

FERRAMENTA DE DETECÇÃO DE PADRÕES DE PLANICIDADE UTILIZANDO REDES NEURAIS*

Arlei Fonseca Barcelos¹
Eduardo Sidney Dias²
Hugo Shokychi Toshimitsu³
Ramon Alves dos Santos⁴

Resumo

Uma das características de qualidade mais importantes das bobinas Full Hard é a planicidade. Essa pesquisa visa confirmar a hipótese de desenvolver e treinar uma rede neural capaz de identificar padrões de planicidade de bobinas laminadas no Laminador de Tiras a Frio#3 da CSN, propiciando o aumento do padrão de qualidade, inovação tecnológica e ganhos econômicos. A metodologia aplicada para alcançar os resultados almejados baseou-se em experimentos realizados com informações de bobinas reais, sendo utilizados softwares distintos para o desenvolvimento do projeto, validando os resultados obtidos através de análises de desempenho, com a correlação entre respostas conhecidas e as obtidas com a utilização da rede neural, assim como, a análise do erro médio quadrático das respostas obtidas com a aplicação, sendo simulados na etapa de treinamento da rede neural ou verificados experimentalmente com a execução da ferramenta desenvolvida.

Palavras-chave: Laminação a frio; Planicidade; Rede neural.

FLATNESS STANDARDS DETECTION TOOL USING NEURAL NETWORKS

Abstract

One of the most important quality features of a Full Hard coils is the flatness. This research aims to confirm the hypothesis to develop and train a neural network able to identify flatness standards of the coils rolled in CSN's Tandem Cold Mill#3, providing increased quality standards, technological innovation and economic gains. The methodology used to achieve the desired results was based on experiments with information of real coils, being used different softwares for the development of the project, validating the results through performance reviews, with the correlation between known results and those obtained with the use of the neural network, as well as the analysis of the mean square error of the results obtained with the implementation of the application, being simulated in the neural network training stage or verified experimentally with the execution of the tool developed.

Keywords: Cold rolling; Flatness; Neural Network.

¹ Engenheiro Eletrônico, Mestrado em Engenharia Elétrica/COPPE/UFRJ, Engenheiro Especialista de Manutenção, GRF/GGGL, CSN, Volta Redonda, Rio de Janeiro e Brasil.

² Engenheiro Eletrônico, MBA em Gestão Empresarial, Gerente do Laminador LTF3, GRF/GGGL, CSN, Volta Redonda, Rio de Janeiro e Brasil.

³ Engenheiro Metalúrgico, Especialista em Laminação, Engenheiro de Processos, GRF/GGGL, CSN, Volta Redonda, Rio de Janeiro e Brasil.

⁴ Engenheiro Eletrônico, Estagiário em Engenharia Eletrônica, GRF/GGGL, CSN, Volta Redonda, Rio de Janeiro e Brasil.

1 INTRODUÇÃO

O sistema nervoso humano é responsável pelas principais funções de controle do nosso organismo, sendo constituído por uma unidade básica chamada neurônio. O neurônio é responsável por diversas funções conhecidas, sendo a de maior destaque a cognição. A cognição é a capacidade do cérebro humano armazenar informação e formar o aprendizado.

Redes Neurais Artificiais, segundo Haykin (2001)[1], é uma técnica de Inteligência Artificial desenvolvida na década de 40 pelo matemático Walter Pitts e o neurofisiologista McCulloch. Eles objetivaram associar um neurônio biológico a um circuito eletrônico. Posteriormente, esse modelo teve readequação para um modelo computacional associado.

A tentativa de simular o sistema nervoso humano possibilitou atribuir às máquinas a capacidade de adquirir aprendizado e o reconhecimento de padrões, ou seja, a Rede Neural Artificial é um conceito matemático que trabalha na modelagem de um sistema real com base no conhecimento específico sobre um assunto em questão, visando o processamento de dados de forma semelhante ao do cérebro humano e disponibilizando à uma aplicação específica.

Para Braga et al. (2000)[2], o neurônio artificial proposto por Pitt e McCulloch é uma simplificação de um neurônio biológico, com sua descrição matemática resultando em um modelo de neurônio com “n” entradas, representando os dendritos, e com apenas uma saída.

Para simular o comportamento das sinapses de cada entrada do neurônio artificial, existem pesos acoplados cujo os valores podem ser inibitórios ou excitatórios, determinando em que grau o neurônio deve considerar o sinal de disparo naquela conexão.

Um neurônio biológico dispara quando a soma dos impulsos que ele recebe ultrapassa o seu limiar de excitação. O corpo do neurônio, por sua vez, é simulado por um mecanismo simples que faz a soma dos valores recebidos pelo neurônio (soma ponderada) e decide se o neurônio deve ou não disparar seu sinal, comparando a soma obtida ao limiar do neurônio.

A Figura 1 mostra um modelo de neurônio artificial, onde segundo Haykin (2001)[1], é possível identificar os elementos básicos que estão descritos abaixo.

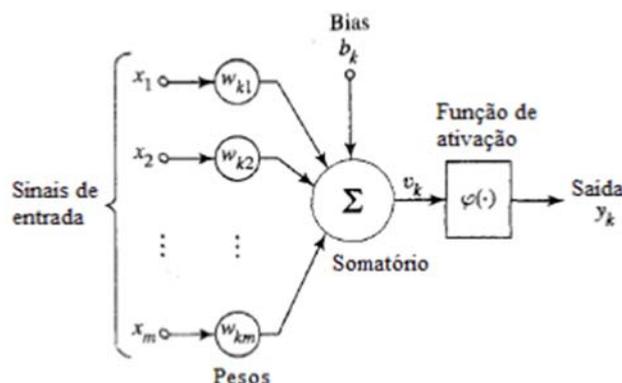


Figura 1. Modelo de um neurônio artificial-Fonte: Haykin, 2001.

No modelo da Figura 1, é representado um conjunto de elos de conexão (ou conjunto de sinapses), cada um caracterizado por um peso ou força própria. Especificamente, um sinal x_m na entrada da sinapse conectada ao neurônio k é multiplicado pelo peso sináptico w_{km} .

A representação do sinal de somatório realiza a adição dos sinais de entrada ponderados pelas respectivas sinapses do neurônio, formando uma espécie de combinador linear.

Uma função de ativação é utilizada para restringir a amplitude da saída de um neurônio. A função de ativação é também referida como função restritiva, já que, restringe (limita) o intervalo permissível de amplitude do sinal de saída a um valor finito.

O bias, representado por b_k , tem o efeito de aumentar ou diminuir a entrada líquida da função de ativação, dependendo se este assume, valores positivos ou negativos, respectivamente.

Adentrando ao meio siderúrgico, de acordo com Silva (2008)[3], a qualidade da planicidade de uma bobina é um defeito claramente percebido e que vem sendo demandada em tolerância cada vez mais restrita pelo mercado. Sendo caracterizadas estas anormalidades principalmente pela presença de ondulações resultantes da acomodação de regiões mais alongadas, que costumam se concentrar em regiões do centro ou nas bordas das bobinas. A planicidade pode ser determinada pela diferença de alongamento ao longo da largura através de uma unidade adimensional chamada I-Unit, ao qual está relacionado com a altura e o comprimento das ondulações.

Existem várias causas que podem levar ao surgimento do defeito de planicidade na tira. Modella et al. (2013)[4] destaca as seguintes causas: ajuste inadequada da abertura entre os cilindros; flexão dos cilindros de laminação; distribuição inadequada do sistema de refrigeração.

Dessa forma o projeto desenvolvido busca aplicar o referido método de inteligência artificial a favor da automatização do processo de classificação de padrões de planicidade dos materiais produzidos no Laminador de Tiras a Frio número 3 da CSN apoiando a tomada de decisão, aplicando a modelagem adequada de forma que este sistema absorva através de um treinamento, padrões adequados e inadequados de planicidade. Podendo após esta etapa, inferir uma resposta quando estimulado com padrões semelhantes. Buscando, além da automatização do processo, o aumento no padrão de qualidade do material produzido, podendo inserir através das entradas da rede neural desenvolvida mais parâmetros de planicidade os quais não eram avaliados, ou ainda, aqueles os quais eram difíceis ou suscetíveis a erros na sua avaliação pelo homem.

2 MATERIAIS E MÉTODOS

O sistema de detecção de planicidade, implantado na laminação a frio da siderúrgica analisada, realiza as medições das tensões nas bobinas pelo método de contato através de um rolo medidor (rolo *shapemeter*), onde esse efetua as medições dessas tensões em seções transversais do material. Essas seções estão dispostas em 32 zonas de medições longitudinais, constando em cada seção 4 células de carga dispostas nos quatro quadrantes do rolo de medição. Cada seção longitudinal é espaçada em 52mm e a coleta de informação de tensões das seções transversais são realizadas a cada 50m ao longo da bobina.

Com a utilização deste sistema para verificar a conformidade de um material torna-se necessária a realização das seguintes etapas:

a) Um operador realiza a análise visual do produto durante todo o processo de produção;

- b) Ao detectar algum defeito visível, é reportado qual bobina que apresentou desvio de padrão de qualidade;
- c) Com as informação da bobina identificada visualmente pelo operador, um técnico especialista de processo analisa a imagem tridimensional da bobina (carta) e retém ou não o material.

O Sistema apresentava oportunidade de ganhos, pois algumas bobinas inadequadas a produção, não detectadas visualmente pela operação, eram entregues aos clientes internos/externos, gerando reclamações e não conformidade.

2.1 Classificação de planicidade

As informações de tensões das bobinas, coletadas pelos sensores de campo, são processadas por um controlador proprietário dedicado, disponibilizando-as em telas de supervisor e em um banco de dados em MS SQL Server, sendo estes dados utilizados em software de análise de sinal. O Logger Viewer é software que exibe um gráfico em três dimensões, que descreve a distribuição das tensões nas 32 secções longitudinais ao longo da bobina interpolando todos os dados de forma gráfica. Na Figura 2 é apresentada a imagem gráfica gerada pelo Logger Viewer.

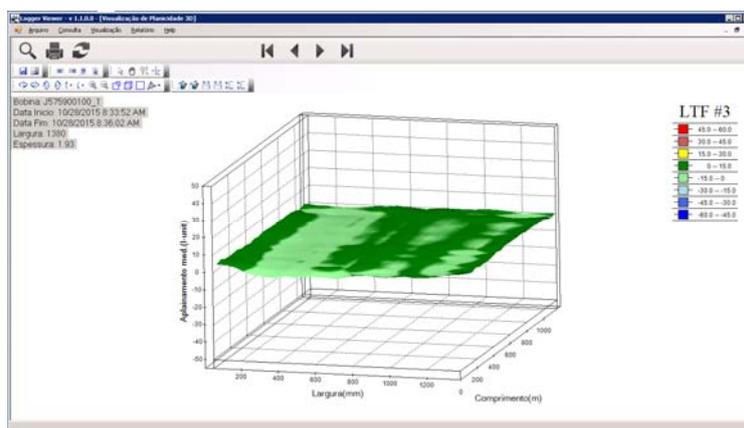


Figura 2. Imagem interpolada de uma Bobina-Fonte: Logger Viwer.

O padrão de classificação de planicidade de bobina é definida através de uma norma interna da siderúrgica em estudo, sendo descrita através de um procedimento operacional que detalha as atividades conforme demonstrado na Tabela 1.

Tabela 1. Procedimento operacional para classificação de planicidade

Atividade	Descrição
Avaliar aplainamento na tela da IHM do Púlpito na Figura 3	O operador de púlpito deverá acompanhar o comportamento do aplainamento das bobinas através do monitor da IHM ao longo do processamento da bobina, verificando o formato da barra
Informar ao inspetor de qualidade em caso de bobinas com suspeita de Falha de Planicidade	Caso seja observado através das barras um perfil irregular de aplainamento o operador do púlpito deverá avisar o inspetor, que deverá verificar o aplainamento no Logger Viewer. Se a bobina apresentar mais de 30% acima de 15UI, o inspetor deverá segregar a mesma.

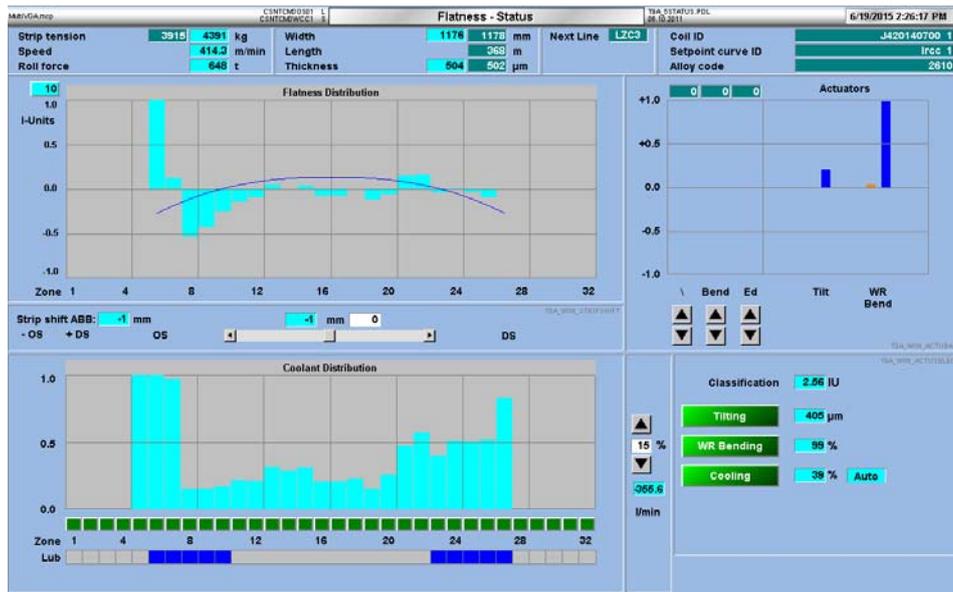


Figura 3. Tela do Supervisor do monitor de aplainamento - IHM.

2.2 Validação da aplicabilidade da Rede Neural

Para validar a aplicabilidade da rede neural neste projeto, foi desenvolvido um banco de dados de treinamento para a RNA. As informações desse banco de dados foram coletadas de um servidor SQL Server da siderúrgica em estudo, utilizando o Microsoft Query associados ao Excel. O algoritmo desenvolvido realiza as consultas das informações das bobinas e realiza uma pré-estruturação dos dados antes de disponibilizá-lo em uma tabela do MS Excel.

Inicialmente o critério de avaliação de qualidade da bobina foi baseado na norma de planicidade interna, onde em linhas gerais, classifica uma bobina como ruim quando majoritariamente há ultrapassagem de algum ponto da bobina acima de 15 I-units. Foram também inseridas no banco de dados as bobinas com assimetria e bobinas que no decorrer da operação o inspetor de qualidade fossem retidas.

No decorrer do desenvolvimento do projeto, o critério de classificação foi alterado, se tornando mais flexível. Essa alteração foi necessária devido a primeira versão da rede neural ter apresentado resultados positivos, porém extremamente críticos. Dessa forma, com objetivo de flexibilizar a rede neural, o critério para bobinas ruins foi alterado para: bobinas que possuíssem grandes seções as quais ultrapassassem 15 I-units ou mesmos grandes picos acima de 15 I-units; bobinas com simetria irregular, mesmo que não ultrapassem os 15 I-units, mas que apresentem assimetria extremamente alta.

Com o algoritmo de pesquisa pronto, foi desenvolvida uma tabela base no MS Excel, onde ficam gravados os dados coletados pelo Microsoft Query, contendo as informações de 32 zonas de medições posicionadas transversalmente na bobina de aço. Além das medições transversais, a base de dados da empresa possui 4 campos referente aos índices de qualidade das bobinas, denominados q1, q2, q3 e q4. Esses itens descrevem a porcentagem da bobina que são menores que 10i-units, 15i-units, 20i-units e acima de 20 I-units respectivamente. Essas medidas são coletadas na bobina a cada 50m, ou seja, para uma bobina que tenha comprimento de 6km, haverá 120 registros, ou linhas, para a referida bobina.

Para efetuar o treinamento da rede neural, fez-se necessário sintetizar cada matriz a qual representava apenas uma bobina em uma única linha, sendo necessário para

tal, a utilização de métodos estatísticos. Foram testados vários métodos estatísticos (média aritmética, desvio padrão, média quadrática, média aritmética somada ao valor da média quadrática e média aritmética dos valores cúbicos), sendo o desvio padrão que apresentou um desempenho superior na validação e treinamento da rede neural. Para otimizar os resultados, os fatores de dimensão da tira foram corrigidos pelos fatores apresentados na Tabela 2.

Tabela 2. Fator de ponderação nas dimensões da tira

Dimensão	Fator
Espessura	10
Largura	1/100
Comprimento	1/1000

Com os valores das dimensões, multiplicado pelo seu respectivo fator de ponderação, evitou-se possíveis distorções provocadas pela diferença de escala das medições em relação ao comprimento, largura e espessura, ou seja, todas as entradas resultaram com valores com a mesma ordem de grandeza.

Com os resultados obtidos e disponibilizado em uma matriz no Excel, a rede foi validada no software NeuroSolutions. Neste software foi realizado a randomização das amostras, ou seja, a randomização das linhas do banco de dados, onde cada linha sintetizava uma bobina. A randomização foi necessária para que a rede neural não “decore” uma sequência de bobinas, comprometendo o aprendizado.

Foram criadas inicialmente redes neurais com 4 neurônios na camada escondida. Realizando diversas simulações e verificações, variando principalmente o tipo de função de ativação da camada escondida e camada de saída, foi constatado que o melhor desempenho foi obtido com redes utilizando funções de ativação do tipo tangente hiperbólica na camada escondida e *sigmoidal* na camada de saída.

Quanto a arquitetura de rede neural utilizada, as redes do tipo *cascade feedforward* apresentaram melhor resultado em comparação com as funções *feedforward multi layer perceptron*.

Com a rede neural criada e treinada, foi validado seu desempenho. Para essa validação foi realizado antes da etapa de treinamento a divisão do banco de dados em amostras de treinamento, validação e teste. As amostras de validação foram testadas pela rede neural e comparados seus resultados, já que, para estas amostras a rede conhecia o valor de saída. Para as amostras de teste, a rede não conhecia o valor de sua saída, ou seja, se a bobina é boa (saída igual a 1) ou se é ruim (saída igual a 0), estimando uma resposta avaliando assim o comportamento da rede neural real.

2.3 Desenvolvimento da rede no Matlab

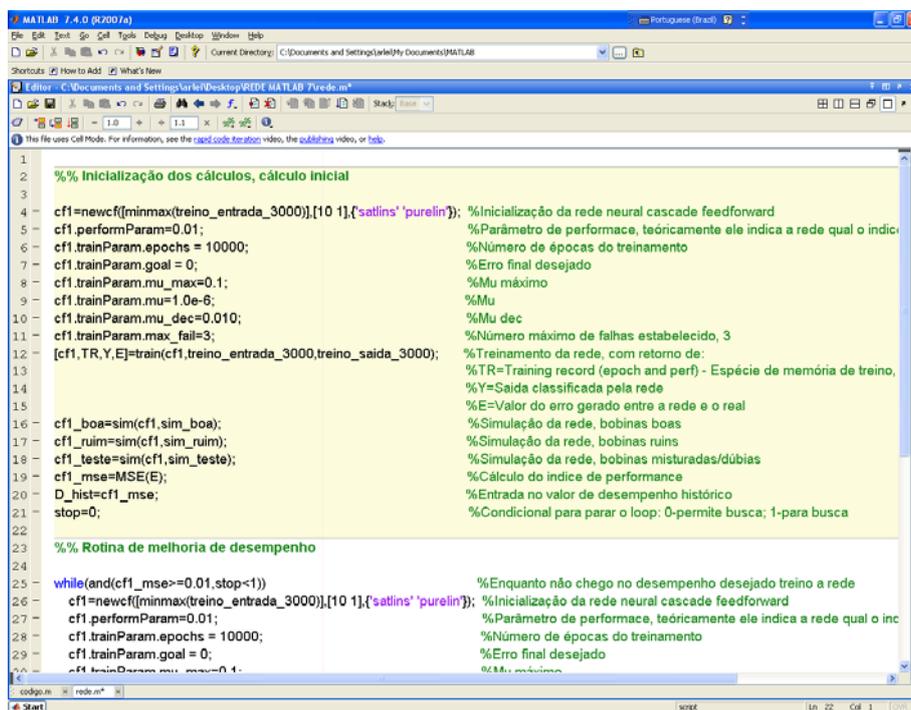
Com o projeto validado iniciou-se o desenvolvimento da rede neural no Matlab, utilizando inicialmente a ferramenta gráfica disponível no software, o NNtools. Dividiu-se o banco de dados em 3 matrizes, com o auxílio do próprio Excel. Cada matriz representa as amostras de treinamento, amostras de validação e amostras de teste. Cada matriz foi dividida em duas tabelas: tabela de entrada e tabela de saída. Como exemplo, as bobinas selecionadas para amostras de treino possuem duas tabelas, uma com as entradas da rede neural e uma com a saída da rede neural, ou seja, os dados tratados pelos respectivos métodos estatísticos e a saída para cada bobina em boa “1” ou ruim “0”.

Com todos os dados inseridos no NNtools, o Matlab gerou a rede neural, a partir dos parâmetros inseridos, como arquitetura de rede neural utilizada, número de camadas escondidas, quantidade de neurônios por camada, tipo de função ativação, tipo de treinamento, tipo de função de adaptação, limites máximo e mínimo com base nos dados de entrada principais, entre outros.

Com a rede neural criada, é acionada a função “Train” para executar o treino, a validação e o teste da rede. Ao término do treinamento da rede neural, um gráfico é gerado, apresentando o índice da média quadrática do erro. Outro parâmetro importante é o número de épocas (quantidade de iterações para treinamento) de treinamento, sendo desejável que este número seja um valor mais próximo do configurado na aba “Training Parameters”. Esse processo foi realizado diversas vezes até alcançar um resultado desejado. Em cada execução, a rede e os dados eram alterados no decorrer do experimento.

A utilização NNtool, apesar de simples e de fácil manipulação, demonstrou ser uma rotina extremamente demorada, pouco automatizada e apresentava limitações quanto aos tipos de função de ativação disponíveis. Diante disso, foi desenvolvido um algoritmo de busca automatizado no Matlab, de forma a facilitar e maximizar o tempo de treinamento da rede neural.

Foi desenvolvido uma nova forma de treinamento, verificação e busca pela rede neural, desenvolvendo este através de linhas de comando no Matlab. Esse novo programa é capaz, de forma automática, buscar a rede com melhor coeficiente de desempenho, ou seja, menor erro médio quadrático entre as saídas da rede neural e o estado da rede. Este algoritmo é exemplificado na Figura 4.



```

1  %% Inicialização dos cálculos, cálculo inicial
2
3
4  cf1=newcfc([minmax(treino_entrada_3000)],[10 1],{'satlins' 'purelin'}); %Inicialização da rede neural cascade feedforward
5  cf1.performParam=0.01; %Parâmetro de performance, teoricamente ele indica a rede qual o indic
6  cf1.trainParam.epochs = 10000; %Número de épocas do treinamento
7  cf1.trainParam.goal = 0; %Erro final desejado
8  cf1.trainParam.mu_max=0.1; %Mu máximo
9  cf1.trainParam.mu = 1.0e-6; %Mu
10 cf1.trainParam.mu_dec=0.010; %Mu dec
11 cf1.trainParam.max_fail=3; %Número máximo de falhas estabelecido, 3
12 [cf1,TR,Y,E]=train(cf1,treino_entrada_3000,treino_saida_3000); %Treinamento da rede, com retorno de:
13 %TR=Training record (epoch and perf) - Espécie de memória de treino,
14 %Y=Saída classificada pela rede
15 %E=Valor do erro gerado entre a rede e o real
16 cf1_boa=sim(cf1,sim_boa); %Simulação da rede, bobinas boas
17 cf1_ruim=sim(cf1,sim_ruim); %Simulação da rede, bobinas ruins
18 cf1_teste=sim(cf1,sim_teste); %Simulação da rede, bobinas misturadas/dúbias
19 cf1_mse=MSE(E); %Cálculo do índice de performance
20 D_hist=cf1_mse; %Entrada no valor de desempenho histórico
21 stop=0; %Condicional para parar o loop: 0-permite busca; 1-para busca
22
23 %% Rotina de melhoria de desempenho
24
25 while(and(cf1_mse>=0.01,stop<1)) %Enquanto não chego no desempenho desejado treino a rede
26 cf1=newcfc([minmax(treino_entrada_3000)],[10 1],{'satlins' 'purelin'}); %Inicialização da rede neural cascade feedforward
27 cf1.performParam=0.01; %Parâmetro de performance, teoricamente ele indica a rede qual o inc
28 cf1.trainParam.epochs = 10000; %Número de épocas do treinamento
29 cf1.trainParam.goal = 0; %Erro final desejado
30 cf1.trainParam.mu_max=0.1; %Mu máximo
31 cf1.trainParam.mu = 1.0e-6; %Mu
32 cf1.trainParam.mu_dec=0.010; %Mu dec
33 cf1.trainParam.max_fail=3; %Número máximo de falhas estabelecido, 3
34 [cf1,TR,Y,E]=train(cf1,treino_entrada_3000,treino_saida_3000); %Treinamento da rede, com retorno de:
35 %TR=Training record (epoch and perf) - Espécie de memória de treino,
36 %Y=Saída classificada pela rede
37 %E=Valor do erro gerado entre a rede e o real
38 cf1_boa=sim(cf1,sim_boa); %Simulação da rede, bobinas boas
39 cf1_ruim=sim(cf1,sim_ruim); %Simulação da rede, bobinas ruins
40 cf1_teste=sim(cf1,sim_teste); %Simulação da rede, bobinas misturadas/dúbias
41 cf1_mse=MSE(E); %Cálculo do índice de performance
42 D_hist=[D_hist;cf1_mse]; %Entrada no valor de desempenho histórico
43 stop=1; %Condicional para parar o loop: 0-permite busca; 1-para busca
44
45

```

Figura 4. Algoritmo Rede Neural – Matlab.

Com o desenvolvimento deste algoritmo base foi possível a busca automática pela melhor rede, podendo ter liberdade quanto a parametrização da rede neural, além de maximização do desenvolvimento do projeto. A busca é cessada quando o valor do erro médio quadrático está na faixa de valores parametrizados, quando o número de tentativas era superior a 10 ou a qualquer momento sendo necessário o usuário

digitar as teclas “CTRL+C”, parando a execução do algoritmo. Encontrado o desempenho desejado para a rede neural, extraiu-se os parâmetros como pesos e bias da rede neural dispostos do Matlab exportando-os para o Excel, dando seguimento ao desenvolvimento do projeto.

2.4 Desenvolvimento da rede no Excel utilizando VBA

O conceito fundamental da rede neural aplicada neste projeto baseia-se em somatórios e multiplicações de matrizes. Objetivando uma facilidade na sua implementação prática e com os conceitos anteriormente citados consolidados implementou-se a aplicação no software no MS Excel, associando a este o suplemento MS Query o qual tem um algoritmo em SQL o qual é responsável por buscar no Servidor SQL Server, as bobinas produzidas no laminador nas últimas 24 horas.

A aplicação desenvolvida faz o tratamento matemático das zonas de medição de cada bobina através do método estatístico de desvio padrão, gerando uma tabela com todos os desvios. Com essa informação, o algoritmo varre a matriz bobina a bobina e insere essas informações na entrada da rede neural implementada no MS Excel, processando estas informações e gerando uma saída identificando a bobina como “boa” ou “ruim”, de forma quantitativa, sendo interpretada e apresentada ao usuário de forma qualitativa nos estados do aplainamento citados. Esta apresentação qualitativa dos dados dá-se através de um relatório diário com as bobinas classificadas nas últimas 24 horas apresentando seu item, data de processamento, estado quantitativo e qualitativo classificado pela rede, bem como apresentando ainda o próximo equipamento o qual irá processar a bobina avaliada podendo haver linhas onde o material deve receber um tratamento mais crítico ou flexível. Todo esse procedimento foi desenvolvido utilizando algoritmos customizados desenvolvidos na linguagem Visual Basic sobre a plataforma MS Excel. Na Figura 5, é apresentado o relatório gerado pela aplicação.

CSN		Relatório planicidade		Data Geração:	09/11/2015 09:12
				Período:	08/11/2015 11:08
CD_COIL	REDE	ESTADO	DT_START	DT_END	NUM_GRUPO_EQUIP
J58880300_1	0,99995813	boa	09/11/2015 11:08	09/11/2015 11:15	OUTROS
Relatório Final					
CD_COIL	REDE	ESTADO	DT_START	DT_END	NUM_GRUPO_EQUIP
J607890600_1	0,99994803	boa	09/11/2015 09:10	09/11/2015 09:12	FRCX
J607890500_1	0,99997478	boa	09/11/2015 09:06	09/11/2015 09:09	FRCX
J607910600_1	0,99991276	boa	09/11/2015 09:04	09/11/2015 09:06	FRCX
J607910500_1	0,99996306	boa	09/11/2015 09:00	09/11/2015 09:02	FRCX
J607910400_1	0,99994745	boa	09/11/2015 08:57	09/11/2015 08:59	FRCX
J607910300_1	0,99996359	boa	09/11/2015 08:54	09/11/2015 08:56	FRCX
J607910200_1	0,99996666	boa	09/11/2015 08:51	09/11/2015 08:53	FRCX
J607910100_1	0,99994521	boa	09/11/2015 08:47	09/11/2015 08:50	FRCX
J607890200_1	0,99998541	boa	09/11/2015 08:44	09/11/2015 08:46	FRCX
J607890100_1	0,99996458	boa	09/11/2015 08:39	09/11/2015 08:42	FRCX
J612730100_1	0,99995874	boa	09/11/2015 08:31	09/11/2015 08:33	FRCX
J592340100_1	0,99998027	boa	09/11/2015 07:14	09/11/2015 07:16	LZC2
J574830100_1	0,99976715	boa	09/11/2015 07:09	09/11/2015 07:12	LZC2
J587780900_1	0,99999862	boa	09/11/2015 07:06	09/11/2015 07:08	LZC2
J587780800_1	0,99999639	boa	09/11/2015 07:03	09/11/2015 07:05	LZC2
J587780600_1	0,99999956	boa	09/11/2015 07:01	09/11/2015 07:02	LZC2
J587780400_1	0,99995252	boa	09/11/2015 06:58	09/11/2015 07:00	LZC2
J587780300_1	0,99996942	boa	09/11/2015 06:54	09/11/2015 06:56	LZC2
J587780200_1	0,99999576	boa	09/11/2015 06:50	09/11/2015 06:54	LZC2
J587780100_1	0,99997353	boa	09/11/2015 06:48	09/11/2015 06:50	LZC2
J578300200_1	0,99999281	boa	09/11/2015 06:43	09/11/2015 06:47	EMBALAGEM
J578300100_1	0,99991607	boa	09/11/2015 06:39	09/11/2015 06:43	EMBALAGEM
J573190100_1	0,99999364	boa	09/11/2015 06:35	09/11/2015 06:38	LZC2
J582790100_1	0,99998176	boa	09/11/2015 06:32	09/11/2015 06:34	EMBALAGEM
J584080200_1	0,99999612	boa	09/11/2015 06:27	09/11/2015 06:31	EMBALAGEM
J584080300_1	0,99994411	boa	09/11/2015 06:22	09/11/2015 06:27	EMBALAGEM

Figura 5. Relatório das bobinas boas e ruins gerado pela aplicação.

3 RESULTADOS E DISCUSSÃO

Através dos softwares NeuroSolutions e do Matlab foi possível validar a exequibilidade desse projeto e obter a rede neural que melhor se adequa a detecção de planicidade. Com essas informações, foi desenvolvida uma ferramenta em Excel capaz de acessar automaticamente a base de dados da siderúrgica, coletar informações das bobinas registradas no Logger Viewer e gerar um relatório das bobinas que são classificadas como Boas e Ruins pela rede neural.

Com a ferramenta obtém-se como resultados potenciais aumento da produtividade da mão de obra, aumento na qualidade do material produzido com a redução de desvios de qualidade e paradas emergenciais em outras linhas de produção devido a aplainamento e desnivelamento do material entregue, redução da influência humana na classificação das bobinas produzidas no laminador, possibilidade de se ter um controle histórico do aplainamento com a ferramenta de apoio desenvolvida e todos estes benefícios a custo zero, caracterizando um projeto ZIAR – Zero Investimento e Alto Retorno. Traduzindo estes benefícios em números segue na Tabela 3 uma estimativa dos ganhos potenciais avaliados dos meses de janeiro à setembro, no ano de 2015.

Tabela 3. Ganhos potenciais estimados para áreas clientes

Áreas	Ganhos		Produção	Relação de ganho
	Horas (h)	Produção (t)	Acum.Set/2015 (t)	
LZC#1	39,88	867	99500	0,87%
LZC#3	26,58	651	148500	0,44%
LRCC	1,43	66	294000	0,02%
Total	68	1584	542000	0,29%

Com a Tabela 3 acima percebe-se os ganhos possíveis os quais poderiam ter sido aproveitados no ano de 2015, gastos com retrabalho e paradas ocasionadas por defeitos de planicidade entregues a clientes internos e externos, podendo estes terem sido evitados com a utilização da ferramenta, transformando-se em ganhos para a empresa. Observando que com a utilização da ferramenta, numa linha de zincagem esta poderia impactar se considerássemos o lucro acumulado em 200 milhões apenas está área geraria um ganho de 1,74 milhões a custo zero.

4 CONCLUSÃO

A técnica de reconhecimento de padrão de planicidade utilizando redes neurais apresentou ser viável a sua implementação. Atingindo seu objetivo inicial, com o viés de complementar o processo atual, gerando de forma tempestiva os resultados de planicidade das bobinas produzidas no período de 24h.

Como melhorias futuras, sugere o desenvolvimento de uma ferramenta que integre a ferramenta desenvolvida e o software gráfico Logger Viewer, disponível na empresa e a automatização do treinamento da rede neural, podendo esta ser feita com a máxima viabilidade e facilidade. Podendo ainda a implementação realizar a classificação dos defeitos por zonas, ou seja, classificar a bobina em boa ou ruim e informar a localização do defeito, diminuindo os desperdícios com descarte.

REFERÊNCIAS

- 1 Haykin S. Redes Neurais - Princípios E Práticas. Porto Alegre: Bookman; 2001.
- 2 Braga AP, Carvalho APLF, Ludemir TB. Redes Neurais Artificiais - Teoria e Aplicações. 2ª Edição. Rio de Janeiro: Editora LTC; 2000.
- 3 Silva CN, Araújo FGS, Fagundes J Jr, Cota AB. Efeito da flexão dos cilindros na laminação de encruamento sobre a planicidade de tiras de aço. *Matéria (Rio J.)*, [s.l.], v. 13, n. 2, p.412-417, 2008. FapUNIFESP (SciELO). DOI: 10.1590/s1517-70762008000200021.
- 4 Molleda J, Usametinga R, García D. On-Line Flatness Measurement in the Steelmaking Industry. *Sensors*, [s.l.], v. 13, n. 8, p.10245-10272, ago. 2013. MDPI AG. DOI: 10.3390/s130810245.