

Diagnóstico de Isoladores Poliméricos empregando Ruído Ultrassônico: uma Análise de Sensibilidade de Redes Neurais Artificiais

¹Marcus Tullius Barros Florentino, M. Sc. ²Edson Guedes da Costa, D. Sc. ²Tarso Vilela Ferreira, D. Sc. ²André Dantas Germano, PhD. Lenilson Andrade Barbosa.

¹Programa de Pós-Graduação em Engenharia Elétrica (PPGEE)

²Departamento de Engenharia Elétrica (DEE)

Universidade Federal de Campina Grande (UFCG)

58429-900, Campina Grande – PB – Brasil

edson@dee.ufcg.edu.br

Resumo— Durante o monitoramento de isoladores, a interpretação dos dados e a tomada de decisão realizadas exclusivamente por humanos podem ser questionáveis. Assim, percebe-se a importância do uso de ferramentas adequadas para o apoio à decisão. Este trabalho apresenta a análise de diferentes algoritmos de redes neurais artificiais utilizados na classificação de padrões acústicos visando a otimização de uma ferramenta de apoio. Para tanto, isoladores poliméricos foram submetidos a alta tensão, em laboratório, e medições acústicas das emissões de descargas nos isoladores foram registradas. Para melhor adequação dos parâmetros de entrada das redes, utilizou-se uma técnica de extração de atributos baseada nas informações espectrais de energia emitida. Os vetores de energia obtidos foram separados segundo as classes de degradação dos isoladores e permitiram o treinamento das redes neurais para classificação dos padrões. As redes baseadas no algoritmo Resilient Propagation e no algoritmo Levenberg-Marquardt obtiveram os melhores desempenhos, acima de 80% de acerto.

Palavras-chave—Análise de sensibilidade; Isoladores poliméricos; Redes neurais artificiais.

I. INTRODUÇÃO

Durante a operação de um sistema elétrico, podem ocorrer interrupções indesejadas que causam diversos prejuízos às empresas do setor elétrico e aos consumidores. Dentre as diversas causas das interrupções, destacam-se as que se relacionam com as falhas em isoladores poliméricos nas linhas de transmissão.

A capacidade de antever a falha, ou seja, a identificação e diagnóstico de um defeito pode ser o diferencial entre a operação continuada e o desligamento de uma linha de transmissão. É necessário, portanto, que se utilizem critérios de decisão e ferramentas que auxiliem na escolha do melhor momento para execução de intervenções, sejam estas de urgência ou programadas.

As inspeções periódicas são comumente utilizadas como técnicas para mitigação de falhas nas linhas de transmissão. Algumas técnicas de inspeção são baseadas nas ondas

acústicas e eletromagnéticas emitidas pelos isoladores em operação. Várias pesquisas comprovam a correlação entre as condições operativas dos isoladores e a emissão de ondas eletromagnéticas e/ou acústicas [1]-[5].

As inspeções acústicas permitem mensurar os níveis dos ruídos acústicos provocados pela presença de descargas elétricas na região dos isoladores. As inspeções são bastante favoráveis no monitoramento de isoladores por serem não invasivas, tendo em vista que as medições são realizadas à distância. Entretanto, a presença de ruídos externos durante as medições e o diagnóstico realizado pela interpretação de um indivíduo podem se tornar um entrave à eficiência do método [2].

Para que se minimizem as incertezas no diagnóstico do estado operacional dos isoladores, as inspeções acústicas devem ser associadas a algoritmos computacionais que auxiliem na tomada de decisão. As redes neurais artificiais (RNA) são uma poderosa ferramenta para realizar o processamento de informações [6].

Assim, neste trabalho, procurou-se unir a inspeção de isoladores poliméricos (IP) por ruídos ultrassônicos ao emprego das RNA, a fim de se embasar com critérios robustos à tomada de decisão. Mais que isso, buscou-se a identificação de um algoritmo de RNA que seja mais adequado a esta tarefa.

Para tanto, sinais dos ruídos ultrassônicos foram processados em computador para extração de atributos, que servem como base de dados para classificação de padrões por meio de arquiteturas de RNA dedicadas. As características das RNA que melhor se adequam à classificação de parâmetros de sinais acústicos foram, portanto, analisadas. Buscou-se, dessa maneira, eliminar algoritmos que apresentem menor eficiência na classificação dos sinais com relação ao grau de separação entre as classes de degradação dos isoladores.

As melhores arquiteturas de RNA servem, por sua vez, como referência para estudos futuros, com diferentes bases de dados, não somente relacionadas a sinais de natureza acústica. Atributos de imagem, por exemplo, também requerem separação e classificação em diferentes classes para melhor estudo de suas características [7].

II. REDES NEURAIS ARTIFICIAIS

As redes neurais artificiais são modelos computacionais adaptativos inspirados nas características de processamento do neurônio biológico [6]. O processamento das redes se dá através de regimes de aprendizado, em que se utilizam pesos sinápticos para armazenamento das informações numéricas [8]. As informações representam o conhecimento adquirido pela rede.

O treinamento da rede neural artificial é a etapa em que se definem as características de aprendizado da rede. Os parâmetros das camadas de entrada, ocultas e de saída são definidos no treinamento. Esses parâmetros representam a quantidade de neurônios utilizada em cada camada, os valores iniciais dos pesos sinápticos, o algoritmo de treinamento e a função de ativação. Assim, eles são definidos juntamente com o tipo de arquitetura que está sendo utilizado pela rede.

O projeto de uma rede neural artificial e, portanto, sua arquitetura depende primeiramente da aplicação a que se destina. Além disso, a natureza dos dados de entrada irá indicar o tamanho da rede, segundo a quantidade de parâmetros inseridos. A escolha de uma arquitetura adequada para a resolução de determinado problema requer uma análise conjunta dos parâmetros da rede. A experiência do projetista e a quantidade de tentativas, com diferentes configurações, pode facilitar a construção de uma rede plenamente capaz de resolver um problema específico [8].

A. Unidade de Processamento

A unidade básica de processamento das redes neurais artificiais é o neurônio artificial, cujo modelo foi proposto inicialmente por [9]. Uma generalização desse modelo é mostrada na Figura 1.

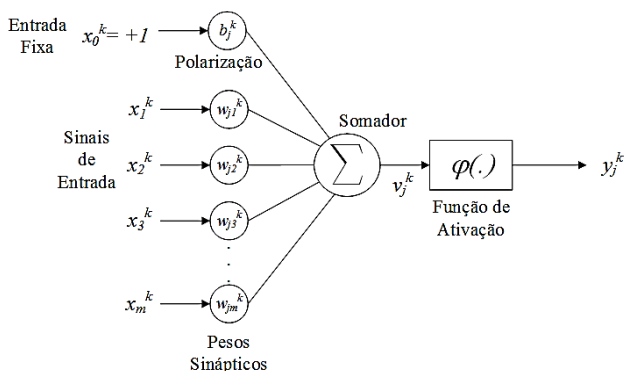


Figura 1. Modelo do neurônio artificial.

A partir do modelo do neurônio artificial, três elementos básicos podem ser identificados [8]:

- Um conjunto de sinapses, caracterizadas cada uma por um peso sináptico (w_{ji}^k);

- Um somador, que realiza uma combinação linear dos sinais de entrada, ponderados pelos respectivos pesos sinápticos;
- Uma função de ativação (φ_j^k), que limita a amplitude da saída em um intervalo normalizado ([0,1] ou [-1,1]).

O modelo do neurônio também inclui a polarização (*bias*) b_j^k , que modifica a entrada da função de ativação mediante a aplicação de um sinal com valor fixo em +1. Isso permite que um neurônio apresente uma saída não nula, mesmo que todas as entradas sejam nulas [8].

B. Principais Características das Redes Neurais

As redes neurais apresentam diversas características vantajosas que justificam seu uso em diversos campos da ciência [10]. As redes possuem a capacidade de aprender com ou sem supervisão. No primeiro caso, o treinamento ocorre com conjuntos de entrada e de saída desejada. No segundo caso, a rede não recebe informações sobre a saída desejada no seu treinamento. Assim, a rede é treinada somente com os valores de entrada e organiza sua estrutura de dados para fazer a classificação dos dados [8].

As redes neurais possuem capacidade de interpretar estímulos sensoriais [8]. Esta característica está intimamente ligada com a sua estrutura, pois a resposta da rede depende do conhecimento adquirido, que é armazenado pelos pesos sinápticos. Outra característica importante das redes neurais é a sua capacidade de realizar associações. Essa característica permite que a rede identifique padrões distintos. Além disso, a rede possui a habilidade de generalização. A generalização permite à rede reconhecer, com sucesso, padrões de mesma natureza, mas distintos dos que foram inseridos durante o seu treinamento [11].

C. Arquiteturas de Rede

As principais arquiteturas de rede são baseadas no *perceptron*, originalmente proposto por [12]. Devido à incapacidade do *perceptron* simples de resolver problemas não linearmente separáveis, é comum o uso dos *perceptrons* de múltiplas camadas (*MLP-Multilayer Perceptron*). O processamento das redes MLP é baseado na alimentação das camadas a partir da entrada até a saída (*feedforward*) e na retropropagação do erro a partir do resultado obtido na camada de saída (*backpropagation*).

A seguir, serão discutidos alguns exemplos de algoritmos de treinamento, dentre os mais populares, utilizados para classificação de padrões com a topologia MLP.

D. Gradiente Descendente

O algoritmo de treinamento baseado no *Gradiente Descendente* (GD) utiliza a função de erro quadrático, na saída da rede, para atualização dos pesos sinápticos. A atualização de cada peso é realizada de forma a minimizar a função erro quadrático na saída da rede. Esse processo se dá pelo cálculo do gradiente negativo da função de erro em função do vetor de pesos, seguido pelo ajuste do peso.

A taxa de aprendizagem utilizada nesse algoritmo tem a mesma função quando utilizada no *Resilient Propagation*. Uma baixa taxa de aprendizagem resulta em uma

convergência mais lenta, enquanto uma taxa muito elevada pode acarretar em oscilação da rede. Uma maneira de aumentar a taxa de aprendizagem no método do *Gradiente Descendente*, tornando a convergência mais rápida sem tornar a rede instável, é utilizando a variável *momentum*. A variável *momentum* realiza a ponderação dos pesos anteriores para atualização dos pesos atuais.

E. Gradiente Descendente com Momentum (GDM)

A presença da variável *momentum* no algoritmo do *Gradiente Descendente* permite à rede não responder somente ao gradiente local, mas também às tendências recentes na superfície do erro [13]. Isso significa que a atualização dos pesos atuais influencia na atualização dos próximos pesos, nas camadas subsequentes. Essa característica permite à rede convergir mais rapidamente na superfície do erro.

Os valores definidos para a variável *momentum* estão entre 0 e 1, em que o valor zero indica a ausência do *momentum* e o valor 1 faz com que a rede ignore o gradiente local, não convergindo para o erro mínimo [13].

F. Resilient Propagation

O algoritmo *Resilient Propagation* (Propagação Resiliente), adotado em redes MLP, utiliza uma estratégia localizada para atualização dos pesos sinápticos. Nesse caso, apenas as variações de sinal no gradiente local de cada peso são levadas em consideração. Todos os pesos são atualizados individualmente ao final de cada época de treinamento, ou seja, cada peso sináptico possui taxa de variação própria.

A taxa de variação é calculada em função do parâmetro da taxa de aprendizagem (η), que é atribuída no início do processamento da rede. A taxa de aprendizagem indica o quão rápido se dará o ajuste dos pesos, e pode assumir dois valores: $\eta = 0,5$ e $\eta = 1,2$ [12].

As principais características do *Resilient Propagation* (RP), que podem ser generalizadas como vantagens, são:

- Velocidade de convergência: a rede converge em uma quantidade grande de épocas, mas o tempo médio por época é da ordem de milissegundos;
- Bom desempenho na localização dos mínimos satisfatórios na superfície do erro;
- Reduzido esforço computacional.

G. Levenberg-Marquardt

O algoritmo *Levenberg-Marquardt* (LM) permite a aceleração da convergência durante a retropropagação do erro [14]. A derivada segunda da função de erro é levada em consideração, visto que o algoritmo utiliza uma aproximação pelo método de otimização de Newton [15]. A Equação (1) representa a aproximação realizada pelo *Levenberg-Marquardt*:

$$w_{j+1}^k = w_j^k + [J^T J + \mu I]^{-1} J^T \xi \quad (1)$$

em que, w_j^k representa o neurônio j da camada k , J é a matriz jacobiana que representa as derivadas primeiras da função erro

em relação aos pesos, ξ é a função erro na saída da rede e I é a matriz identidade.

Dessa maneira, o algoritmo não necessita calcular a matriz Hessiana, processo que é muito mais complexo do que calcular simplesmente o jacobiano [13]. Além disso, a equação apresenta a variável μ , que torna a matriz Hessiana definida positiva. Um valor mínimo de μ que seja positivo é suficiente para produzir uma direção descendente de busca na superfície do erro [14].

As principais características do *Levenberg-Marquardt* são:

- Velocidade de convergência: a rede converge em poucas épocas. A experiência demonstra que essa quantidade está em torno de 70 épocas;
- Robustez: isso o torna um algoritmo bastante difundido, visto que as taxas de acerto são elevadas para a resolução de muitos problemas;
- Elevado esforço computacional: o preço pago pela sua eficiência é o tempo decorrido em cada época do treinamento;
- Eficiência limitada para redes com até algumas centenas de conexões.

III. MATERIAL E MÉTODOS

O material e os métodos utilizados para a composição deste trabalho são apresentados a seguir.

A. Material

Os isoladores poliméricos objetos deste estudo foram cedidos pela Companhia Hidroelétrica do São Francisco (Chesf). Eles são de um único modelo, para linhas de transmissão de 230 kV. Uma fotografia, bem como o esquemático dimensional do modelo de isolador utilizado são mostrados na Figura 2(a) e 2(b), respectivamente.

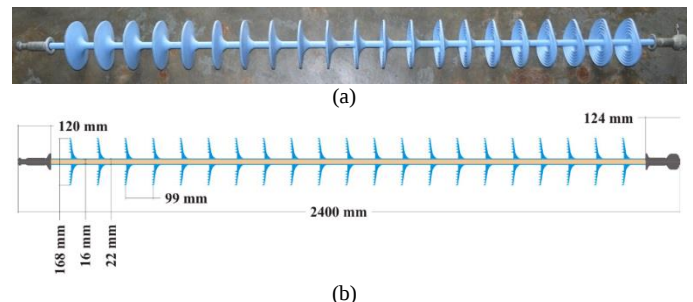


Figura 2. (a) Fotografia do isolador polimérico de 230 kV. (b) Características dimensionais do isolador polimérico.

Para a detecção dos sinais de ruído ultrassônico, foram utilizados o *Ultrasonic 2000 MPH* e o *Trisonic Scanning Module*, ambos fabricados pela *UE Systems*.

B. Métodos

O fluxograma que representa as etapas metodológicas deste trabalho é apresentado na Figura 3. Os procedimentos iniciaram com a separação dos isoladores em classes de degradação por meio de inspeção visual. Em seguida, foram realizados ensaios em laboratório para detecção dos sinais de ruído ultrassônico. Um pré-processamento dos sinais foi

realizado, com o objetivo de replicar padrões para treinamento das redes neurais. O resultado deste pré-processamento serviu de entrada para o processo de extração de atributos por meio dos Vetores de Energia dos Centróides das Subbandas Espectrais (VECSE) [4, 16].

Os comportamentos espectrais dos isoladores foram avaliados segundo a frequência de radiação acústica dominante [17]. A variável está intimamente ligada com a natureza da descarga elétrica que ocorre na superfície do isolador, bem como com a energia emitida pela descarga.

A partir da avaliação da frequência de radiação acústica dominante, os isoladores foram reagrupados nas classes de degradação. Cada classe deve conter, portanto, a mesma quantidade de amostras para serem inseridas no processamento das redes. Isso se justifica no fato de evitar a polarização da rede neural para o treinamento com os dados de determinada classe.



Figura 3. Fluxograma da metodologia adotada.

As redes neurais foram avaliadas variando-se os parâmetros a seguir:

- Função de ativação: na plataforma do Matlab, podem ser indicadas as funções tangente hiperbólica (*tansig*) e sigmóide (*logsig*);
- Algoritmo de treinamento: Resilient Propagation (*trainrp*), Gradiente Descendente (*traingd*), Gradiente Descendente com Momentum (*traingdm*) e Levenberg-Marquardt (*trainlm*);
- Taxa de aprendizagem;
- Variável *momentum*: exclusivamente para o algoritmo do Gradiente Descendente com Momentum;

As taxas de acerto médio, bem como a quantidade de épocas de treinamento e o tempo de processamento foram tomados como indicadores de desempenho das redes. Cada processamento ocorreu em 10 execuções consecutivas para obtenção dos valores médios desses parâmetros.

As redes construídas com os algoritmos RP, GD, GDM e LM sofreram variações nas quantidades de neurônios nas

camadas de entrada e oculta. No treinamento de todas as redes, foram utilizados os casos com 20, 40 e 60 neurônios na camada de entrada. A quantidade de neurônios na camada oculta variou na proporção de 1,5 neurônios da camada de entrada, resultando em 30, 60 e 90 neurônios; com exceção do *Levenberg-Marquardt*, em que foi utilizada a proporção de 0,9 entre a quantidade de neurônios na camada oculta e a quantidade de neurônios na camada de entrada, resultando em 18, 36 e 54 neurônios para os casos 1, 2 e 3, respectivamente. Para todos as redes, a quantidade de neurônios da camada de saída corresponde ao número de classes de degradação dos isoladores, que são 3. Essas variações podem ser visualizadas na Tabela I.

TABELA I. CONFIGURAÇÕES DE ENTRADA DAS REDES

Caso	Neurônios na camada de entrada	Neurônios na camada oculta	Neurônios na camada oculta com LM	Padrões na entrada
1	20	30	18	1575
2	40	60	36	
3	60	90	54	

A configuração da rede treinada com o LM foi assim escolhida após testes com variações distintas. A relação encontrada entre a quantidade de neurônios das camadas de entrada e oculta, de 0,9, foi considerada adequada para ser utilizada neste trabalho.

IV. RESULTADOS

A. Avaliação dos Parâmetros das Redes

Durante a fase de implementação dos algoritmos, os principais parâmetros que influenciam no desempenho das redes foram analisados. Dessa forma, para o caso do algoritmo RP, a quantidade de neurônios na camada oculta resultou em uma relação de 3/2 neurônios da camada de entrada, que foi considerada ótima para este caso. Outras proporções foram testadas e constatou-se um decréscimo em torno de 1% nas taxas de acerto. Ainda, como o esforço computacional varia na proporção direta da quantidade de conexões, foi considerada uma relação de compromisso entre a quantidade de neurônios e o esforço computacional. Testando-se a função de ativação, observou-se que o RP convergiu adequadamente com o uso da função tangente hiperbólica (*tansig*) nas camadas oculta e de saída. A taxa de aprendizagem utilizada, neste caso, foi $\eta = 1,2$ [13].

Avaliando-se o algoritmo GD, o mesmo valor para taxa de aprendizagem pôde ser utilizado. Entretanto, para o GD, tal valor é considerado alto, o que pode levar a rede a oscilar em torno de um mínimo local e não convergir ou atingir o critério de parada. Assim, utilizou-se $\eta = 0,01$, que é considerado adequado para a implementação do GD [13]. Neste mesmo algoritmo, a função de ativação utilizada foi a sigmóide (*logsig*). O seu uso se justifica por ser positiva e centrar a busca realizada pelo algoritmo em uma única direção [11].

A adição da variável *momentum* ao algoritmo GD permitiu a implementação de outro algoritmo, o GDM. A variável foi utilizada para aumentar a velocidade de convergência. Neste caso, a variável *momentum* utilizada foi de 0,9, que é o valor *default* do método. Valores entre 0,1 e 0,8 demonstraram

pouca variação em relação ao desempenho, portanto optou-se por não variar o valor considerado ótimo. Outra observação é que o GDM permitiu o uso da função tangente hiperbólica, pois possibilita a busca na superfície do erro em direções baseadas nas tendências de mínimos locais e mínimos globais.

No algoritmo LM, a taxa de aprendizagem não é uma variável considerada. Em vez disso, a variável μ pode assumir qualquer valor positivo. Para acelerar a velocidade de convergência do LM o valor de μ adotado é muito próximo de zero. Isso se justifica pela aproximação com o método de Newton, que converge rapidamente para o erro mínimo. O valor utilizado foi o *default* para o algoritmo: $\mu = 0,001$. A função de ativação adotada no LM foi a sigmóide. A escolha se baseou na quantidade de testes em que a rede atingiu a quantidade máxima de épocas utilizadas: 100. Nesse caso, considera-se uma não convergência.

B. Avaliação do comportamento das Redes Neurais Artificiais

O resultado dos processamentos das redes neurais, com os padrões de entrada separados nas três classes distintas, conforme a Tabela I, são mostrados nas Tabelas II, III e IV.

1) Caso 1: 20 neurônios na camada de entrada

Observando-se o comportamento das redes, como na Tabela II, constatou-se um melhor desempenho do algoritmo *Levenberg-Marquardt* em relação aos demais, com uma taxa de acerto de 88,87%. O mesmo não pode ser observado nos algoritmos *Gradiente Descendente* e *Gradiente Descendente com Momentum*: suas taxas de acerto foram da ordem de 77,59% e 70,52%, respectivamente. Além disso, ambos atingiram o número máximo de épocas (1000), não convergindo para o erro mínimo adequado.

O algoritmo *Resilient Propagation* obteve um bom desempenho. Comparativamente ao *Levenberg-Marquardt*, observou-se que obteve uma boa taxa de acerto, 81,84%, em um tempo de processamento muito reduzido. Embora a quantidade de épocas do primeiro seja alta, a rede convergiu em poucos segundos. Por outro lado, o bom desempenho do *Levenberg-Marquardt* foi pago pelo esforço computacional realizado que, mesmo convergindo em poucas épocas, realizou o processamento em uma quantidade de tempo muito superior aos demais.

TABELA II. RESULTADO DO PROCESSAMENTO DAS RNA DO CASO 1

Rede	Acerto Médio (%)	Desvio Padrão	Tempo Médio por Época (ms)	Média de Épocas
RP	81,84	0,44	34,42	512,2
GD	77,59	0,89	18,62	Não convergiu
GDM	70,52	2,85	22,09	Não convergiu
LM	88,87	0,79	623,1	64,5

2) Caso 2: 40 neurônios na camada de entrada

No Caso 2, se analisou o comportamento das redes quando se utilizou uma quantidade maior de neurônios nas camadas de entrada e oculta. Isso significa uma maior quantidade de conexões e, portanto, a rede deve processar em um tempo de processamento maior.

O esperado, neste caso, e o que se observou é que a rede obteve um desempenho maior com relação ao caso 1. O algoritmo LM obteve um bom desempenho, com uma taxa de acerto de 94,67%, o RP obteve uma taxa de acerto de 85,38%, e o GD e o GDM obtiveram 79,31% e 71,37%, respectivamente, mas novamente não convergiram. Observa-se, então, que as diferenças entre as redes persistiram.

TABELA III. RESULTADO DO PROCESSAMENTO DAS RNA DO CASO 2

Rede	Acerto Médio (%)	Desvio Padrão	Tempo Médio por Época (ms)	Média de Épocas
RP	85,38	0,82	50,55	641,3
GD	79,31	0,76	38,58	Não convergiu
GDM	71,37	1,28	22,58	Não convergiu
LM	94,67	0,60	2.707,0	66,9

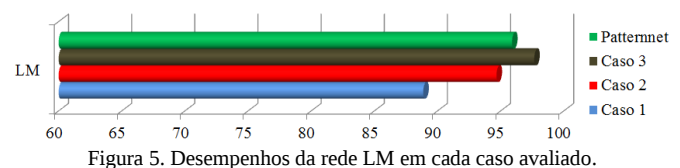
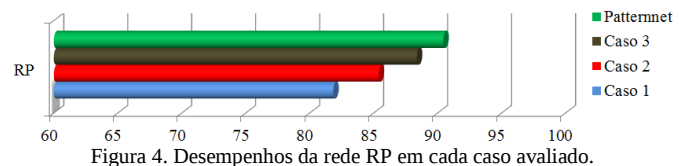
3) Caso 3: 60 neurônios na camada de entrada

O caso 3 apresenta os melhores desempenhos de cada uma das redes. O desempenho do *Levenberg-Marquardt* demonstrou sua popularidade na resolução de problemas de reconhecimento de padrões. Com uma taxa de acerto de 97,66%, observou-se que o algoritmo realizou o processamento com um esforço computacional elevado, mesmo convergindo em poucas épocas. Ainda assim, seu uso se justifica pela confiabilidade do resultado. As redes *Gradiente Descendente* e *Gradiente Descendente com Momentum* não são adequadas para resolução de problemas deste tipo, visto que não convergem para o erro mínimo. O que se infere, nesses casos, é a convergência para erros mínimos locais em que as redes oscilam, não convergindo para erros mínimos globais na superfície do erro.

TABELA IV. RESULTADO DO PROCESSAMENTO DAS RNA DO CASO 3

Rede	Acerto Médio (%)	Desvio Padrão	Tempo Médio por Época (ms)	Média de Épocas
RP	88,37	0,65	64,27	686,1
GD	78,23	3,37	58,61	Não convergiu
GDM	74,38	1,09	39,13	Não convergiu
LM	97,66	0,54	8.853,8	55,4

Os processamentos realizados até então foram implementados com redes *feedforward* no Matlab. O programa ainda apresenta uma ferramenta específica para reconhecimentos de padrões, a *patternnet*. Os desempenhos das redes RP e LM foram comparados e seus resultados são mostrados na Figura 4 e na Figura 5, respectivamente.



C. Comparação entre o Resilient Propagation e o Levenberg-Marquardt com a 'patternnet'

Com o intuito de melhorar o desempenho das redes, os algoritmos *Resilient Propagation* e *Levenberg-Marquardt* foram implementados com a *patternnet*. Os resultados, para os melhores casos, estão mostrados na Tabela V. Observou-se, neste caso, uma melhora no desempenho do algoritmo *Resilient Propagation*. A taxa de acerto aumentou em relação ao melhor caso apresentado anteriormente. Além disso, a rede convergiu em uma quantidade de épocas significativamente menor. Deste resultado, pode-se inferir que o comportamento do algoritmo é mais adequado quando utilizado em uma ferramenta específica para reconhecimento de padrões, como a *patternnet*.

O mesmo resultado não pode ser observado no algoritmo *Levenberg-Marquardt*. Sua taxa de acerto diminuiu e o esforço computacional foi maior. Em casos muito específicos, como este, a escolha do algoritmo mais adequado fica sujeita ao desempenho durante o teste, em que o esforço computacional é menor, já que os parâmetros estão definidos desde a etapa de treinamento.

TABELA V. RESULTADO COM O PATTERNNET

Rede	Acerto Médio (%)	Desvio Padrão	Tempo Médio por Época (ms)	Média de Épocas
RP	90,44	1,51	68,56	309,2
LM	95,90	1,5e-14	23.485,7	56

A evolução dos comportamentos das redes ao longo dos casos testados está mostrada gráfico de barras, na Figura 4. Constatou-se, pela análise da Figura 6, que o algoritmo que obteve os melhores desempenhos em relação à taxa de acerto, em todos os casos estudados, foi o *Levenberg-Marquardt*.

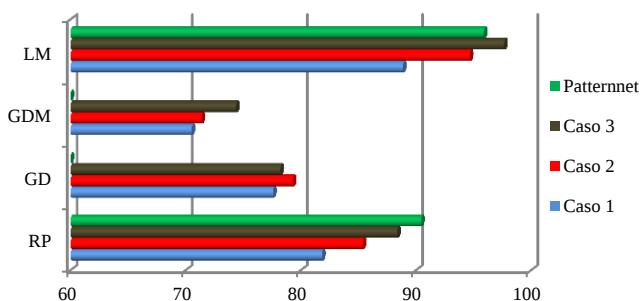


Figura 6. Desempenhos das redes para cada caso avaliado.

V. CONCLUSÕES

A busca de algoritmos que permitam classificar sinais acústicos foi o foco principal deste trabalho. Constatou-se o bom desempenho dos algoritmos baseados no *Resilient Propagation* e no *Levenberg-Marquardt* na resolução de problemas de reconhecimento de padrões. Ambos demonstraram taxas de acerto bastante satisfatórias, em que pese a necessidade de busca do algoritmo com o menor nível de incerteza possível. Além disso, opta-se por aquele algoritmo que apresente a melhor taxa de acerto em um tempo de processamento menor, durante o teste da rede.

Tais algoritmos serão utilizados em trabalhos futuros, em que padrões de natureza distinta, como de imagens ultravioletas, podem ser reconhecidos com uso de métodos robustos e eficientes e computacionalmente fáceis de ser implementados.

AGRADECIMENTOS

Os autores agradecem ao CNPq e Proex/CAPES pelo apoio ao desenvolvimento da pesquisa, e à Chesf pela gentileza em ceder as amostras de isoladores.

REFERÊNCIAS

- [1] HARROLD, R. T. Acoustic Waveguides for Sensing and Locating Electrical Discharges in High Voltage Power Transformers and other Apparatus. IEEE Transactions on Power Apparatus and Systems, v. 98, n. 2, p. 449-457, 1979.
- [2] LUNDGAARD, L. E. Partial Discharge XIII: acoustic partial discharge detection-fundamental considerations. IEEE Electrical Insulation Magazine, v. 8, p. 25-31, 1992.
- [3] ABDEL-SALAM, M.; ABDEL-SATTAR, S.; SAYED, Y.; GHALLY, M. Early Detection of Weak Point in MEEC Distribution System. In: Industry Applications Conference Record of the 2001 IEEE, 2001, Chicago. v. 4, p. 2541-2545.
- [4] FERREIRA, T. V.; GERMANO, A. D.; COSTA, E. G. Ultrasound and Artificial Intelligence Applied to the Pollution Estimation in Insulations. IEEE Transactions on Power Delivery, v. 12, p. 583-589, 2012.
- [5] ROCHA, P. H. V.; FONTGALLAND, G., Measuring the radiation bands of overhead power lines glass insulators. Proceeding of the IEEE 2014 International Conference on Antenna Measurements & Applications. França, 2014.
- [6] HAYKIN, S. Redes Neurais: princípios e prática. 2. ed. Porto Alegre: Bookman, 2001.
- [7] ABDALLA, L. S.; VOLOTÃO, C. F. S. Estudo da Configuração de Diferentes Arquiteturas de Redes Neurais Artificiais MLP para Classificação de Imagens Ópticas. Simpósio Brasileiro de Sensoriamento Remoto - XVI SBSR. Foz do Iguaçu, 2013.
- [8] LUDWIG JR., O.; COSTA, E. M. M. Redes Neurais: Fundamentos e Aplicações com Programas em C. 1. ed. Rio de Janeiro: Editora Ciência Moderna, 2007.
- [9] MCCULLOCH, W. S.; PITTS, W.A Logical Calculus of the Ideas Immanent in Nervous Activity. Bulletin of Mathematical Biophysics, v. 5, p. 115-133, 1943.
- [10] AL-GEELANI, N. A.; PIAH, M. A. M.; BASHIR, N. A Review on Hybrid Wavelet Regrouping Particle Swarm Optimization Neural Networks for Characterization of Partial Discharge Acoustic Signals. Renewable and Sustainable Energy Reviews, v. 45, p. 20-35, 2015.
- [11] VELLASCO, M. M. B. R. Redes Neurais Artificiais. Apostila. Pontifícia Universidade Católica do Rio de Janeiro: PUC-RJ, 2007.
- [12] RIEDMILLER, M.; BRAUN, H. RPROP - A Fast Adaptive Learning Algorithm. In: International Symposium on Computer and Information Science, 1992. Anais do ISCIS VII.
- [13] MATHWORKS. Matlab® Neural Network Toolbox. Function Approximation and Linear Regression. Gradient Descent with Momentum Backpropagation, 2015. Disponível em: http://www.mathworks.com/help/nnet/ref/traingdm.html?s_tid=srchtitle
- [14] BISHOP, C. M. Neural Networks for Pattern Recognition. Clarendon Press, Oxford. 1995.
- [15] HAGAN, M. T; MENHAJ, M. B. Training Feedforward Networks with the Marquardt Algorithm". IEEE Transactions on Neural Networks, v. 5: p. 989-993, 1994.
- [16] FERREIRA, T. V. Estimção Inteligente da Poluição de Isolamentos Elétricos Baseada nos VECSE do Ruído Ultrassônico. Universidade Federal de Campina Grande. Tese de Doutorado. Campina Grande: UFCG, 2011.
- [17] HARROLD, R. T. Acoustical Technology Applications in Electrical Insulation and Dielectrics [J]. IEEE Transactions on Electrical Insulation, v. 20, n. 1, p. 3-19, 1985.