

ANÁLISE COMPARATIVA DO DESEMPENHO DE VÁRIAS ARQUITETURAS DE REDES NEURAIS ARTIFICIAIS APLICADAS AO PROCESSAMENTO DE ALARMES

TATIANE M. MACHADO¹, ADRIANO P. DE MORAIS¹, GHENDY CARDOSO JR¹, JOÃO M. ZAUK¹, PAULO C. FRITZEN^{1,2}
e RÉGIS BOLZAN¹

¹*Centro de Estudos em Energia e Meio Ambiente (CEEMA), Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Santa Maria, Santa Maria, RS*

²*Laboratório de Indústria, Instituto Federal de Educação, Ciência e Tecnologia do Tocantins, IFTO Campus Palmas, Palmas, TO*

E-mails: thaty_m_m@yahoo.com.br; adriano@mail.ufsm.br, ghendy@smail.ufsm.br, joaozauk@gmail.com; paulocicero@ifto.edu.br, regisbolzan@gmail.com

Resumo— A tarefa de processamento de alarmes e diagnóstico de faltas em sistemas de potência tem sido motivo de inúmeras pesquisas, uma vez que as empresas de energia necessitam reduzir ao máximo a duração das interrupções no fornecimento de energia, quando estas forem ocasionadas por desligamentos não programados. A ocorrência de uma falta no sistema elétrico é reportada aos centros de controle e supervisão por meio de inúmeras mensagens de alarmes. Cabe ao operador interpretá-los e agir apropriadamente de modo a restabelecer o sistema. No entanto, o tempo e o esforço envolvido nessa tarefa são bastante onerosos. No sentido de solucionar esse problema, diversas técnicas baseadas em inteligência computacional têm sido propostas na literatura. Seguindo esta tendência, este artigo propõe uma análise comparativa entre sete diferentes tipos de redes neurais artificiais, de modo a identificar a mais adequada à tarefa de processamento de alarmes.

Palavras-chave— Processamento de alarmes, proteção de sistemas de potência, redes neurais artificiais.

1 Introdução

O crescente aumento na demanda de energia elétrica e a atual sobrecarga dos sistemas elétricos de potência tornam necessário o desenvolvimento de ferramentas computacionais capazes de aumentar a confiabilidade do sistema e garantir a continuidade do fornecimento de energia. Essas ferramentas devem levar em conta a experiência dos operadores do sistema no que diz respeito ao reconhecimento de uma falta, diagnóstico e tomada de ações corretivas [1].

Quando ocorre um desligamento não programado, os operadores do sistema são submetidos a uma situação de *stress*, uma vez que é necessário tomar decisões rápidas e corretas no sentido de restabelecer o fornecimento de energia. No entanto, essa tarefa pode se tornar bastante complexa, pois cada falha que ocorre no sistema gera uma grande quantidade de alarmes e mensagens que são enviadas ao centro de controle, cabendo ao operador agir o mais rápido possível de modo a corrigir o problema.

Grande parte das pesquisas que vêm sendo desenvolvidas na área de processamento de alarmes está relacionada à inteligência computacional. Em [2], é proposto um processador de alarmes baseado em sistemas especialistas. O conjunto de regras é baseado no conhecimento técnico e empírico de engenheiros e técnicos. No entanto, os sistemas especialistas apresentam desvantagens, pois é difícil construir regras capazes de cobrir todas as possíveis situações de faltas [3].

A teoria dos conjuntos *fuzzy* também tem sido muito utilizada. Em [4], é descrito um processador de

alarmes que utiliza essa técnica para manipular as incertezas e a falta de informação.

Em [5], são propostos métodos de processamento de alarmes baseados em algoritmos genéticos, sendo que estes possuem a vantagem de encontrar múltiplas soluções globais de maneira direta e eficiente.

Em [1], é apresentado um processador de alarmes baseado em Redes Neurais Artificiais (RNAs). O sistema é capaz de diagnosticar faltas múltiplas mesmo se o conjunto de alarmes estiver incompleto. Em [6], o processador de alarmes é composto por diversas RNAs, sendo cada uma responsável por detectar defeitos envolvendo um número limitado de componentes do sistema.

Seguindo esta tendência, este artigo está direcionado ao uso de redes neurais, tendo como objetivo analisar comparativamente diversas arquiteturas de redes, de modo a identificar a mais adequada ao processamento de alarmes.

As seguintes redes neurais são consideradas: BP (*Backpropagation*), RBF (*Radial Basis Function*), PNN (*Probabilistic Neural Network*), GRNN (*Generalized Regression Neural Network*), SOM (*Self-Organizing Maps*), Kohonen e Elman.

2 Processamento de alarmes

Atualmente, os centros de controle são compostos por diversos computadores interconectados que, por meio de *softwares* específicos, têm acesso imediato às informações sobre as condições operacionais da rede elétrica. O *software* que presta suporte ao operador é denominado de EMS (*Energy Management System*) e é composto de diversas ferramentas, entre elas o sistema SCADA

(*Supervisory Control and Data Acquisition*), que coleta, em tempo real, dados sobre as condições atuais do sistema elétrico. Estes dados consistem em medidas analógicas (tensão e corrente) e digitais (estado de disjuntores, dispositivos de proteção, entre outros). O SCADA permite também que o operador execute ações sobre os equipamentos remotamente controlados, como abertura de disjuntores, chaveamento de tapes de transformadores, entre outras [7].

Em situações que podem levar o sistema a uma condição insegura ou instável, tais como tempestades, flutuações de carga ou falhas em equipamentos, os operadores devem atuar de forma a restabelecer as condições aceitáveis de operação. As ações envolvidas nesse restabelecimento podem ser a identificação do problema, localização das possíveis falhas e acionamento de equipes de manutenção, recomposição de subestações ou redes desligadas por dispositivos de proteção, entre outras. Essas tarefas podem exigir bastante tempo, trazendo prejuízo tanto para os consumidores quanto para as concessionárias de energia. A ocorrência de equívocos por parte do operador em tais situações pode levar à deterioração do estado do sistema, implicando em desligamentos de cargas, danificação de equipamentos e até mesmo ao *blackout* do sistema [7].

Segundo [3-8], os operadores do sistema sentem dificuldades em interpretar o grande número de alarmes reportados pelo sistema de supervisão. Esta dificuldade se torna ainda maior quando uma grande quantidade de equipamentos é desligada, implicando em um grande número de alarmes e mensagens, disparados em um curto período de tempo.

O principal objetivo dos processadores inteligentes de alarmes é reduzir a quantidade de informações a ser processada pelos operadores, acelerando o processo de tomada de decisões e reduzindo a probabilidade de erros [9]. Ainda, os processadores de alarmes têm como funções melhorar a forma e o conteúdo das mensagens apresentadas ao operador, informar o período de início e fim das condições anormais, apresentar, quando possível, conclusões cronológicas sobre a falha e, em alguns casos, sugerir ao operador as ações corretivas a serem tomadas.

Uma vez que não existe uma formulação analítica eficaz capaz de solucionar o problema de processamento de alarmes, soluções baseadas em inteligência computacional se tornaram indispensáveis.

3 Arquitetura das redes neurais analisadas

As redes neurais artificiais são compostas por elementos processadores, ou nós, que recebem informações dos neurônios de entrada, processam a informação, e produzem uma saída que é dependente da função de transferência (ou função de ativação) utilizada [10]. A seguir, serão apresentadas as

principais características das RNAs analisadas, tão bem como as suas configurações finais de simulação.

As redes neurais foram implementadas no software MATLAB[®]. Para a fase de treinamento das RNAs, foi criado um sistema teste constituído de 22 alarmes. A combinação desses alarmes pode indicar 13 falhas diferentes ou 11 casos de operação normal de um equipamento elétrico, totalizando 24 situações, as quais deverão ser identificadas pelas redes. Os padrões de treinamento e de teste utilizados nas simulações serão melhores apresentados na seção 4.

3.1 Rede Backpropagation (BP)

É uma rede *Perceptron* de múltiplas camadas (MLP), que utiliza o algoritmo de treinamento retropropagação, do inglês *backpropagation*. Possui uma camada de entrada, n camadas intermediárias e uma camada de saída. O número n de camadas intermediárias é ajustado de maneira heurística, de modo que a rede apresente o melhor desempenho.

A rede BP utilizada nas simulações possui três camadas: uma de entrada, uma escondida e uma de saída. As funções de ativação para as quais a rede apresentou o melhor desempenho foram: *tansigmóide* para a camada intermediária e *sigmóide* para a camada de saída.

A camada de entrada possui 22 neurônios, que recebem a combinação dos alarmes e a camada de saída possui 14 neurônios (13 falhas mais operação normal).

O treinamento foi realizado com a função *traingdx*, que atualiza os valores dos pesos e bias de acordo com o gradiente do *momentum* e uma taxa de aprendizado adaptativa. A rede foi treinada com o coeficiente de *momentum* igual a 0,8 e com 700 épocas.

3.2 RBF

A rede RBF apresenta uma camada intermediária com neurônios com função de ativação de base radial e uma camada de saída com neurônios lineares.

A implementação desta rede requer apenas a definição de um parâmetro denominado *spread*. A RBF não permite que se escolha o número de neurônios da camada intermediária, uma vez que a própria rede ajusta a quantidade de neurônios a ser utilizada. Esse ajuste é bastante simples, ou seja, o número de neurônios da camada intermediária é igual ao número de casos apresentados à rede durante a fase de treinamento, e a camada de saída apresenta tantos neurônios quantos forem as saídas.

Dessa forma, para o caso em análise, a RBF apresenta uma arquitetura 22 – 24 – 14. Por meio do critério da tentativa e erro, o *spread* foi ajustado em 2,9.

3.3 PNN

A rede probabilística é uma variação da RBF. É composta por uma camada de entrada, uma intermediária e a camada de saída. A camada escondida apresenta uma função de base radial como função de ativação, enquanto a camada de saída possui neurônios competitivos.

A PNN apresenta uma fácil adaptação a mudanças no cenário na qual foi treinada, uma vez que os pesos da camada intermediária (camada de base radial) são iguais aos vetores de entrada, e os pesos da camada de saída são iguais aos vetores de saída (já estabelecidos). Então, qualquer mudança nos padrões de entrada pode ser facilmente repassada à matriz peso, sem a necessidade de um novo treinamento.

A implementação desta rede é tão simples quanto a RBF. O único parâmetro de ajuste continua sendo o *spread*. Após várias simulações, foi escolhido um *spread* igual a 0,45, baseado no critério da tentativa e erro.

3.4 GRNN

A GRNN também é uma variação da rede RBF. Possui três camadas, sendo que a camada intermediária apresenta uma função de ativação de base radial e a camada de saída é formada por neurônios lineares.

Do mesmo modo que a rede probabilística, a GRNN também possui os pesos da camada intermediária iguais aos vetores de entrada e os pesos da camada linear iguais aos vetores de saída. Esta rede tem a característica de indicar a real semelhança entre os vetores utilizados no treinamento e os vetores que são apresentados à rede, fato que não se verifica nas demais redes e que a torna bastante atraente para a solução de problemas de classificação de padrões.

Por ser uma rede muito semelhante à PNN, foi utilizado o mesmo *spread* de 0,45 para que se pudesse fazer uma comparação entre ambas.

3.5 SOM

Diferentemente das redes já citadas, a rede SOM possui aprendizado não-supervisionado. Possui somente duas camadas: a camada de entrada, por onde é alimentada com os dados relativos a cada variável, e a camada de saída, que possui os neurônios competitivos.

A rede SOM não foi capaz de separar os padrões de treinamento em grupos, como ocorreu com as outras redes já citadas. Diante desse problema, optou-se por utilizar duas redes: uma especializada em identificar as 13 falhas e outra especializada em identificar os casos de operação normal do equipamento.

Foi utilizada uma rede com topologia retangular, e para calcular a distância entre um neurônio

particular e seus vizinhos, foi usada a distância Euclidiana. Foram necessárias 1000 épocas e 13 neurônios para treinar a rede especializada em encontrar as 13 falhas, e 400 épocas e 1 neurônio para a rede que detecta a condição de operação normal.

3.6 Kohonen

A rede de Kohonen é bastante semelhante à rede SOM. Possui somente duas camadas, sendo a última do tipo competitiva. O aprendizado também é do tipo não-supervisionado.

Para a implementação desta foram utilizados 24 neurônios de saída, sendo que 13 neurônios representam as 13 falhas e os outros 11 neurônios representam a operação normal do equipamento. Foram necessárias 1200 épocas e uma taxa de aprendizado de 0,04.

3.7 Elman

A rede Elman faz parte do grupo das redes recorrentes. Possui três camadas e é muito semelhante à rede *Backpropagation*, no entanto, possui uma realimentação da camada intermediária para a de entrada. Devido a essa realimentação, tem a capacidade de reconhecer padrões temporais, ou seja, considera a ordem cronológica em que os dados são apresentados.

Foram utilizados 17 neurônios na camada intermediária, e as funções de ativação *tansigmóide* e *sigmóide* para a camada escondida e de saída, respectivamente. Outros parâmetros utilizados foram 100 épocas, taxa de aprendizado de 0,03 e *momentum* de 0,8.

4 Padrões de treinamento e de teste

Conforme já mencionado, o treinamento das redes neurais foi baseado em um sistema teste composto de 22 alarmes. A partir desses alarmes, foram criados 24 padrões de treinamento, ou seja, esses padrões de treinamento são combinações de alarmes que podem representar falhas ou a operação normal de um equipamento elétrico. A Tabela 1 apresenta os conjuntos de alarmes utilizados na fase de treinamento e suas respectivas interpretações.

Para a fase de teste das redes, foram criadas outras 160 combinações de alarmes, sendo que estas foram divididas em:

- 47 casos com um bit diferente do padrão de treinamento;
- 44 casos com dois bits diferentes do padrão de treinamento;
- 38 casos com três bits diferente do padrão de treinamento;
- 31 casos com quatro bits diferentes do padrão de treinamento.

Esses casos de teste foram criados com o objetivo de verificar a capacidade de generalização das redes.

Tabela 1 - Padrões de treinamento

Nº	Alarmes	Falha
1	a1, a3, a4, a15, a19	FLH - 1
2	a3, a4, a18, a19, a20	FLH - 2
3	a3, a4, a11, a12, a18, a19, a20	FLH - 3
4	a5, a8, 19, a16	FLH - 4
5	a5, a8, a9, a11, a12, a16, a17	FLH - 5
6	a3, a18, a20	FLH - 6
7	a2, a10, a19	FLH - 7
8	a4, a6, a7, a15	FLH - 8
9	a3, a13, a21, a22	FLH - 9
10	a14, a15	FLH - 10
11	a16, a17	FLH - 11
12	a18, a19, a20	FLH - 12
13	a21, a22	FLH - 13
14	Sem alarmes	O.N.
15	a1, a5, a9, a13, a20	O.N.
16	a7, a15, a20	O.N.
17	a4, a10, a13, a17, a22	O.N.
18	a1, a4, a11, a13	O.N.
19	a6, a16, a20, a22	O.N.
20	a2, a5, a11, a21	O.N.
21	a14, a17, a19	O.N.
22	a4, a8, a12, a16, a22	O.N.
23	a5, a11, a15	O.N.
24	a2, a9, a17, a21	O.N.

FLH- X: representa a falha nº X
 O.N.: operação normal

A representação binária foi utilizada para representar os vetores de entrada e saída das RNAs. Para indicar a recepção de um alarme, foi utilizado o valor binário "1", enquanto que o valor "0" foi utilizado para indicar a não recepção do alarme. A Tabela 2 apresenta a representação binária dos vetores de entrada, de acordo com os alarmes mostrados na Tabela 1.

Tabela 2 - Representação binária dos padrões de treinamento

	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	a12	a13	a14	a15	a16	a17	a18	a19	a20	a21	a22
FLH - 1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 2	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
FLH - 3	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
FLH - 4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 7	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLH - 13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O.N.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Quadro cheio – representa o bit "1"
 Quadro vazio – representa o bit "0"

Para ilustrar alguns casos utilizados na fase de testes, será considerado o conjunto de alarmes que representa a Falha 9. Utilizando a notação binária, esta seqüência de alarmes pode ser escrita como:

0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0

A partir dessa seqüência de alarmes, foram inseridos ruídos de 1 a 4 bits, de modo a identificar a capacidade das redes em reconhecer conjuntos de alarmes diferentes dos apresentados na fase de treinamento. A seguir são apresentados os casos de teste originados a partir da seqüência de alarmes que representa a Falha 9.

Ruído de 1 bit na posição a6:

0 0 1 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 1 1 0

Ruído de 2 bits nas posições a6 e a10:

0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 1 1 0

Ruído de 3 bits nas posições a6, a10 e a18:

0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 1 1 1 0

Ruído de 4 bit nas posições a6, a10, a18 e a23:

0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 1 1 1 1

5 Resultados e discussões

A Tabela 3 apresenta a porcentagem de acertos das RNAs, quando submetidas aos padrões de teste.

Tabela 3 – Porcentagem de acertos das RNAs

Nível de ruído	BP	RBF	PNN	GRNN	SOM	Kohonen	Elman
1 bit (4,5%)	91,5%	85,1%	96,8%	100%	100%	96,8%	81,9%
2 bits (9,1%)	77,8%	61,4%	89,8%	100%	94,3%	89,8%	77,3%
3 bits (13%)	68,4%	63,2%	94,7%	100%	97,4%	94,7%	69,8%
4 bits (18%)	54,8%	50,0%	93,5%	100%	95,2%	93,5%	59,7%

Nota-se que a rede *Backpropagation* reduz o seu desempenho com o acréscimo de ruídos, de modo que para um ruído de 18%, seu desempenho reduz-se aproximadamente à metade. No caso da rede RBF, foram consideradas como verdadeiras as respostas maiores que 0,46. Muitas vezes a rede apresentou a resposta esperada juntamente com outros resultados errados. Ainda, em muitos casos, a rede não forneceu nenhuma resposta acima do limiar estabelecido.

A PNN obteve melhor desempenho para seqüências de alarmes com ruídos de 3 e 4 bits do que para seqüências com apenas 2 bits de ruído. Isso aconteceu porque a adição de 2 bits de ruído gerou vários casos que exigiam respostas múltiplas, enquanto as seqüências com ruídos de 3 e 4 bits resultaram em poucos casos com falhas múltiplas. Como a PNN não tem a capacidade de detectar mais de uma falha, obteve melhor desempenho para níveis de ruídos maiores.

Uma solução para a rede PNN e outras redes que não detectam falhas múltiplas é utilizar a técnica de redes modulares, agrupando em redes distintas os eventos que possam ocorrer simultaneamente.

A GRNN apresentou o desempenho máximo em todos os casos. Possui a importante característica de mostrar exatamente o quanto uma seqüência de alarmes apresentada à rede é semelhante às seqüências de alarmes utilizadas no treinamento.

Conforme já mencionado, foi criada uma rede SOM especializada em localizar as 13 falhas e outra especializada em localizar os casos de operação normal. No entanto, duas seqüências de alarmes apresentadas à rede exigiam respostas múltiplas que estavam localizadas em redes distintas, e ainda assim a rede não as classificou corretamente.

A rede de Kohonen, embora tenha obtido uma porcentagem de acertos inferior à rede SOM, possui um melhor desempenho, pois foi feita apenas uma rede e esta teve a capacidade de distinguir os 13 casos de falhas e os 11 casos de operação normal, mesmo que tenha sido utilizado um neurônio para representar cada caso de operação normal.

A rede Elman não pode ser comparada diretamente com as demais porque possui um comportamento diferente. A rede Elman leva em conta a ordem temporal em que os alarmes são apresentados. Assim, com a inserção de ruídos aos padrões originais, não foram apenas alguns bits que foram alterados, mas sim toda a ordem temporal da seqüência apresentada à rede.

Quanto ao tempo de treinamento, a rede SOM foi a que a exigiu um maior tempo, levando cerca de um minuto para que o treinamento fosse concluído. No que diz respeito à facilidade de implementação, a rede de Kohonen foi a que exigiu maior esforço para se chegar ao número de épocas e taxa de aprendizado adequadas.

Considerando um *ranking* de desempenho, a GRNN foi a que mais se destacou. Logo após, tem-se a PNN e a Kohonen, que fornecem resultados iguais e sempre apresentam resultados corretos, a menos que existam falhas múltiplas. Em seguida está a rede *Backpropagation*, que identifica falhas múltiplas, mas em alguns casos fornece respostas erradas. A rede SOM se encontra logo após, já que não teve a capacidade de distinguir os casos de falhas e operação normal quando foi utilizada apenas uma rede. A RBF foi a que obteve o menor desempenho, pois apresentou alguns resultados errados e muitas vezes não forneceu nenhuma resposta. A rede Elman não foi colocada no *ranking* pelo fato de apresentar um comportamento diferente.

6 Conclusões

Os resultados obtidos indicam que a rede mais apropriada para ser utilizada em um processador inteligente de alarmes é a GRNN, pois obteve desempenho máximo para todos os casos testados, além de apresentar um treinamento bastante simples e rápido. A GRNN também possui a importante característica de fornecer respostas múltiplas. Com isso, é possível ordenar as possíveis explicações para os alarmes recebidos, de modo que se o operador se certificar de que a primeira solução não é correta, outra com maior probabilidade poderá ser selecionada.

De modo geral, o desempenho das RNAs foi satisfatório, o que indica que a aplicação de redes neurais em processamento de alarmes é bastante adequada, principalmente devido à capacidade de generalização das redes.

Referências Bibliográficas

- [1] NAVARRO, V. *et al.*, "Artificial Neural Networks for Power Systems Diagnosis". IEEE International Conference on Neural Networks. IEEE World Congresso on Computational Intelligence. Vol. 6, July 1994, pp. 3738-3743.
- [2] VALE, Z.A.; MOURA A.M., "An expert system with temporal reasoning for alarm processing in power system control centers". IEEE Transactions on Power Systems, Vol. 8, N° 3, Agosto 1993, pp. 1307-1314.
- [3] EL-SAYED, M.H.; ALFUHAID, A.S., "ANN-based approach for fast fault diagnosis and alarm handling of power systems". International Conference on Advances in Power System Control, Operation and Management, APSCOM-00. Vol. 1, 30 Oct.-01 Nov. 2000, pp. 54-58.
- [4] CHOI, S. S. *et al.*, "Development of an on-line fuzzy expert system for integrated alarm processing in nuclear plants. IEEE Transactions on Nuclear Science, Vol. 42, N° 4, Agosto 1995, pp. 1406-1418.
- [5] NEIS, P. *et al.*, "Processamento de alarmes em sistemas elétricos de potência utilizando algoritmos genéticos". VI SIMPASE. Simpósio de Automação de Sistemas Elétricos. São Paulo, SP, Brasil, 2005.
- [6] COUTTO FILHO, M.B. do *et al.* "Localização de defeitos em sistemas de energia elétrica utilizando sistemas inteligentes". In: XV SEMINÁRIO NACIONAL DE PRODUÇÃO E TRANSMISSÃO DE ENERGIA ELÉTRICA, 1999, Foz do Iguaçu.
- [7] NEIS, P., "Processamento inteligente de alarmes empregando algoritmos genéticos". Dissertação de Mestrado. Universidade Tecnológica Federal do Paraná (UTFPR). Curitiba, Brasil, 2006.
- [8] PRICE, W. R. *et al.*, "Survey on excessive alarms". IEEE Transactions on Power Systems, Vol. 4, N° 3, August 1989, pp. 950-956.
- [9] KIRSCHEN, D. S., WOLLENBERG, B. F., "Intelligent alarm processing in power systems". Proceedings of the IEEE, Vol. 80, N° 5, May 1992, pp. 663-672.
- [10] CHAN, E. H. P., "Using Neural Network to Interpret Multiple Alarms". IEEE Computer Applications in Power, Vol. 3, Abril 1990, pp. 33-37.