975), e que buscam otimizar solução para problemas complexos. Foram baseados na teoria da evolução natural de Darwin, que diz que indivíduos mais aptos possuem mais chances de sobreviver e perpetuar seu código genético, e na reprodução genética, que diz que cada indivíduo carrega uma identificação em seu código genético e que esse passa para

gerações seguintes. Resumidamente, o AG procura boas soluções para problemas complexos por meio da evolução da população de soluções, da sobrevivência do indivíduo mais apto e do cruzamento aleatório de informações (GOLDSCHMIDT, 2010; MICHALEWICZ, 1994).

Dessa forma, em um paralelo entre a evolução natural e os AGs o meio ambiente em que o indivíduo é inserido seria o problema a se resolver, a população seria o conjunto de soluções, o indivíduo seria uma possível solução, o cromossomo seria a estrutura de dados que representa a solução, a reprodução sexual da evolução natural seria representado pelo operador de cruzamento, as mutações seriam o operador de mutação e as gerações seriam os ciclos.

Os algoritmos genéticos são caracterizados pelos seguintes componentes:

- Problema;
- Representação das soluções do problema;
- Decodificação do cromossomo;
- Função de avaliação;
- Seleção;
- Operadores genéticos;
- Técnicas e parâmetros;
- Critério de inicialização da população.

Os tópicos seguintes detalharão alguns deles.

2.4.2 Inicialização e tamanho das populações

A inicialização é a geração da população inicial que será executada no AG, assim é importante a criação de uma população inicial adequada para que o algoritmo tenda a convergir para uma solução eficaz. Um método comum utilizado para inicialização é a formação de indivíduos iniciais aleatoriamente gerados. Quanto ao tamanho da população é necessário teste para verificar valores que favoreçam a convergência para a solução desejável (COSTA *et al*, 1999; RENNER E EKART, 2003).

2.4.3 Função de avaliação

A função de avaliação determina a aptidão (do inglês *fitness*) de um cromossomo e se esse tem potencial para fornecer a obtenção de melhores repostas para o sistema. Em sistemas de controle, são utilizadas comumente as seguintes configurações de *fitness*:

- Integral do erro absoluto ponderado no tempo (ITAE – do inglês *Integral of Squared Error*).

$$\text{ITAE} = \int t \times |e(t)| dt \quad (20)$$

- Integral do erro absoluto (IAE – do inglês *Integral of Absolute Error*)

$$\text{IAE} = \int |e(t)| dt \quad (21)$$

- Integral do erro ao quadrado vezes o tempo (ITSE – do inglês *Integral of Time Multiply Squared Error*)

$$\text{ITSE} = \int t \times e^2(t) dt \quad (22)$$

- Integral do erro ao quadrado (ISE – do inglês *Integral of Squared Error*)

$$\text{ISE} = \int e^2(t) dt \quad (23)$$

Em que $e(t)$ é o sinal de erro proveniente da diferença entre o sinal de saída e o sinal de referência. Como a sintonia de controlador busca parâmetros que minimizem o máximo sobressinal e tempo de acomodação, esses também podem ser usados como índice de desempenho a ser minimizado pela função *fitness* (GUIMARÃES, 2013).

2.4.4 Operadores Genéticos de Seleção

É nesse ponto em que, após a fase de avaliação, os indivíduos que mostraram ter melhor aptidão poderão ser selecionados para a aplicação dos demais operadores genéticos. Dentre as técnicas de seleção usadas podem ser destacadas (TENG *et al*, 2003):

- Seleção por Ordenamento: os indivíduos são ordenados por ordem de seu valor de função de aptidão e os n melhores indivíduos da população substituem os n indivíduos com pior valor de função de aptidão.
- Seleção pelo método da Roleta: os indivíduos são representados de forma proporcional ao seu valor de aptidão e ela define a probabilidade de que permaneçam na próxima geração. Sendo assim, se f_i é a avaliação do i -ésimo indivíduo a probabilidade desse ser selecionado é calculada por:

$$p_i = \frac{f_i}{\sum_{i=1}^n f_i} \quad (24)$$

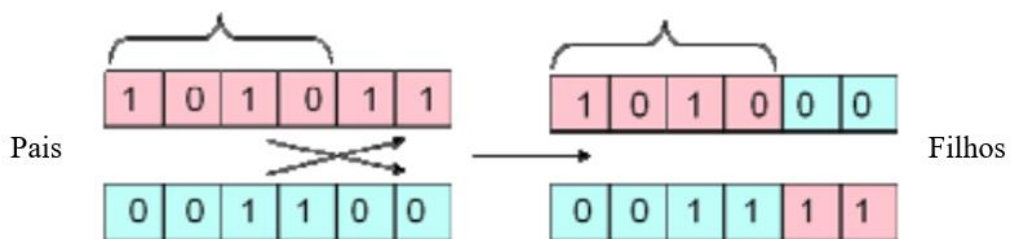
- Seleção por Torneio: Nesta seleção, uma parcela aleatória dos indivíduos da população é escolhida e o cromossomo que apresente maior função de aptidão será selecionado nos torneios até que os n melhores indivíduos sejam escolhidos para prosseguir no processo evolutivo.
- Elitismo: É o processo em que parcela de uma nova população pode ser criada simplesmente copiando indivíduos selecionados da população atual.

2.4.5 Operadores de Cruzamento e Operadores de Mutação

Quando os indivíduos mais aptos são selecionados, o AG deve criar novas soluções, gerar descendentes, combinando as informações dos cromossomos dos pais, essa combinação pode ser feita por cruzamento (*crossover*) ou por mutação. A maneira a qual o material genético dos genitores será combinado é particular de cada problema, porém existem operadores mais tradicionais como: o *crossover* de 1 ponto, o *crossover* de 2 pontos, o *crossover* de n pontos, o *crossover* uniforme, o *crossover* aritmético e *crossover* de média (GOLDSCHMIDT, 2010; YALCINOZ E ALTUN, 2002; DAGLI E SCHIERHOLT, 1997).

No *crossover* de 1 ponto, por exemplo, são combinadas duas soluções a partir de um ponto de corte aleatório, assim dois pais têm sua troca de informação genética a partir desse ponto de posição e geram dois filhos como ilustrado na Figura 4.

Figura 4 – Exemplo de aplicação do *crossover* de 1 ponto



Fonte: Adaptado de Goldschmidt (2010)

Na Figura 4 os pais são representados em rosa e azul, o ponto de cruzamento está localizado no quarto dígito, assim os dois filhos gerados têm quatro dígitos do primeiro pai e dois do segundo e quatro dígitos do segundo pai e dois do primeiro, respectivamente.

Já a mutação é o processo que contribui para a diversidade genética de uma espécie, alterando o valor do gene por outro valor aleatório distribuído, normalmente dentro de uma faixa pré-estabelecida bem baixa ($<1\%$), visto que uma taxa alta poderia causar instabilidade no modelo e fazer com que bons indivíduos fossem descartados (NASCIMENTO, 2010). Similar ao cruzamento existem vários tipos de mutação, a mutação clássica aplicada das representações binárias inverte o valor de um bit aleatório, ou na representação real substitui um gene sorteado por um número aleatório. (GOLDSCHMIDT, 2010).

2.5 Redes Neurais Artificiais

2.5.1 Princípios Básicos de Rede Neurais

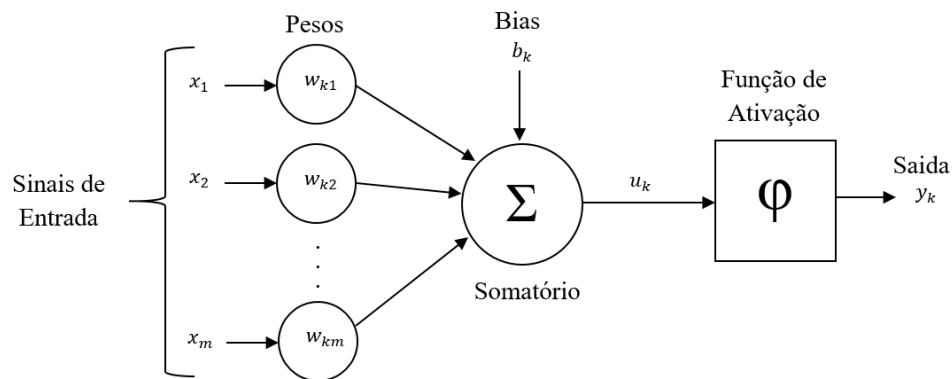
O estudo das RNAs surgiu do reconhecimento de que o cérebro humano processa informação de uma forma inteiramente diferente do computador convencional e de que esse pode realizar certos processamentos de forma mais rápida, por ser altamente complexo, não-linear e paralelo (HAYKIN, 2001).

O primeiro trabalho sobre RNAs foi realizado em 1943, por McCulloch e Pitts e tinha como objetivo construir um modelo matemático baseado no neurônio biológico. As conclusões desse estudo originaram os princípios de ativação ou não do neurônio, caso sua saída ultrapasse um valor limite, e o de que a sinapse é a responsável por inibir ou excitar um neurônio naquele instante, o que originou o conceito dos pesos de neurônios artificiais. Desde então, vários trabalhos que foram realizados sobre o tema, o desenvolvimento do modelo de *perceptron*, realizado por Frank Rosenblatt em 1958, a ideia de aprendizado não-supervisionado, por Teuvo Kohonen em 1972, as conexões de realimentação (*feedback*), por John Hopfield em 1982 (CARDON, MÜLLER e NAVAUX, 1994).

A estrutura básica do neurônio artificial é apresentada na Figura 5, sendo composto pelos seguintes elementos:

- a) Sinais de entrada ($x_1, x_2, \dots, x_m$);
- b) Pesos dos nós ($w_{k1}, w_{k2}, \dots, w_{km}$);
- c) Somatório (Σ);
- d) Bias (b_k);
- e) Função de ativação (ϕ);
- f) Saída (y_k).

Figura 5 – Estrutura básica do neurônio artificial



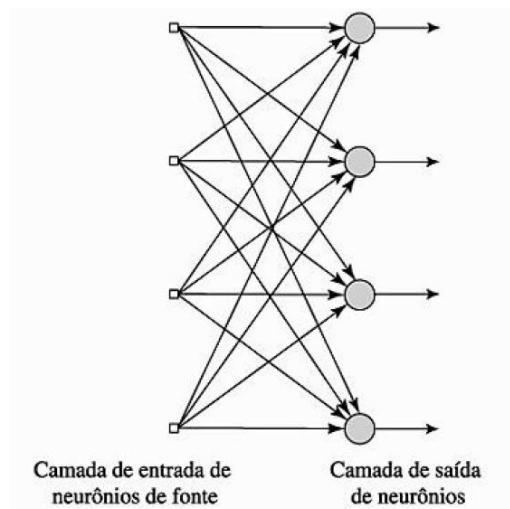
Fonte: Adaptado de Haykin (2001).

2.5.2 Arquiteturas de Redes Neurais

A arquitetura da rede refere-se a como os neurônios estão distribuídos e a como eles interagem entre si e está intimamente ligada ao algoritmo de aprendizagem usado para treinar a rede. As três classes de arquiteturas de redes são: redes de alimentação direta com camada única (*feedforward*), redes de alimentação direta com múltiplas camadas (*MLP- Multilayer Perceptron Network*) e redes recorrentes (*feedback*) (HAYKIN, 2001).

As redes de alimentação direta com camada única são a forma mais simples de uma rede em camadas e apresenta uma camada de entrada e uma camada de saída. É chamada de camada única, pois a camada de entrada não é contabilizada, visto que não ocorre nenhuma computação nela. Essa topologia pode ser observada na Figura 6.

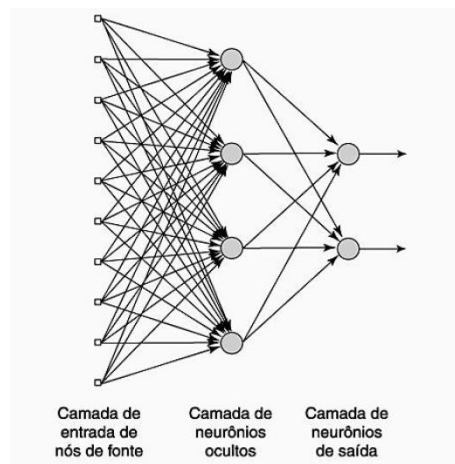
Figura 6 – Rede de alimentação direta com camada única.



Fonte: Haykin (2001).

As redes de alimentação direta com múltiplas camadas são formadas por uma camada de neurônios de entrada, uma ou mais de neurônios oculto e uma de neurônios de saída. Segundo Haykin (2001), adicionar camadas ocultas torna a rede capaz de extrair e estatísticas de ordem elevadas sendo uma prática adotada quando há muitas variáveis de entrada do problema, o que torna o tamanho da camada de entrada grande. A Figura 7 ilustra essa classe de redes.

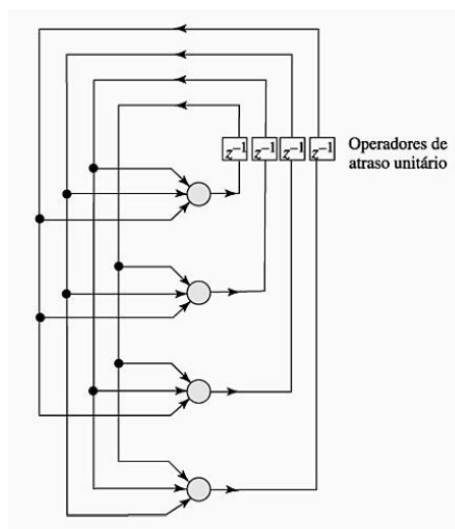
Figura 7 – Rede com múltiplas camadas



Fonte: Haykin (2001).

As redes recorrentes se caracterizam por ter pelo menos um laço de alimentação, o que pode causar um impacto significativo na capacidade de aprendizagem e desempenho, além de admitir que a rede neural tenha unidades não lineares. A Figura 8 ilustra uma rede recorrente.

Figura 8 – Rede recorrente



Fonte: Haykin (2001).

2.5.3 Paradigmas de Aprendizagem

Para Haykin (2001) o conhecimento se refere a informação que se tem armazenada ou ao modelo que foi utilizado para interpretar, prever e responder apropriadamente ao mundo exterior. Dessa forma, para rede a aprendizagem ocorre quando os parâmetros e pesos são ajustados através de um processo de estimação pelo ambiente que a rede está inserida.

O modelo usado para treinar uma rede pode ser caracterizado pelo tipo de entrada, forma de conexão e o paradigma de aprendizado aplicado. As entradas podem ser binárias ou intervalares, a conexão por alimentação a frete, retroalimentação ou competitiva e o tipo de paradigma de aprendizado pode ser supervisionado ou não-supervisionado.

Na aprendizagem supervisionada há uma definição sobre a saída que se deseja para rede, o que leva ao ajuste de pesos, de forma que seja representado o sinal desejado. São exemplos de modelos que seguem o aprendizado supervisionado: o *perceptron* e *backpropagation* (CARDON, MÜLLER e NAVAUX, 1994).

Já no aprendizado não supervisionado é feita uma distribuição de probabilidades de forma a encontrar padrões. São exemplos de modelos que seguem o aprendizado não-supervisionado: o modelo de Hopfiel, o de Kohone o ART (*Adaptive Resonance Theory*).

Para esse trabalho foi utilizado o paradigma de aprendizagem supervisionado, uma arquitetura do tipo direta (*feedforward*), como conectividade de no mínimo uma camada oculta (*Multilayer*) e foram testados os algoritmos de treinamento de retroalimentação (*backpropagation*) e o algoritmo de *Levenberg-Marquardt*.

3 RESULTADOS E DISCUSSÕES

3.1 Caracterização do Problema

Para realizar o controle de posição do eixo a partir de uma tensão fornecida foi analisado o mesmo sistema utilizado no trabalho envolvendo ajuste de parâmetros de Guimarães (2013) e desenvolvido anteriormente por Adhikari et al (2012). Os parâmetros do estão dispostos na Tabela 3.

Tabela 3 – Parâmetros do Motor CC

Parâmetros	Valor	Unidade
Momento de Inércia J	0,022	Kg-m ² /rad
Coefficiente de Amortecimento B	0,5x10 ⁻³	N-m/ (rad/s)

Constante da Força Eletromotriz Kb	1,2	V.s/rad
Resistência Elétrica R	2,45	Ω
Indutância Elétrica L	0,035	H

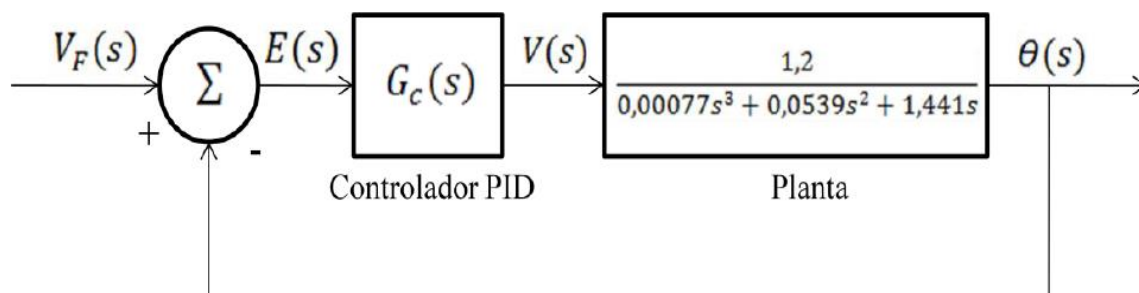
Fonte: Adaptado de Guimarães (2013)

Substituindo os parâmetros contidos na Tabela 3 na função de transferência mostrada na segunda seção deste trabalho na Equação 10, pode-se obter o seguinte modelo matemático para o funcionamento do sistema em questão:

$$\frac{\theta(s)}{V(s)} = \frac{1,2}{0,00077s^3 + 0,0539s^2 + 1,441s} \quad (25)$$

O diagrama de blocos representativo do sistema de controle a ser projetado é representado conforme a Figura 9:

Figura 9 – Representação genérica do sistema de controle a ser projetado

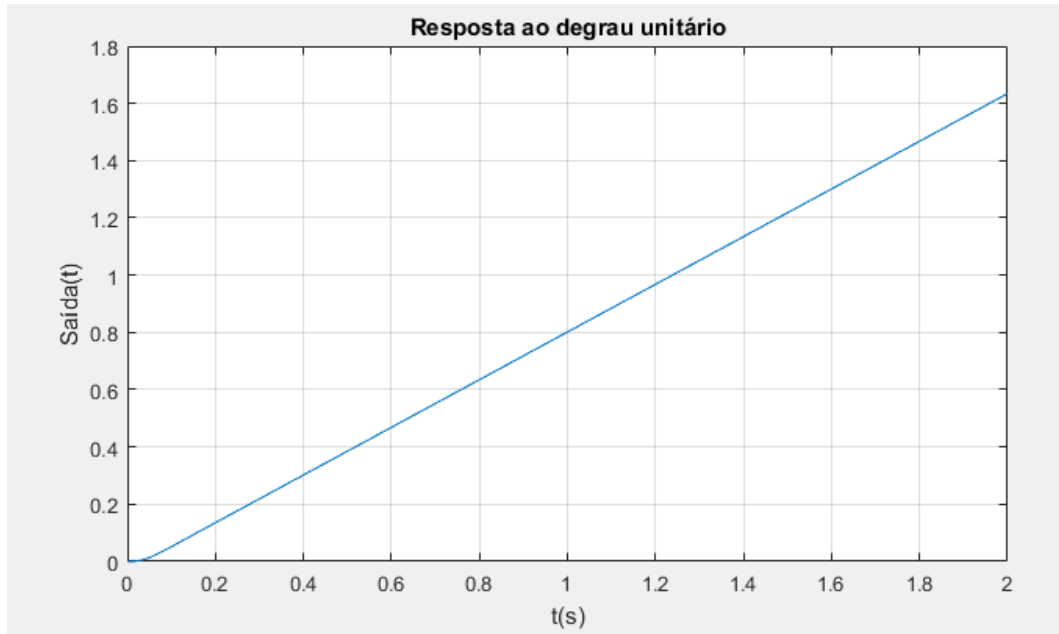


Fonte: Autoria própria.

3.2 Sintonia pelos métodos de Ziegler – Nichols

Para saber qual dos dois métodos de Ziegler-Nichols utilizar na sintonia inicial dos parâmetros é necessária a resposta do sistema à entrada ao degrau. Dessa forma, provavelmente o segundo método será adotado tendo em vista que a planta em análise possui integrador e polos complexos conjugados dominantes. Portanto, para confirmar o descarte do primeiro método, a resposta do sistema à entrada ao degrau será simulada no software MATLAB, o resultado gráfico está ilustrado na Figura 10.

Figura 10 – Resposta do sistema à entrada ao degrau sem controlador



Fonte: Autoria própria.

A análise do comportamento curva do sistema ao degrau unitário não apresenta aspecto de “S”, o que confirma que primeiro método não poderia ser utilizado. Sendo assim, o ponto de partida para a aplicação do segundo método de Ziegler-Nichols é adotar $Ti=\infty$ e $Td=0$, resultando na seguinte função de transferência representada na Equação (26):

$$\frac{\theta(s)}{V(s)} = \frac{1,2K_p}{0,00077s^3 + 0,0539s^2 + 1,441s + 1,2K_p} \quad (26)$$

Para encontrar o valor K_p que torna o sistema marginalmente estável aplica-se o critério de estabilidade de Routh, para o qual se utiliza a equação característica do sistema em malha fechada dada pela Equação (27):

$$0,00077s^3 + 0,0539s^2 + 1,441s + 1,2K_p \quad (27)$$

O critério de Routh é calculado conforme se mostra a seguir:

s^3	0,00077	1,441
s^2	0,0539	1,2 K_p
s^1	$\frac{0,0776699 - 0,000924K_p}{0,0539}$	

$$s^0 \quad 1,2K_p$$

Analisando a primeira coluna é possível determinar que as oscilações sustentadas existirão se $K_p = 84,05$. Portanto, o valor crítico é dado por $K_{cr} = 84,05$ e a equação característica resulta em:

$$0,00077s^3 + 0,0539s^2 + 1,441s + 84,05 \quad (28)$$

Pela qual pode-se encontrar a frequência de oscilação sustentada substituindo $s = jw$. Assim:

$$0,00077(jw)^3 + 0,0539(jw)^2 + 1,441(jw) + 84,05$$

$$w = 39,48 \text{ rad/s}$$

Dessa forma, o período de oscilação sustentada é:

$$P_{cr} = \frac{2\pi}{w} = \frac{2\pi}{39,48} = 0,1591 \text{ s} \quad (29)$$

Utilizando a Tabela 2 e os valores encontrado para K_{cr} e P_{cr} , pode-se determinar os parâmetros do controlador da seguinte forma:

$$K_p = 0,6K_{cr} = 0,6 \times 84,05 = 51,03$$

$$T_i = 0,5P_{cr} = 0,5 \times 0,1591 = 0,07955 \text{ s}$$

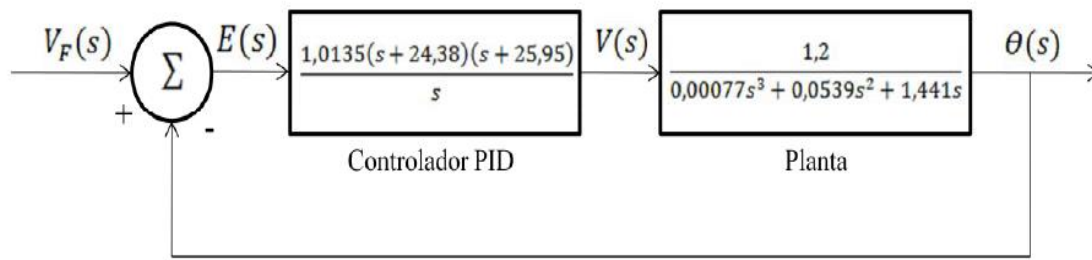
$$T_d = 0,125P_{cr} = 0,125 \times 0,1591 = 0,01988 \text{ s}$$

Substituindo na função de transferência do controlador PID, tem-se:

$$\begin{aligned} G_c &= 51,03 \left(1 + \frac{1}{0,07955s} + 0,01988s \right) \\ G_c &= \frac{1,0135(s^2 + 50,34s + 632,91s)}{s} \\ G_c &= \frac{1,0135(s + 24,38)(s + 25,95)}{s} \end{aligned} \quad (30)$$

Que é a função de transferência do controlador e está representada no diagrama da Figura 11:

Figura 11 – Diagrama de blocos do sistema sintonizado pelo método de Ziegler-Nichols



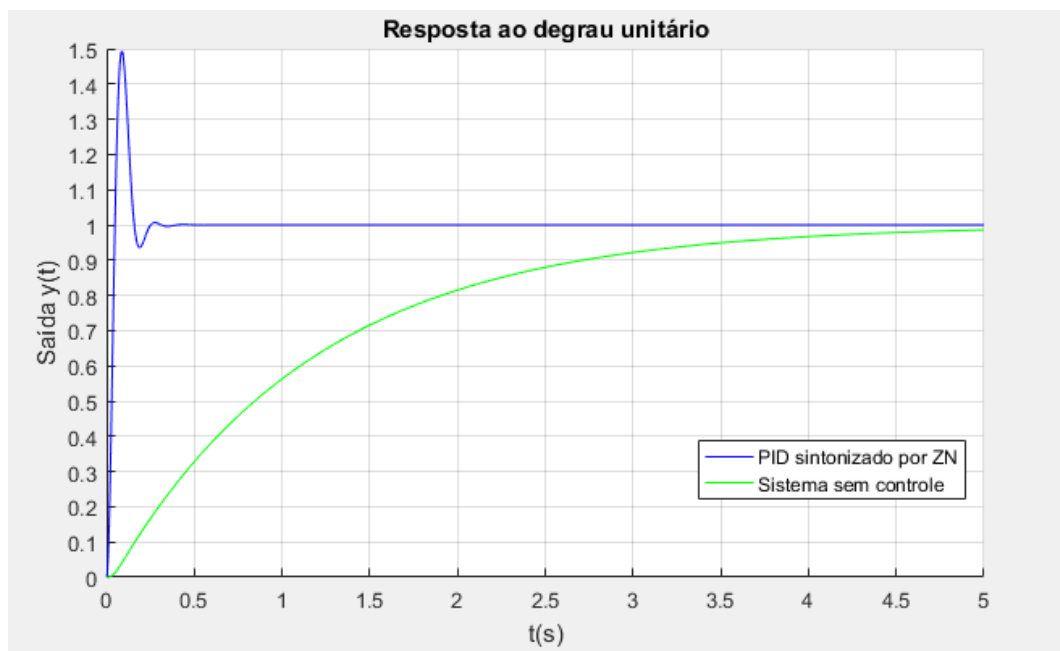
Fonte: Autoria própria.

Portanto, a função de transferência de malha fechada é dada por:

$$\frac{\theta(s)}{V_F(s)} = \frac{1,2162s^2 + 61,222s + 769,745}{0,00077s^4 + 0,0539s^3 + 2,6572s^2 + 61,222s + 769,745} \quad (31)$$

A Figura 12 mostra a resposta do sistema com o controlado sintonizado pelo segundo método de ZN e a resposta ao sistema sem controle.

Figura 12 – Resposta do sistema sintonizado pelo método de Ziegler-Nichols



Fonte: Autoria própria.

Pode-se observar que essa sintonia resultou em uma resposta rápida de 0,230s, mas que o máximo sobressinal é acentuado, chegando a 49,19%. O sistema sem controle, por sua vez, apresenta uma resposta lenta, sendo o tempo de acomodação de 4.59s. Dessa forma se faz necessária a realização de uma sintonia fina dos parâmetros que objetiva reduzir o sobressinal sem tornar o sistema muito mais lento.

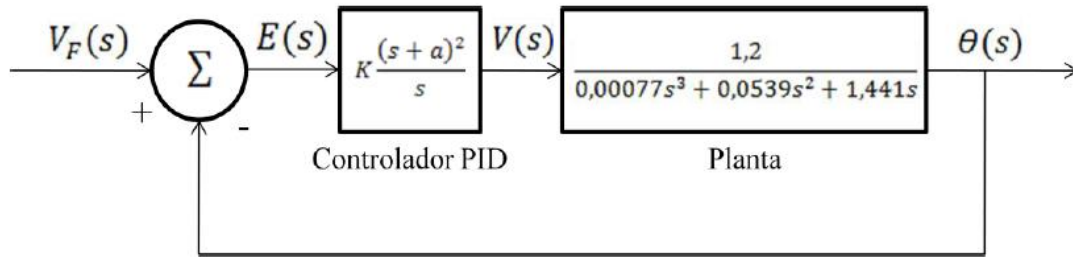
3.3 Sintonia fina com abordagem de otimização computacional

A sintonia fina consiste de uma rotina desenvolvida em MATLAB por Ogata (2010), na qual se aplica o controlador PID genérico descrito pela equação a seguir à função de transferência de malha aberta de um sistema para encontrar uma combinação de valores de K e a de modo que, o sistema de malha fechada tenha um máximo sobressinal e tempo de acomodação dentro de intervalos pré-determinados.

$$G_c(s) = K \frac{(s + a)^2}{s} \quad (32)$$

O sistema está representado na Figura 13.

Figura 13 – Diagrama de blocos do sistema sintonizado pelo ajuste fino.



Fonte: Autoria própria.

Na prática, a rotina computacional adota uma região de teste para K e a , cria uma matriz contendo os intervalos desses valores e os valores de máximo sobressinal e tempo de acomodação e ordena esses valores em ordem crescente de forma que seja fácil identificar o máximo sobressinal e tempo de acomodação que melhor se adequa.

Para esse ajuste foram consideradas combinação que tivesse tempo de acomodação inferior a 2 segundos e a região testada consistiu na mesma utilizada por Ogata (2010):

$$2 \leq K \leq 50 \text{ e } 0,05 \leq a \leq 2$$

Os valores obtidos foram de $K=5$ e $a=1,9$. Assim, por analogia:

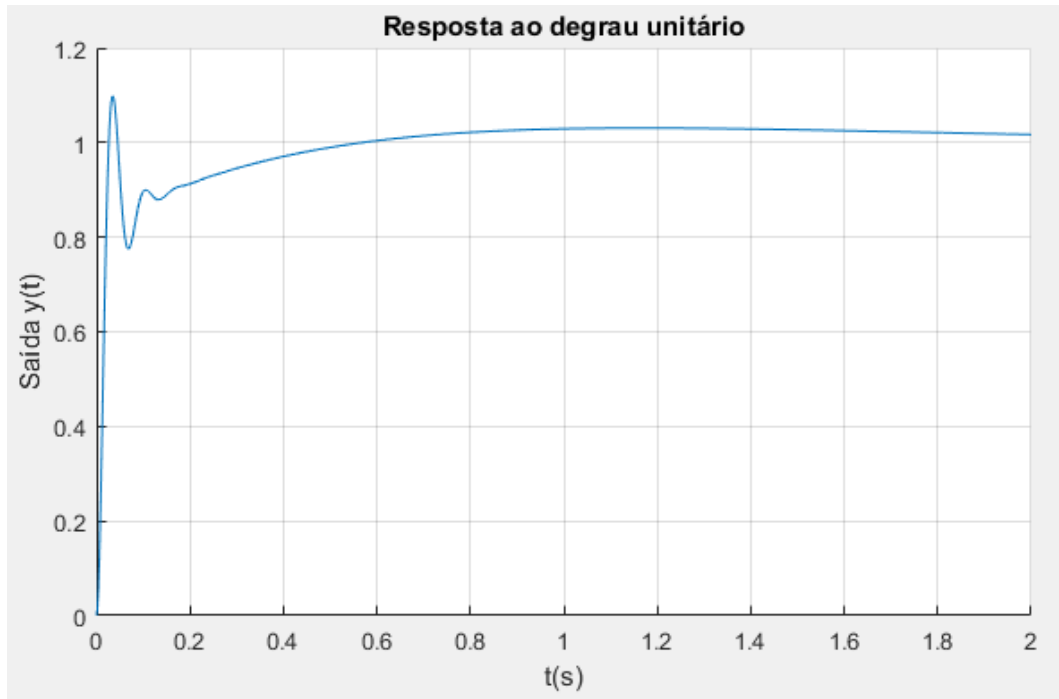
$$K_p = 2K * a = 2 * 5 * 1,9 = 19$$

$$K_i = K * a^2 = 5 * 1,9^2 = 18,05$$

$$K_d = K = 5$$

Os valores encontrados para os parâmetros foram: $Kp = 19$, $Ki = 18,05$ e $Kd = 5$. A resposta do sistema a entrada ao degrau é verificada na Figura 14.

Figura 14 – Resposta do sistema sintonizado pelo ajuste fino



Fonte: Autoria própria.

A partir dos resultados obtidos, pode-se observar que a resposta do sistema sintonizado pelo ajuste fino conseguiu diminuir o sobressinal para 9,85%, o que resultou em uma resposta mais lenta (1,85s) em relação à obtida pelo método de Ziegler-Nichols (0,23s).

3.4 Sintonia por Algoritmo Genético

A inicialização do Algoritmo Genético foi realizada com uma população aleatória em forma binária, constituída por valores de (Kp, Ki, Kd) , a qual será avaliada e, posteriormente submetida a todos os operadores genéticos até atingir o número de gerações desejadas (GUIMARÃES, 2013).

Primeiramente, o número binário é convertido em número real para que todas as possíveis respostas sejam avaliadas através do cálculo da função de *fitness*. A expressão matemática utilizada para converter o número binário a um número real é apresentada na Equação (33) a seguir (LINDEN, 2008).

$$Real = inf + + \left(\frac{sup - inf}{2^k - 1} \right) \quad (33)$$

Em que:

Real = valor do número real;

sup= valor superior do intervalo;

inf = valor inferior do intervalo;

k = número de bits.

Como o objetivo é encontrar os melhores valores para os ganhos do controlador PID que minimizam os valores de sobressinal e tempo de acomodação, a seguinte função de aptidão para o AG foi utilizada:

$$fitness = Ms + Ta \quad (34)$$

Onde:

Ms = Máximo Sobressinal

Ta = Tempo de Acomodação

A seleção de um bom indivíduo de uma próxima geração foi feita a partir da aptidão ou *fitness*, as populações atuais são organizadas em uma matriz de tamanho - Número de populações (Npop) x Valores de Kp , Ki , Kd , Ms e Ta , são ordenados e selecionados de acordo com o seu valor de função de aptidão.

Para a etapa de cruzamento foram selecionados dois pais a partir de 1 até Npop e combina 50% de cromossomos de cada um dos pais, originando dois novos indivíduos filho, ou seja, foi utilizado o *crossover* de 1 ponto.

Após a etapa de cruzamento, existe a probabilidade de um indivíduo de toda a população ser escolhido para a mutação, caso aconteça, todos os valores binários que representam o seu código genético serão invertidos. Ao final da primeira geração de descendentes, os n melhores indivíduos da atual população serão integrados a uma nova população gerada aleatoriamente (operação de elitismo) para, dessa forma, serem avaliados novamente até que seja atingido o critério de parada (número de gerações previamente programado).

O ponto de partida para a definição do intervalo de busca dos parâmetros (Kp , Ki e Kd) foi obtido dos resultados das técnicas tradicionais de sintonia. Já o número de indivíduos, gerações, taxas de cruzamento, mutação e elitismo, foram inicialmente consultados valores

comumente utilizados na literatura, em especial os utilizados por Guimarães (2013) por se tratar de um estudo similar, porém foram validados a partir de simulações para verificação da convergência dos valores ótimos fornecidos pelo AG.

A Tabela 4 contém os parâmetros utilizados no AG proposto para a sintonia do controlador PID.

Tabela 4 – Parâmetros do Algoritmo Genético

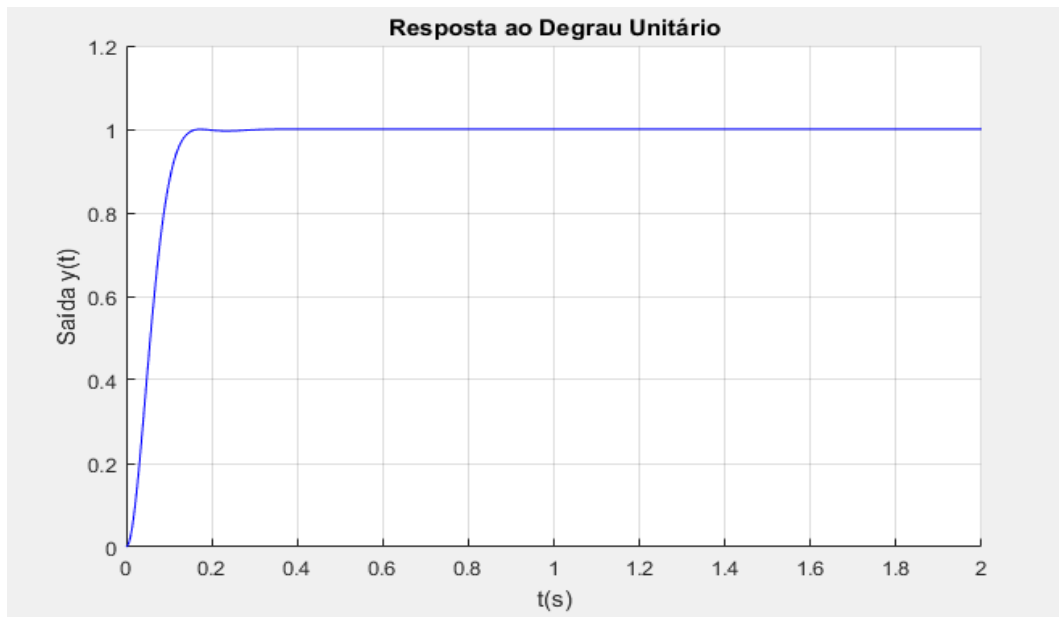
Intervalo de Busca	0 a 20
Número de Indivíduos da População	100
Número de Gerações	50
Genitores	2
Número de Filhos por Geração	2
Pontos de Corte	1
Taxa de Cruzamento (%)	80
Taxa de Mutação (%)	1
Taxa de Elitismo (%)	75
Critério de Parada	Número de Gerações

Fonte: Guimarães, (2013).

Foram obtidos os seguintes parâmetros para o controlador PID e, a resposta do sistema a entrada ao degrau é mostrada na Figura 15.

$$K_p = 19,7490, K_i = 0,1525 \text{ e } K_d = 0,3336.$$

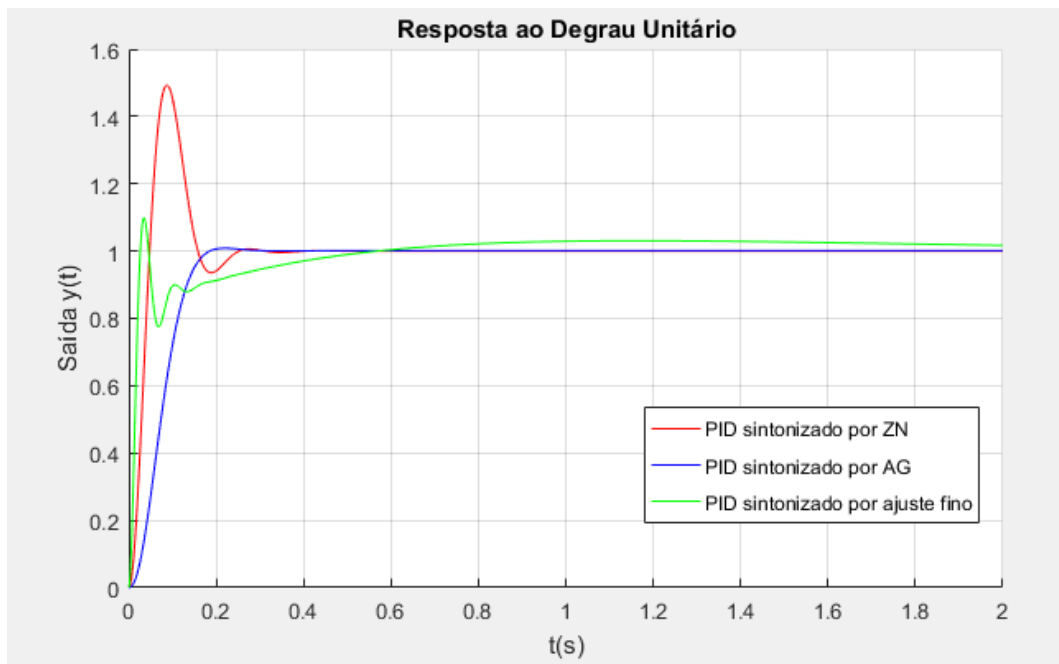
Figura 15 – Resposta do sistema sintonizado por Algoritmo Genético



Fonte: Autoria própria

O sistema sintonizado por AG se mostrou mais rápida (0,135s) do que pelos métodos tradicionais de ZN e sintonia fina, além de conseguir eliminar totalmente o sobressinal. A Figura 16 mostra a comparação entre os três controladores ajustados nesse estudo.

Figura 16 – Comparação entre ajustes de controladores



Fonte: Autoria própria.

Como esperado, a sintonia de parâmetro por ZN ofereceu uma resposta rápida (0,230s), porém com um alto valor de sobressinal (49,19%). A sintonia por ajuste fino ajudou a reduzir esse sobressinal buscando por valores que fornecesse o menor sobressinal, porém que não ultrapassasse o valor de 2s, o que resultou no máximo sobressinal de 6,98% e uma resposta um pouco mais lenta de 1,85s. O AG se mostrou uma ótima técnica de ajuste para esse sistema, conseguindo um valor de máximo sobressinal muito próximo de zero e apresentando uma resposta rápida (0,135s).

3.5 Sintonia por Redes Neurais Artificiais

Os dados utilizados para treinar uma RNA que seja capaz de fornecer os valores de K_p , K_i e K_d (saída) para valores de máximo sobressinal e tempo de acomodação (entrada) foram gerados em forma de banco de dados que parte dos parâmetros encontrados anteriormente e seus valores correspondentes de máximo sobressinal e tempo de acomodação. Os intervalos dos parâmetros são:

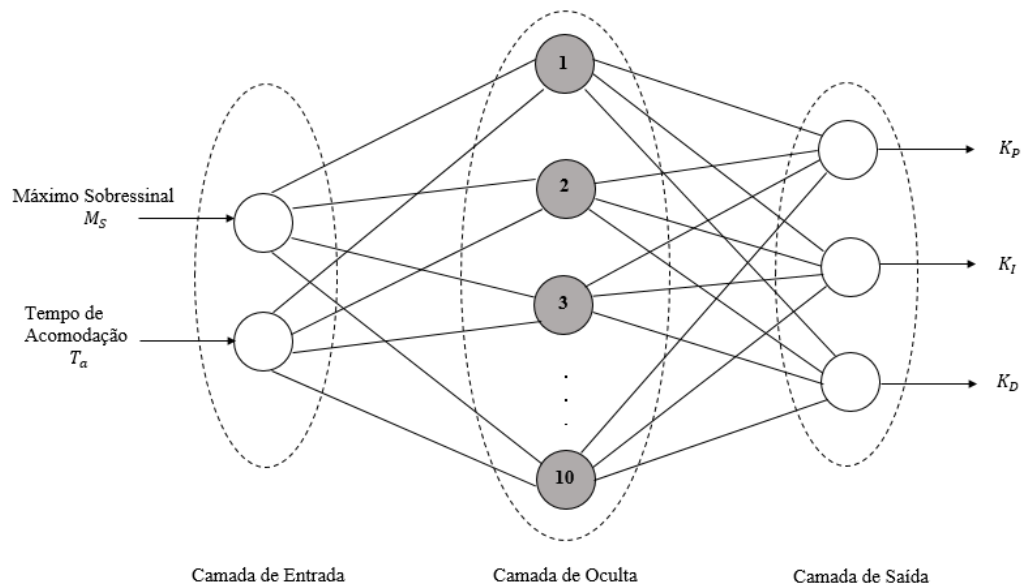
$$17 \leq K_p \leq 22; 0,1 \leq K_i \leq 0,5 \text{ e } 0,1 \leq K_d \leq 0,6$$

O passo adotado para K_p e K_i foi de 1 e para K_d um passo de 0,5. Também foram eliminados dados com mais de 2s para o tempo de acomodação, o que resultou numa amostra de 1647 dados.

Para a resolução do problema foi utilizado o paradigma de aprendizado supervisionado, a arquitetura de múltiplas camadas do tipo *feedforward*, além do treinamento usando os algoritmos de *backpropagation*. Foram testadas diferentes quantidades de camadas, algoritmos de treino (gradiente descendente, gradiente médio descente e *Levenberg-Marquardt*) e funções de ativação *tansig* e *logsig*.

Como o aumento de camadas não resultou em mudança significativas, assim como o aumento de neurônios na camada oculta, foi adotado uma rede 2-10-3, com função de ativação *logsing*, que gerou melhor resultado.

Figura 17- Estrutura RNA proposta

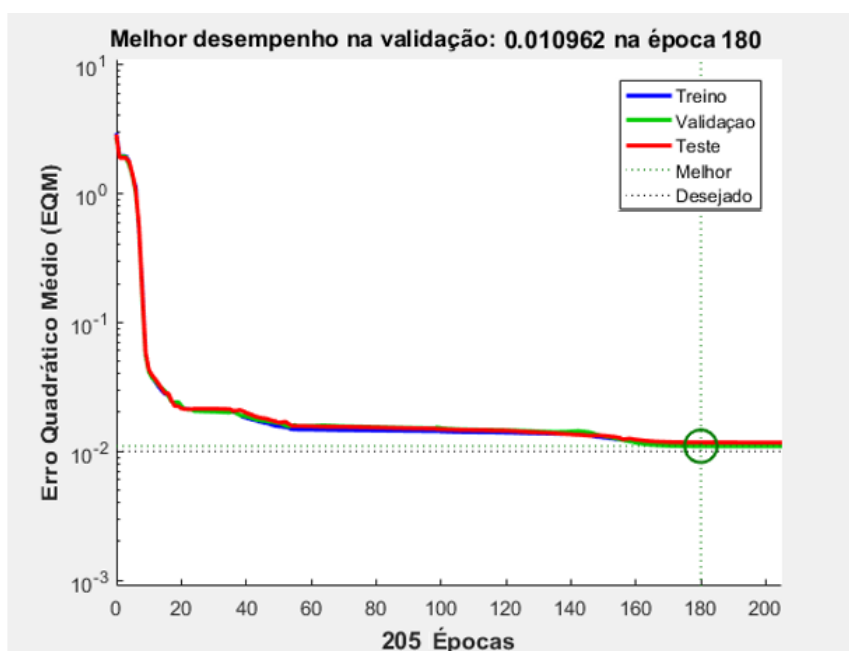


Fonte: Autoria própria

Para o treinamento de todas as redes foram usados o número máximo de 3000 épocas, erro desejado de 10^{-2} e máxima falha de 25.

Na Figura 18, tem-se o melhor desempenho, obtido com o algoritmo de *Levenberg-Marquardt*, função de ativação *logsing* e configuração 2-10-3. Como pode-se observar o erro final de validação foi de 0,010962 na época 180.

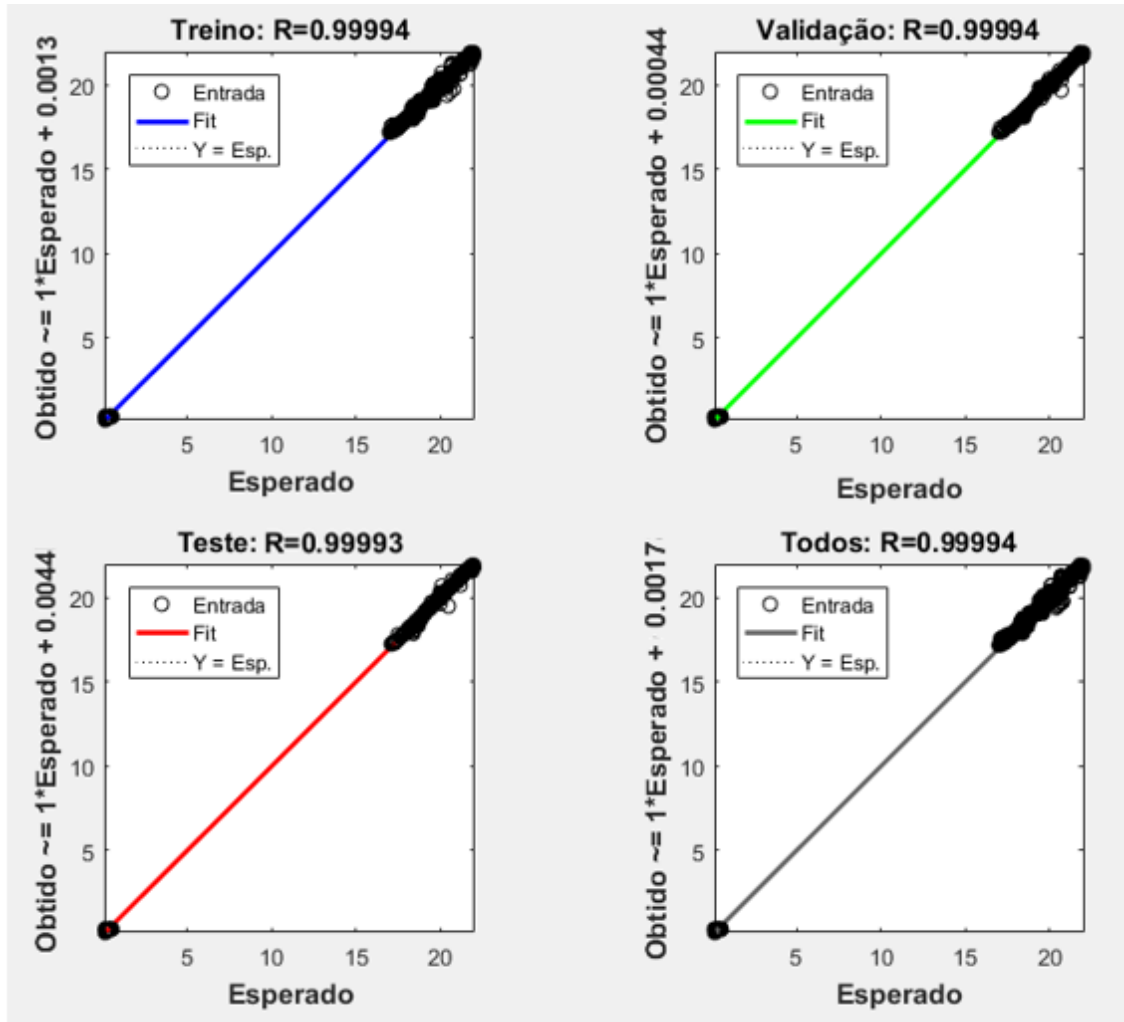
Figura 18 – Desempenho da rede neural durante o treinamento



Fonte: Autoria própria

Na Figura 19 pode-se observar as curvas de regressão, onde observa-se um desempenho de 0,99994 nas fases de treinamento, validação e teste.

Figura 19 – Curvas de regressão



Fonte: Autoria própria

A validação foi feita usando 5 amostras distintas das usadas para construção do banco de dados. Nessa validação foram fornecidos pares de valores de máximo sobressinal e tempo de acomodação e aplicados como entradas a rede, que sugeriu parâmetros para o controlador PID. Esses parâmetros foram simulados na planta com uma entrada de degrau unitário para que fosse encontrado o máximo sobressinal e tempo de acomodação que eles forneceram quando aplicados no controlador do sistema. O erro foi calculado considerando esses valores encontrados pela simulação e os valores esperados quando fornecidos a rede neural. Na Tabela 5 tem-se os resultados dessa validação.

Tabela 5 – Dados de validação da RNA

Resultados Esperados					Resultados Obtidos na RNA				
Kp	Ki	Kd	Ms (%)	Ta (s)	Kp	Ki	Kd	Ms (%)	Ta (s)
17,5	0,15	0,13	3,18	0,220	17,500	0,3094	0,1221	3,52	0,226
18,5	0,35	0,21	2,00	0,180	17,997	0,2766	0,2097	1,34	0,140
15,5	0,21	0,12	0,84	0,164	17,000	0,3492	0,2057	0,35	0,154
19,5	0,42	0,37	0,13	0,148	17,000	0,3878	0,2564	0,16	0,170
17,7	0,28	0,245	0,11	0,150	17,000	0,3846	0,2570	0,16	0,170
Erros									
Erro abs. de Ms (%)		Erro rel. de Ms (%)		Erro abs. de Ta (s)		Erro rel. de Ta (%)			
0,34		0,097		0,006		0,026			
0,66		0,492		0,040		0,280			
0,49		1,400		0,010		0,065			
0,03		0,187		0,022		0,129			
0,05		0,312		0,020		0,118			

Fonte: Autoria própria

Na Tabela 5, as primeiras colunas trazem informações sobre as amostras de validação. *Ms*, o máximo sobressinal esperado, e *Ts*, o tempo de acomodação esperado são os dados que foram fornecidos a rede treinada, enquanto *Kp*, *Ki* e *Kd* esperados são os parâmetros que geraram os valores de entrada. Nas colunas de resultados obtidos pela RNA, *Kp*, *Ki* e *Kd* correspondem a saída da rede. Esses valores sugeridos foram aplicados ao sistema e controlador PID e geraram *Ms* e *Ts*. Os erros foram calculados entre o *Ms* esperado e o obtido, assim como entre o *Ts* esperado e o obtido. *Erro abs. Ms* e *Erro rel. Ms* são os erros relativos e absolutos para o máximo sobressinal e *Erro abs. Ts* e *Erro rel. Ts* são os erros relativos e absolutos para o tempo de acomodação.

A Figura 20, ilustra o comparativo entre respostas ao degrau unitário para o controlador utilizando os parâmetros da terceira amostra da Tabela 5 (em vermelho) e a resposta obtida com parâmetros sugeridos pela rede (em azul).

Figura 20 - Comparativo entre respostas da amostra 3



Fonte: Autoria própria

Como é possível observar as respostas são semelhantes, mesmo para amostra 3 que apresentou o maior erro relativo ao M_s

4 CONSIDERAÇÕES FINAIS

Nesse trabalho foi apresentado um estudo de diferentes técnicas de sintonia dos parâmetros de um controlador PID para um motor de corrente contínua (motor CC), além de propor o treinamento de uma RNA capaz de sugerir valores para K_p , K_i e K_d que garantam um sobressinal e tempo de acomodação bem próximos do especificado pelo projetista como dados de entrada da rede.

Para tanto, utilizou-se do método clássico de sintonia de Ziegler-Nichols para encontrar um ponto de partida, do ajuste fino proposto por Ogata (2010) para refinar ainda mais esses valores e de técnicas de inteligência computacional já sugeridas em literaturas anteriores para otimizar o sistema, utilizando algoritmos genéticos, e para treinar a rede neural que serviria para ajustes mais específicos. O melhor resultado encontrado foi a otimização do sistema por meio do algoritmo genético. Além disso, a sintonia pela RNA se mostrou satisfatória, apresentando valores de erro relativamente pequenos, tanto para o treinamento quanto para a validação dos resultados.

Dessa forma, pode-se concluir que os resultados obtidos com os diferentes valores encontrados deixaram explícitas as diferenças no funcionamento do sistema diante do correto ajuste dos parâmetros do controlador PID, além de demonstrar a utilização de técnicas modernas aplicadas a esse problema clássico da indústria.

Para trabalhos futuros, sugere-se a implementação prática do sistema proposto para a comparação com os dados simulados. Além disso, pode-se utilizar um Algoritmo Genético Adaptativo (AGA) para a obtenção dos ganhos do controlador PID para a planta do motor de corrente contínua em estudo para a otimização em tempo real.

REFERÊNCIAS

ADHIKARI, N. P.; CHOUBEY, M.; SINGH, R. Dc Motor Control Using Ziegler Nichols and Genetic Algorithm Technique. **International Journal of Electrical, Electronics and Computer Engineering**, Jabalpur, v. 1, p. 33-36, 15 Abril 2012. ISSN 2277-2626. Department of Electrical Engineering, Gyan Ganga College of Technology.

CANIATO, L. C. **Modelagem e controle de nível e temperatura em sistema de armazenamento de água purificada para uso em empresa farmacêutica**. Escola de Engenharia de Mauá. São Caetano do Sul, p. 156. 2006.

CARDON, André; MÜLLER, Daniel Nehme; NAVAUX, Philippe. Introdução às redes neurais artificiais. **Instituto de Informática. Universidade Federal do Rio Grande do Sul**. Porto Alegre, 1994.

COSTA, J. C.; TAVARES, R.; ROSA, A. An experimental study on dynamic random variation of population size. **IEEE SMC'99 Conference on Systems, Man, and Cybernetics**, Tokyo, 1, 1999. 607-612. Digital Object Identifier: 10.1109/ICSMC.1999.814161.

DAGLI, C. H.; SCHIERHOLT, K. Evaluating the performance of the genetic neuro scheduler using constant as well as changing crossover and mutation rates. **Proceedings of the 21st international conference on Computers and industrial engineering**, 33, October 1997. 253-256. [http://dx.doi.org/10.1016/S0360-8352\(97\)00086-7](http://dx.doi.org/10.1016/S0360-8352(97)00086-7).

DORF, R. C.; BISHOP, R. H. **Sistemas de Controle Moderno**. 8ª. ed. Rio de Janeiro: LTC, 2001.

FITZGERALD, A. E.; KINGSLEY, C. J.; STEPHEN, D. U. **Máquinas Elétricas: Com introdução à eletrônica de potência**. 7ª. ed. Porto Alegre: Bookman, 2014.

GUIMARÃES, Ádller de Oliveira. **ALGORITMO GENÉTICO APLICADO NO CONTROLE DE POSIÇÃO DO ROTOR DE UM MOTOR DE CORRENTE CONTÍNUA COM REJEIÇÃO A DISTÚRBIOS POR AÇÃO FEEDFORWARD**. 2013. 72 f. Dissertação (Mestrado) - Curso de Engenharia Elétrica, Programa de Pós-graduação em Sistemas de Comunicação e Automação, Universidade Federal Rural do Semi-Árido, Mossoró, 2013.

GOLDSCHMIDT, R. R. **Uma Introdução à Inteligência Computacional: fundamentos, ferramentas e aplicações**. 1ª. ed. Rio de Janeiro: IST-Rio, 2010.

HAYKIN, S. **Redes Neurais**: Princípios e Práticas. 2º. ed. Porto Alegre: Bookman, 2001.

LINDEN, R. **Algoritmos Genéticos**: Uma Importante Ferramenta da Inteligência Computacional. 2ª. ed. Rio de Janeiro: Brasport, 2008.

MICHALEWICZ, Z. **Genetic Algorithms+Data Structures=Evolution Programs**. 3ª. ed. New York: Springer-Verlag, 1996.

NASCIMENTO, L. S. V. **Estudo da operação otimizada aplicada a um sistema de reservatórios destinado à geração de energia elétrica**. Escola de Engenharia de São Carlos, Universidade de São Paulo. São Paulo, p. 106. 2006.

OGATA, K. **Engenharia de Controle Moderno**. 3ª. ed. Rio de Janeiro: Livros Técnicos e Científicos Editora, 1998.

OGATA, K. **Engenharia de Controle Moderno**. 5ª. ed. São Paulo: Pearson Education, 2010.

RENER, G.; EKART, A. Genetic algorithms in computer aided design. **Computer aided Design**, 35, 2003. 709-726.

TENG, T. K.; SHIEH, J. S.; CHEN, C. S. Genetic algorithms applied in online autotuning PID parameters of a liquid-level control system. **Transactions of the Institute of Measurement and Control**, Taiwan, 25, Dezembro 2003. 433-450. <http://tim.sagepub.com/content/25/5/433>.

YALCINOZ, Y.; ALTUN, H. Environmentally constrained economic dispatch via a genetic algorithm with arithmetic crossover. **IEEE AFRICON Conference**, George, 2, October 2002. 923-928.