



INSTITUTO DE ENGENHARIA NUCLEAR

RT-IEN- 37/2002

Sistema de Auxílio ao Operador para Identificação Dinâmica de Transientes em Centrais Nucleares

por

Antônio Carlos de Abreu Mól

Setembro/2002

NOTA
ESTE RELATÓRIO É PARA USO EXCLUSIVO DO
INSTITUTO DE ENGENHARIA NUCLEAR

O direito a utilização de informações relacionadas ao trabalho de pesquisa realizado no IEN é limitado aos servidores da CNEN e pessoal de organizações associadas, nos limites dos termos contratuais que regem os respectivos convênios. O conteúdo dos relatórios não pode ser separado ou copiado sem autorização escrita do IEN



INSTITUTO DE ENGENHARIA NUCLEAR

Título: Sistema de Auxílio ao Operador para Identificação Dinâmica de Transientes em Centrais Nucleares

Autor(es):
Antônio Carlos de Abreu Mól, Roberto Schirru e Aquilino Senra Martinez

e-mail:
mol@ien.gov.br

Identificação:
RT-IEN-37

Nº de
páginas:
25

Tipo de Divulgação:
Irrestrita (x) Restrita ()

Divulgar para:
IEN

Localização:
INTRANET

Publicação externa associada (congresso/periódico):

2º. CONGRESSO INTERNACIONAL DE AUTOMAÇÃO, SISTEMAS E INSTRUMENTAÇÃO

Palavras chave:

Identificação de transientes, usinas nucleares, redes neurais.

Resumo:

Este trabalho apresenta uma nova metodologia, baseada em RNA, para identificação de transientes a qual permite tanto uma identificação dinâmica quanto a resposta “Não Sei” para eventos desconhecidos. O método usa duas RNAs multicamadas treinadas com o algoritmo “backpropagation”. A primeira rede é responsável pela identificação dinâmica. Esta rede utiliza como entradas um pequeno conjunto (através de uma janela temporal móvel) dos valores recentes das variáveis de estado, não necessitando desta forma de um evento inicializador. A outra rede é usada para validar a identificação feita pela primeira rede através da validação de cada variável. Esta RNA permite que o sistema possa emitir a resposta “Não Sei”.

No processo de validação, foi proposto um problema de identificação de transientes de uma usina nuclear PWR, composto de 15 acidentes simulados, com dados ruidosos de modo a avaliar também a robustez do método. Os resultados obtidos mostraram a capacidade do método em trabalhar com sistemas dinâmicos e a emitir a resposta “Não Sei” para eventos desconhecidos.

Abstract:

This work presents a new approach for neural network based transient identification which allows either dynamic identification and “don’t know” response. Such approach uses two multilayer neural networks trained with backpropagation algorithm. The first one is responsible for the dynamic identification. This NN uses, as input, a short set (in a movable time window) of recent measurements of each variable avoiding the necessity of using starting events. The other one is used to validate the instantaneous identification (from the first net) through the validation of each variable. This net is responsible for allowing the system to provide “don’t know” response.

In order to validate the method, a NPP transient identification problem comprising 15 postulated accidents, simulated for a pressurized water reactor, was proposed in the validation process it has been considered noising data in order to evaluate the method robustness. Obtained results reveal the ability of the method in dealing with both dynamic identification of transients and correct “don’t know” response.

Emissão		Nome	Rubrica	Data
Data:	Elaboração:	Antônio Carlos de Abreu Mol		
Divisão:	Revisão:	José Carlos Soares de Almeida		
Serviço:	Aprovação :	Isaac José Obadia		

Instituto de Engenharia Nuclear:

Via 5 s/n, Cidade Universitária, Ilha do Fundão, CEP 21945-970, CP 68.550, Rio de Janeiro – RJ - Brasil .

Tel.: 00 55 21 2560-4113 Internet: www.ien.gov.br

Logo após o acidente de Three Mile Island Unit-2 (TMI-2), em março de 1979, foi feita uma revisão crítica nos projetos das usinas nucleares, principalmente no que se referia a interface homem-máquina. Os resultados fizeram que a Comissão de Regulamentação Nuclear (NRC) emitisse uma série de documentos normativos, que visando aumentar a capacidade de resposta à condição de acidentes, incluíram sistemas computadorizados no auxílio à operação, genericamente denominados de “Safety Parameter Display System (SPDS)”. Estes sistemas computadorizados tiveram inicialmente suas funções definidas de forma superficial, e posteriormente através do Suplemento 1 do documento NUREG 0737 [1] que foram definidas, de forma mais detalhada.

Mais recentemente, aspectos relacionados a fatores humanos foram incorporados no projeto, na construção, nos procedimentos de operação, na manutenção, no treinamento e na qualificação dos operadores. Mais precisamente, em 1994, a NRC emitiu o documento NUREG 711[2] estabelecendo critérios para implementar um programa de Engenharia de Fatores Humanos (EFH) em projetos de salas de controle, onde é avaliado o impacto causado pela introdução de interfaces operador sistema. Posteriormente em 1996, a NRC emitiu o documento NUREG 800 [3], o qual estabeleceu o conteúdo e o formato do capítulo 18 a ser incluído no relatório final de análise de segurança (FSAR), cujo objetivo é assegurar que as salas de controles das usinas nucleares sejam projetadas e avaliadas segundo os tópicos descritos pela NUREG 711.

Particularmente, o tópico referente a interfaces operador sistema apresentado pela NUREG 711, recomenda que a atividade do operador seja centrada nas tarefas de monitoração, tomada de decisão e controle da usina. Desta forma, sistemas que auxiliam objetivamente o operador no rápido entendimento do estado da usina, minimizando o tempo gasto com atividades secundárias como navegação entre telas e análise de dados, melhorariam o desempenho do mesmo na tarefa de gerenciamento da usina.

Entretanto, para sistemas que auxiliam o operador no diagnóstico de acidente, não basta apenas que o sistema seja rápido, a identificação também deve ser robusta e confiável com relação a ruído nos dados. Além disto, tendo em vista que uma central nuclear é uma instalação de segurança crítica, é de extrema importância que o sistema de identificação de transientes seja capaz de fornecer uma resposta “*não sei*” para transientes desconhecidos, isto é, não pertencentes ao conjunto de aprendizado, pois um diagnóstico errado pode trazer sérias conseqüências, tanto para pessoas quanto para a usina.

Neste trabalho é apresentado um sistema, de auxílio ao operador, para classificação de eventos que explora o bom desempenho obtido nas RNAs multicamadas com algoritmo de treinamento “backpropagation”[4-6], mesmo quando aplicados a dados ruidosos. Este sistema também possui um procedimento de validação de diagnóstico de modo a obter a saída “Não Sei” para os eventos fora do escopo de treinamento, evitando assim classificações incorretas ou não confiáveis. Uma outra característica importante do sistema é que ele se mostra independente da variável tempo, resultado

extremamente desejável pois dispensa a detecção do início do transiente que é um problema complexo e não totalmente resolvido na literatura.

O restante deste trabalho é organizado como a seguir. Seção 2 apresenta uma revisão de redes neurais. Seção 3 mostra a utilização desta RNA em sistemas dinâmicos ruidosos. Seção 4 descreve os procedimentos para identificação e validação do evento. Seção 5 apresenta uma aplicação deste método a um caso exemplo de transientes de uma usina nuclear, tipo PWR. Finalmente na seção 6 é feita a análise dos resultados e a conclusão do trabalho.

2. REDES NEURAIS ARTIFICIAIS

Redes Neurais Artificiais são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura cerebral e que adquirem conhecimento através da experiência. Uma RNA é composta por várias unidades de processamento cujo funcionamento é bastante simples. Esta unidade (neurônios artificiais) é geralmente conectada por canais de comunicação (sinapses) que são associados a determinados pesos. Os neurônios fazem operações apenas sobre seus dados locais, que são as entradas recebidas pelas suas conexões. Como no caso da rede biológica, o comportamento “inteligente” de uma rede neural artificial vem das interações entre os neurônios da rede. Estas operações de um neurônio artificial podem ser resumidas da seguinte maneira:

- Sinais são apresentados nas entradas;
- Cada sinal é ponderado pela sinapse;
- É feita a soma dos sinais ponderados produzindo um nível de atividade ;
- Em função do nível de atividade é produzida uma determinada resposta na saída.

Suponha que tenhamos p sinais de entrada, x_1, \dots, x_p e sinapses w_1, \dots, w_p , como mostrado na figura 2.1

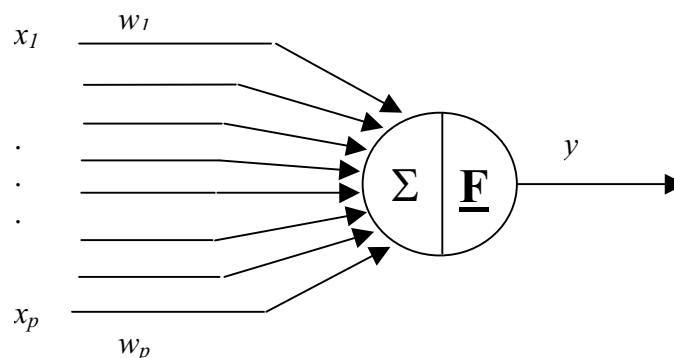


Figura 2.1 - Neurônio artificial

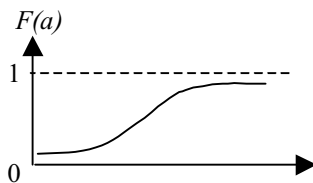
Assim temos:

$$a = \text{atividade} = \sum_{n=1}^p (w_n \cdot x_n)$$

$$y = \text{saída} = F(a)$$

A função F , utilizada para obter a saída do neurônio é chamada *função de ativação*. As funções de ativação mais utilizadas são do tipo sigmoidal, especialmente a logística e a tangente hiperbólica (Figura 2.2).

Logística: $F(a) = \frac{1}{1 + e^{-a}}$



Tangente hiperbólica: $F(a) = \text{tgh}(a)$

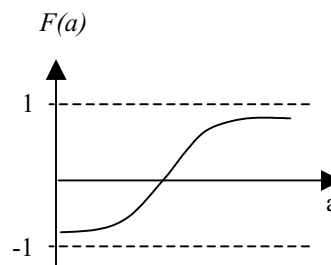


Figura 2.2. Funções de ativação

Usualmente as RNAs são organizadas em camadas divididas em três grupos básicos (fig 2.3):

- *Camada de entrada*: onde os padrões são apresentados à rede;
- *Camadas Intermediárias*: onde é feita a maior parte processamento por intermédio das sinapses que podem ser consideradas como extratores de características;
- *Camada de saída*: onde o resultado final é concluído e apresentado.

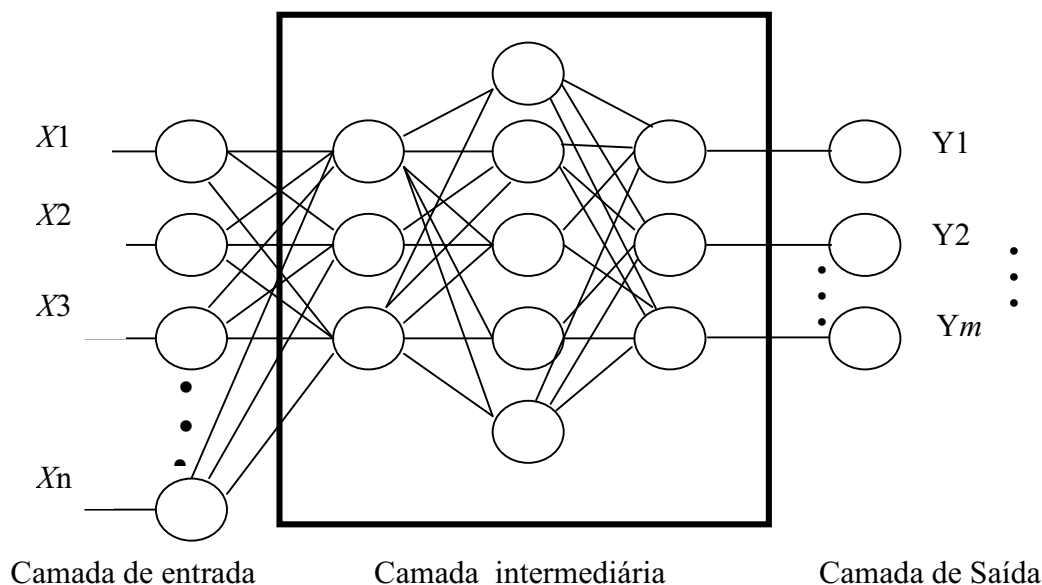


Figura 2.3 - Organização em Camadas

A propriedade mais importante das redes neurais é a habilidade de aprender a partir de seu ambiente e com isso melhorar seu desempenho. Isto é feito através de um processo iterativo de ajustes aplicado a suas sinapses chamado *treinamento*. Pode-se dizer que o aprendizado ocorre quando a rede neural atinge uma solução generalizada para uma classe de problemas. Existem diversos tipos de algoritmos de aprendizado específicos para determinados modelos de redes neurais. Estes algoritmos diferem entre si, principalmente pelo modo como as sinapses são modificadas. Em relação ao aprendizado podemos destacar dois paradigmas: (1) aprendizado Supervisionado, quando é utilizado um agente externo que indica à rede a resposta desejada para o padrão de entrada; (2) aprendizado Não Supervisionado (Auto-Organizável), quando não existe o agente externo indicando a resposta desejada para o padrão de entrada.

Um método de aprendizado supervisionado utilizado com sucesso em vários tipos de RNAs é o “backpropagation”. A seguir, serão apresentadas algumas considerações importantes a respeito deste algoritmo, em particular sua aplicação no tipo de rede a ser utilizada neste trabalho.

A figura 2.4 mostra a arquitetura de uma RNA multicamadas tipo “jump”, com uma camada de entrada, duas camadas intermediárias e uma camada de saída. Para este tipo de RNA, um neurônio de qualquer camada da rede é conectado a todos os outros neurônios das camadas posteriores (não há realimentação). O fluxo de sinal através desta rede se propaga “para frente”, da esquerda para direita, passando por todas as camadas.

A figura 2.5 mostra uma parte da rede “jump”. Dois tipos de sinais são identificados nesta rede: (1) o “sinal de estímulo”, que é o sinal de entrada vindo da primeira camada da rede, se propagando “para

frente” (neurônio por neurônio) através da rede, emergindo na saída da rede como sinal de saída; (2) “sinal de erro”, que é o erro originário em um neurônio de saída da rede, e propagado de volta (neurônio por neurônio) através da rede.

Cada neurônio da rede é designado para realizar duas tarefas: (1) computar o sinal estímulo oriundo da saída de um neurônio, o qual está expressando uma função não linear do sinal de entrada, e pondera-lo pelo peso da sinapse associada àquele neurônio; (2) computar a estimativa do vetor gradiente (isto é, o gradiente da função erro, relativo às ponderações das sinapses conectadas às entradas de um neurônio) , a qual é necessária para a etapa de retorno através da rede.

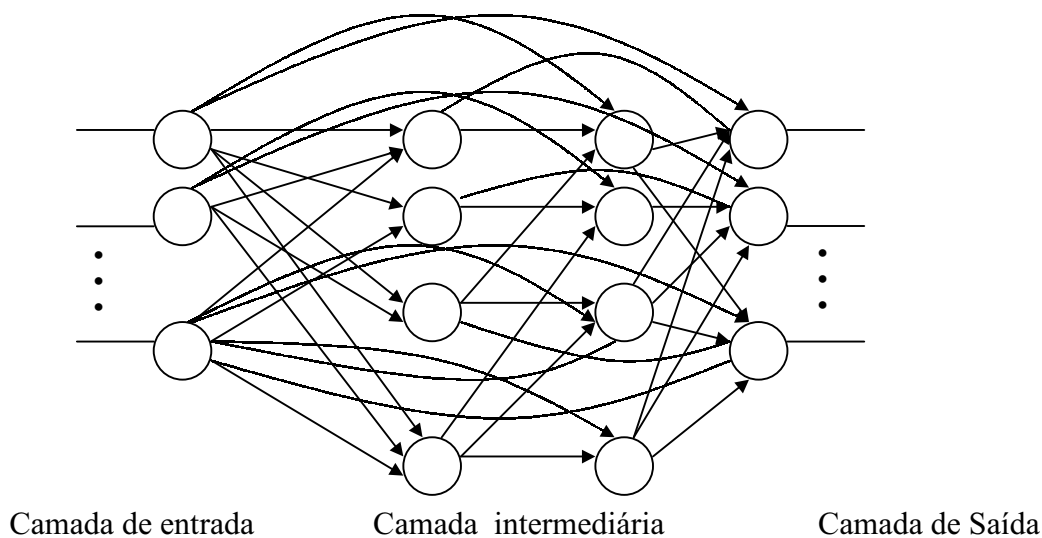


Figura 2.4 - RNA Jump

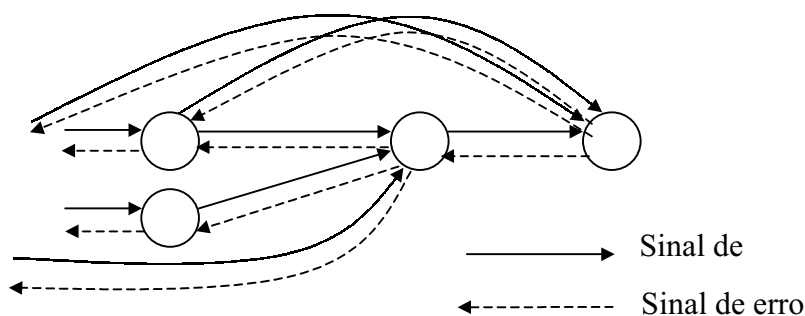


Figura 2.5 - Ilustração das duas direções do fluxo do sinal

Seja e_j o sinal de erro na saída de um neurônio j , da última camada, depois de interações (isto é, depois da n -ésima apresentação de um padrão de treinamento), definidas por:

$$e_j(n) = d_j(n) - y_j(n), \quad e_j(n) = d_j(n) - y_j(n) \quad d_j \text{ é o valor esperado do neurônio } j \text{ da camada de saída}$$

Definindo também o erro quadrático instantâneo como sendo:

$$\xi(n) = \frac{1}{2} \sum e_j^2(n), \quad \text{onde o somatório inclui todos os neurônios da camada de saída.}$$

O erro quadrático instantâneo $\xi(n)$, é uma função de todos os parâmetros livres da rede (isto é, dos pesos das sinapses w_{ij}). Para um dado conjunto de treinamento, $\xi(n)$ representa a função custo como uma forma de medir o desempenho do aprendizado. O objetivo, então, do processo de aprendizado é ajustar os parâmetros livres da rede de forma a minimizar $\xi(n)$. Desta forma, o algoritmo “backpropagation” aplica uma correção $\Delta w_{ji}(n)$ no valor da sinapse $w_{ji}(n)$, de forma proporcional à derivada parcial $\frac{d\xi(n)}{dw_{ji}}$. Pela

regra da cadeia, pode-se expressar este gradiente como sendo:

$$\frac{d\xi(n)}{dw_{ji}} = \frac{d\xi(n)}{de_j(n)} \cdot \frac{de_j(n)}{dy_i(n)} \cdot \frac{dy_i(n)}{da_i(n)} \cdot \frac{da_i(n)}{dw_{ji}} \quad (2.1)$$

A derivada parcial $\frac{d\xi(n)}{dw_{ji}}$ representa o fator de sensibilidade, determinando a direção de busca dentro do espaço valores para os pesos das sinapses.

Após alguns algebrismos, obtém-se para $\frac{d\xi(n)}{dw_{ji}}$ a seguinte expressão:

$$\frac{d\xi(n)}{dw_{ji}} = -e_j(n) \cdot \frac{dF(a_j(n))}{da_j(n)} \cdot y_j(n) \quad (2.2)$$

A correção $\Delta w_{ji}(n)$ aplicado à sinapse $w_{ji}(n)$ é definido pela regra delta como sendo:

$$\Delta w_{ji}(n) = -\eta \frac{d\xi(n)}{dw_{ji}}, \quad \text{onde } \eta \text{ é um parâmetro que representa a taxa de aprendizado do algoritmo}$$

“backpropagation”. (2.3)

Definido o gradiente local $\delta_j(n)$ como sendo:

$$\begin{aligned} \delta_j(n) &= -\frac{d\xi(n)}{da_j(n)} \\ &= \frac{d\xi(n)}{de_j(n)} \cdot \frac{de_j(n)}{dy_j(n)} \cdot \frac{dy_j(n)}{da_j(n)} \end{aligned} \quad (2.4)$$

$$= -e_j(n) \cdot \frac{dF(a_j(n))}{da_j(n)} \quad (2.5)$$

Aplicando 2.5 a 2.2 e posteriormente a 2.3, tem-se:

$$\Delta w_{ji}(n) = -\eta \cdot \delta_j(n) \cdot y_i(n) \quad (2.6)$$

O gradiente local é necessário para alterar os pesos das sinapses. De acordo com a Eq. (2.5), o gradiente local $\delta_j(n)$ para o neurônio de saída é igual ao produto do erro, $e_j(n)$, pela derivada da função de ativação $\frac{dF(a_j(n))}{da_j(n)}$.

Generalizando a equação acima 2.4 para qualquer neurônio das camadas anteriores, obtém-se:

$$\delta_j(n) = \frac{dF(a_j(n))}{da_j(n)} \cdot \sum \delta_k(n) \cdot w_{kj}(n), \text{ onde } k \text{ representa qualquer neurônio das camadas posteriores a camada do neurônio } j. \quad (2.7)$$

Aplicando 2.7 a 2.6 tem-se a regra geral de atualização das sinapses pelo algoritmo “backpropagation”:

$$\Delta w_{ji}(n) = \eta * \delta_j(n) * y_i(n) \quad (2.8)$$

Fator de Correção das sinapses	Taxa de aprendizado	Gradiente local	Sinal de entrada do neurônio j, oriundo do neurônio i
$\Delta w_{ji}(n)$	η	$\delta_j(n)$	$y_i(n)$
=	•	•	

A grande diferença entre a rede “jump” utilizada neste trabalho e uma rede feedforward, em relação ao processo de treinamento pelo algoritmo “backpropagation”, é o fato que o termo de gradiente local $\delta_j(n)$, para a rede “jump”, sofre influência direta de todos os neurônios das camadas subsequentes à camada do neurônio j . Esta influência direta faz com que as alterações das sinapses das primeiras camadas da rede sejam mais sensíveis aos erros apresentados na saída da rede, possibilitando assim que a rede “jump” funcione melhor ao separar padrões em espaços muito complexos, que é o caso do espaço das variáveis de estado durante alguns eventos anormais.

3. REDES NEURAIS ARTIFICIAIS EM SISTEMAS DINÂMICOS RUIDOSOS

A RNA apresentada na seção anterior, tem uma boa capacidade de generalização. Entretanto, este tipo RNA, por não ter recorrência em sua estrutura interna (onde existe uma dependência direta com o tempo), não são aptas a trabalharem diretamente com sistemas dinâmicos. Desta forma, objetivando superar esta limitação, adotou-se o sistema de “janela temporal móvel”, ou seja, incorporou-se à arquitetura externa da rede a dependência temporal, através da apresentação seqüencial no tempo do histórico recente das variáveis de estado utilizadas na identificação dos eventos, figura 3.1. Para definir o número de elementos desta “janela”, deve-se considerar que esta deve ser grande o suficiente para identificar o comportamento dinâmico do sistema, porém sem comprometer a identificação de eventos rápidos.

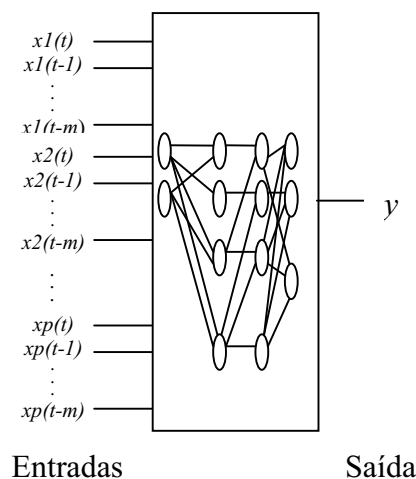


Figura 3.1 RNA com janela temporal

De modo a aumentar a robustez desta rede ao ruído, é necessário que uma grande quantidade de padrões sejam apresentados à rede na fase de treinamento, isto é feito adicionando exemplos de padrões com ruídos superpostos e forçando, através do treinamento, que a rede reconheça estes novos exemplos

como pertencendo as respectivas classes originais (sem ruído). Desta forma, foi adicionado a cada variável de estado, um ruído branco com distribuição normal e três desvios padrão igual a 1% do valor máximo da respectiva variável. Este valor de 1% é bem razoável se comparado aos valores reais apresentados na usina nuclear PWR, que se encontraram com valores bem inferiores a 1%.

4. CRITÉRIOS PARA IDENTIFICAÇÃO E VALIDAÇÃO DO EVENTO

A arquitetura da RNA utilizada no primeiro critério de identificação e validação, é do tipo apresentada na figura 3.1, utilizando como entrada o histórico recente de algumas das variáveis de processo da usina. Como este tipo de arquitetura permite que o valor apresentado na sua saída possa assumir valores contínuos, e por outro lado, o sistema de classificação de eventos adotado representa cada um dos eventos por um valor inteiro discreto (isto é, evento A é representado pelo valor 1, evento B é representado pelo valor 2, e assim sucessivamente para todos os eventos previstos), os valores apresentados na saída da rede para cada evento, oscilavam em torno do valor discreto esperado. Desta forma, foi determinado o raio da zona de alcance (R_{ev}) de cada evento como sendo o maior desvio entre o valor esperado e o valor apresentado na saída da rede para o respectivo evento.

$$R_{ev} = \text{Máximo} | y_{ev} - \hat{y}_{ev} |, \text{ } ev \text{ indica o evento, } y_{ev} \text{ é o valor esperado, } \hat{y}_{ev} \text{ é o valor obtido pela RNA} \quad (4.1)$$

Uma vez que a resposta da RNA se torna imprevisível quando deparada com entradas representativas de eventos fora do escopo de treinamento, fazendo com que este tipo de rede não seja capaz de obter uma saída “não sei” para estes eventos, torna-se então, imprescindível definir um procedimento de validação para a resposta dada pela rede. Assim sendo, foram definidos dois critérios para o procedimento de validação.

O primeiro critério é definido como critério de identificação e validação da rede (CIVR). Segundo este critério, o evento identificado (y), será aquele que mais se aproximar do valor obtido na saída da RNA \hat{y} , e este evento será considerado válido se, e somente se, o desvio absoluto entre estes dois valores $|Y - \hat{Y}|$ for igual ou menor ao R_{ev} do evento.

O segundo critério de validação, denominado critério de validação das variáveis (CVV), é complementar ao primeiro e tem por objetivo validar pontualmente todas as variáveis de estado utilizadas como entradas da rede. Para este fim um simulador neural auto associativo é utilizado para estimar o valor de cada uma das variáveis de estado. Este simulador estima o valor das variáveis, tendo como entradas os valores medidos das próprias variáveis de processo e o tipo de evento identificado pelo CIVR, figura 4.1 .

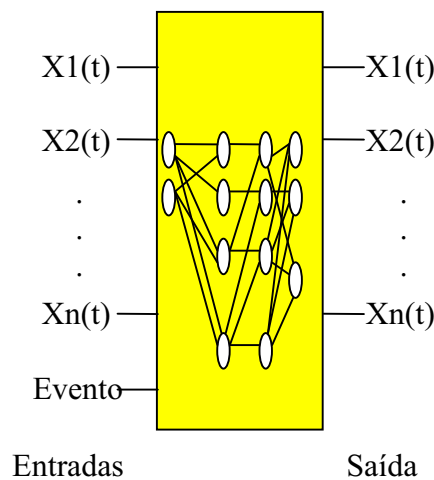


Figura 4.1 Simulador auto-associativo

Da mesma forma que a RNA do CIVR, a rede do simulador também foi treinada com uma grande quantidade de padrões, durante esta fase é determinado, para cada variável de estado, o maior desvio Δx_{max} , entre o valor esperado da variável e o valor apresentado na saída da rede. Desta forma, durante a operação, uma variável é dita válida se, e somente se, o valor absoluto da diferença entre o valor da variável $X(t)$ e o valor de referência para esta mesma variável $X_{ref}(t)$, gerada pelo simulador neural, for igual ou menor ao Δx_{max} da respectiva variável. Finalmente, segundo o CVV, O diagnóstico é dito válido, se todas as variáveis também forem válidas. A figura 4.2 abaixo, apresenta o diagrama de blocos do sistema de identificação e validação de eventos.

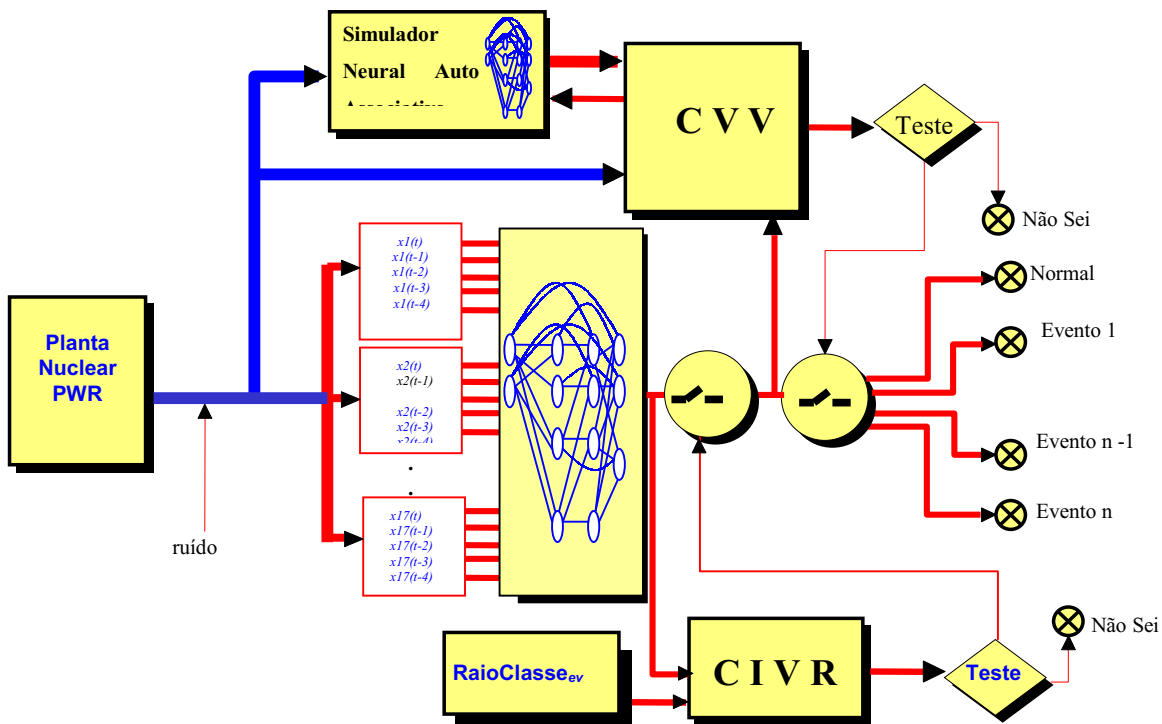


Figura 4.2 Diagrama de blocos do sistema

Definidos os critérios CIVR e CVV, pode-se descrever o procedimento para identificação e validação de eventos através do algoritmo apresentado a seguir em forma de fluxograma, figura 4.3.

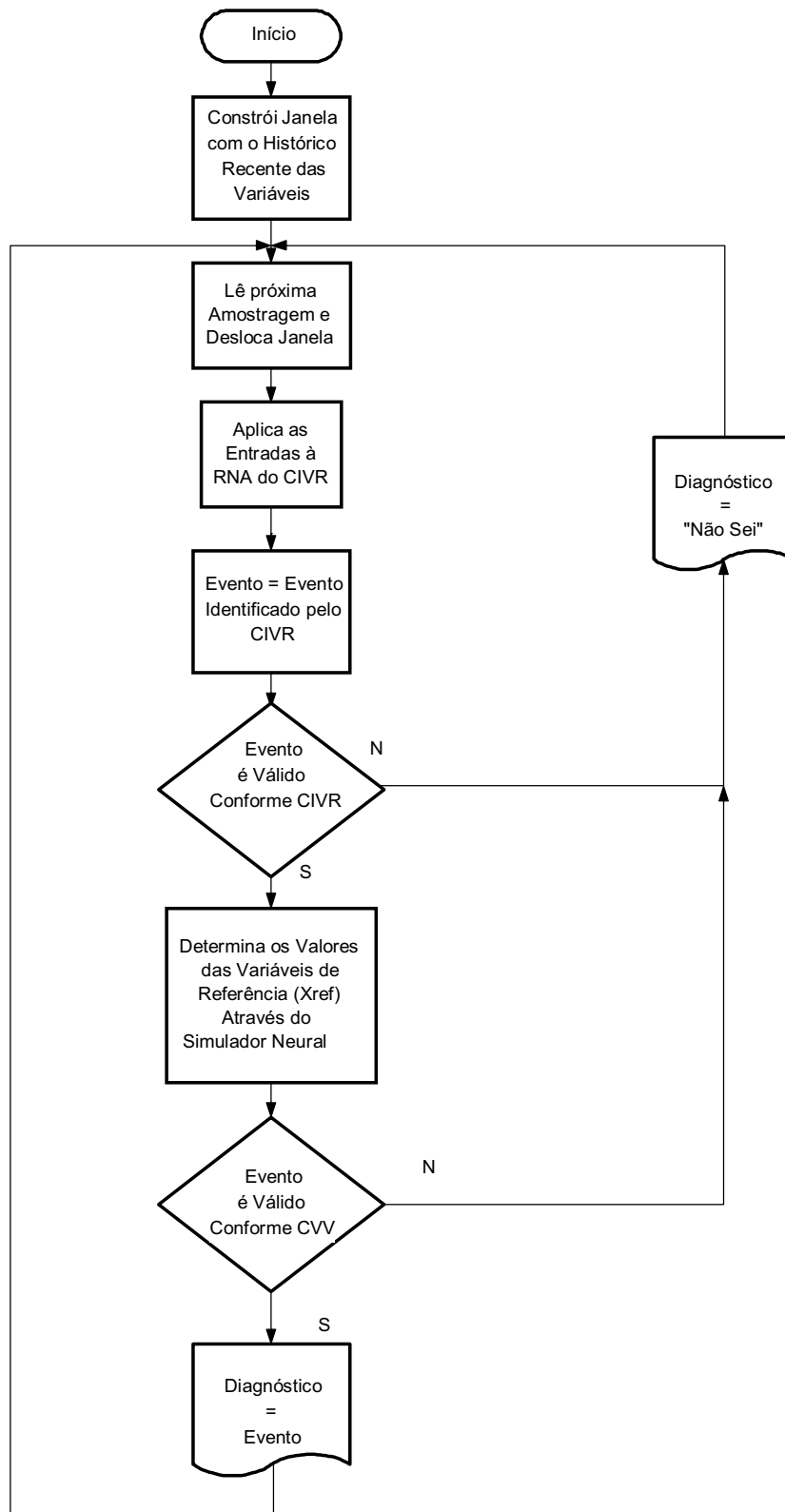


Figura 4.3 Fluxograma dos critérios de identificação e validação

5. APLICAÇÃO DO MÉTODO NA IDENTIFICAÇÃO DE TRANSIENTES

Nesta seção o método descrito anteriormente é aplicado na identificação de transientes pertencentes ao conjunto de acidentes postulados para a central nuclear ANGRA-II, tipo PWR, gerados por simulação computacional, operando a uma potência de 100%. Desta forma, além da condição de normalidade, foram gerados quinze transientes: (a) perda de alimentação elétrica externa (BLACKOUT); (b) perda de refrigerante do sistema primário (LOCA); (c) ruptura de tubos do gerador de vapor (SGTR); (d) ruptura da alimentação principal (MFWBR); (e) desligamento do reator e da turbina (TRIP); (f) isolamento da alimentação principal e auxiliar (MEFWISO); (g) isolamento da alimentação principal (MFWISO); (h) isolamento da linha de vapor principal (MSTMISO); (i) ruptura da linha de vapor principal (STMLIBR); (j) perda de alimentação elétrica sem SCRAM (BLACKSEM); (k) ruptura de alimentação principal sem SCRAM (MFWBRSEM); (l) isolamento da linha de vapor principal sem SCRAM (MSTMISEM); (m) isolamento da alimentação principal e auxiliar sem SCRAM (MEFWISEM); (n) isolamento da alimentação principal sem SCRAM (MFWISEM); (o) desligamento da turbina sem SCRAM (TRIPTUR). Cada um desses transiente foi descrito por valores de 17 variáveis de estado, tabela 5.1, amostrada em intervalos de 1 segundo, totalizando 60 segundos de simulação.

Tabela 5.1- Variáveis de estado utilizadas no cenário de teste.

Número da Variável	Variável de processo	Unidade
1	Vazão no núcleo	%
2	Temperatura na perna quente	°C
3	Temperatura na perna fria	°C
4	Vazão no núcleo	Kg/s
5	Nível no gerador de vapor – Faixa larga	%
6	Nível no gerador de vapor – Faixa estreita	%
7	Pressão no gerador de vapor	MPa
8	Vazão de água de alimentação	Kg/s
9	Vazão de vapor	Kg/s
10	Vazão na ruptura	Kg/s
11	Vazão no circuito primário	Kg/s
12	Pressão no sistema primário	MPa
13	Potência térmica	%
14	Potência Nuclear	%
15	Margem de sub-resfriamento	°C
16	Nível do pressurizador	%
17	Temperatura média no primário	°C

Os transientes simulados foram divididos em dois grupos em função da ocorrência ou não do sinal de SCRAM. O primeiro grupo é constituído dos acidentes onde ocorreu o SCRAM, ou seja: BLACKOUT, LOCA, SGTR, MFWBR, TRIP, MEFWISO, MFWISO, MSTMISO e STMLIBR. Consequentemente, os demais acidentes, juntamente com a condição de normalidade, formaram o segundo grupo, ou seja: BLACKSEM, MFWBRSEM, MSTMISEM, MEFWISEM, MFWISEM, TRIPTURSEM. Esta divisão possibilitou que o sistema de diagnóstico fosse dividido em dois subsistemas independentes, especializando cada um deles a um dos grupos anteriores.

Definidas as variáveis de entrada e os dois grupos de acidentes formou-se, para cada grupo, o conjunto de treinamento para a RNA do CIVR. O primeiro conjunto de treinamento foi constituído de 480 padrões sem ruído e 2400 padrões com ruído, representando todos os acidentes do primeiro grupo, exceto o acidente MFWBR que servirá como evento desconhecido para a verificação de classificações "não sei". O segundo conjunto de treinamento foi constituído de 360 padrões sem ruído e 1800 padrões com ruído, representando todo primeiro grupo, exceto o acidente MFWBRSEM que servirá como evento desconhecido para a verificação de classificações "não sei".

Para definir todos os parâmetros da RNA, foi necessário realizar vários ensaios, uma vez que não existe um critério geral, bem definido, para escolha destes parâmetros. Após estes ensaios, a RNA que apresentou melhor resultado, de acordo com cada grupo, obteve a seguinte configuração:

Primeiro grupo:

- camada de entrada constituída de 85 neurônios com funções de ativação linear (17 “janelas temporais móveis” com cinco elementos cada);
- 2 camadas intermediárias com 34 neurônios com funções de ativação do tipo logística;
- camada de saída com 1 neurônio com função de ativação do tipo logística.

Segundo grupo:

- camada de entrada constituída de 85 neurônios com funções de ativação linear (17 “janelas temporais móveis” com cinco elementos cada);
- 2 camadas intermediárias com 18 neurônios com funções de ativação do tipo logística;
- camada de saída com 1 neurônio com função de ativação do tipo logística.

Após o treinamento das RNAs, foram obtidos os raios da zona de alcance (R_{ev}) para os acidentes de cada um dos grupos, tabelas 5.2 e 5.3 .

Tabela 5.2- Raio de alcance de cada acidente do primeiro grupo

Acidente	Codificação da RNA	R_{ev}
BLACKOUT	2	0,1
LOCA	3	0,09
SGTR	4	0,1
MEFWISO	5	0,1
MFWISO	6	0,14
MSTMISO	7	0,14
STLIBR	8	0,16
TRIPTUR	9	0,05

Tabela 5.3 - Raio de alcance de cada acidente do segundo grupo

Acidente	Codificação da RNA	R_{ev}
NORMAL	1	0,035
BLACKSEM	2	0,069
MEFWISEM	3	0,15
MFWISEM	4	0,11
MSTMISEM	5	0,08
TRIPTURSEM	6	0,17

Da mesma forma que formou-se dois grupos de treinamento para as RNAs do CIVR, também foram formados dois grupos de treinamento para as RNAs do simulador, constituídos com o mesmo número de padrões e para os mesmos acidentes. Assim sendo, após algumas simulações obteve-se as seguintes configurações para as RNAs:

Primeiro grupo:

- camada de entrada constituída de 18 neurônios com funções de ativação linear (17 variáveis de estado e a saída da RNA do CIVR);
- 2 camadas intermediárias com 20 neurônios com funções de ativação do tipo logística;

- camada de saída com 17 neurônios com função de ativação do tipo logística.

Segundo grupo:

- camada de entrada constituída de 18 neurônios com funções de ativação linear (17 variáveis de estado e a saída da RNA do CIVR);
- 2 camadas intermediárias com 15 neurônios com funções de ativação do tipo logística;
- camada de saída com 17 neurônios com função de ativação do tipo logística.

Após o treinamento das RNAs dos simuladores, foram obtidos os desvios máximos de cada variável (Δx_{max}) para os acidentes de cada um dos grupos, tabelas 5.4 e 5.5 .

Tabela 5.4 – Valores dos desvios máximos Δx_{max} do primeiro grupo

Variável \ Acidente	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
BLACKOUT	1.8	1.1	1.2	132	1.3	0.5	0.1	5.0	10	1.3	2.0	0.1	0.5	0.6	0.4	0.6	0.5
LOCA	0.3	0.6	0.8	217	0.2	0.2	0.1	4.0	2.7	3.5	0.4	0.2	0.3	0.3	0.3	1.3	0.5
SGTR	0.5	1.5	1.3	639	0.4	0.3	0.1	2.9	6.0	2.8	0.3	0.2	0.4	0.3	0.2	0.9	0.7
MEFWISO	0.7	2.3	0.8	461	0.7	0.7	0.1	22	14	1.5	0.7	0.1	1.0	0.9	0.3	1.0	1.2
MFWISO	0.8	1.5	0.7	316	0.6	0.8	0.5	6.1	11	1.8	0.5	0.1	0.5	0.5	0.3	0.4	0.9
MSTMISO	0.9	2.1	1.0	274	1.3	0.5	0.1	11	25	1.1	0.9	0.1	0.8	0.5	0.4	0.4	1.2
STLIBR	0.3	0.6	1.2	99	0.5	0.2	0.1	7.6	3.1	0.9	0.3	0.1	0.4	0.4	0.1	0.5	0.6
TRIPTUR	0.4	0.5	0.2	186	0.7	0.4	0.1	11	8.3	1.2	0.3	0.1	0.4	0.2	0.4	0.3	0.3

Tabela 5.5 – Valores dos desvios máximos Δx_{max} do segundo grupo

Variável \ Acidente	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
NORMAL	0.2	0.5	0.9	289	0.3	0.2	0.1	2.6	5.1	0.1	0.3	0.1	0.2	0.2	0.1	0.3	0.8
BLACKSEM	1.9	1.1	1.5	334	1.2	0.6	0.1	12	39	0.1	1.4	0.1	2.3	2.4	0.7	0.8	1.5
MEFWISEM	0.5	0.6	1.5	434	1.0	0.9	0.1	19	11	0.1	0.5	0.2	0.4	0.3	0.6	0.5	0.8
MFWISEM	0.4	0.7	1.0	394	1.0	0.5	0.1	6.1	9.4	0.1	0.4	0.1	0.3	0.3	0.3	0.5	0.3
MSTMISEM	0.5	0.8	2.5	448	0.9	0.8	0.1	8.4	18	0.1	0.6	0.1	0.6	0.5	0.4	0.7	1.3
TRIPTURSEM	0.4	0.6	1.0	382	0.9	0.3	0.1	6.1	45	0.1	0.5	0.1	0.4	0.4	0.2	0.6	0.7

O teste final do sistema foi realizado em duas etapas, onde cada uma delas visa avaliar a resposta do sistema para cada um dos grupos. Desta forma, foi gerado para os dois grupos um total de 840 novos padrões representativos dos eventos incluídos na fase de treinamento (adicionando, para cada acidente, novos ruídos às variáveis de estado) e 120 padrões representativos dos eventos "MFWBR" e MFWBRSEM que não foram incluídos nos treinamentos. Para cada evento do primeiro grupo foi utilizado um conjunto de 60 padrões, onde todos os padrões representam a condição de acidente, a figura 5.1 apresenta os gráficos das respostas da RNA do CIVR para esse grupo e a tabela 5.6 apresenta a resposta do sistema. Para cada evento do segundo grupo também foi utilizado um conjunto de 60 padrões, entretanto os dez primeiros padrões deste conjunto representam condição de normalidade e os últimos 50 padrões representam a condição de acidente, a figura 5.2 apresenta os gráficos das respostas da RNA do CIVR para esse grupo e a tabela 5.7 apresenta a resposta do sistema.

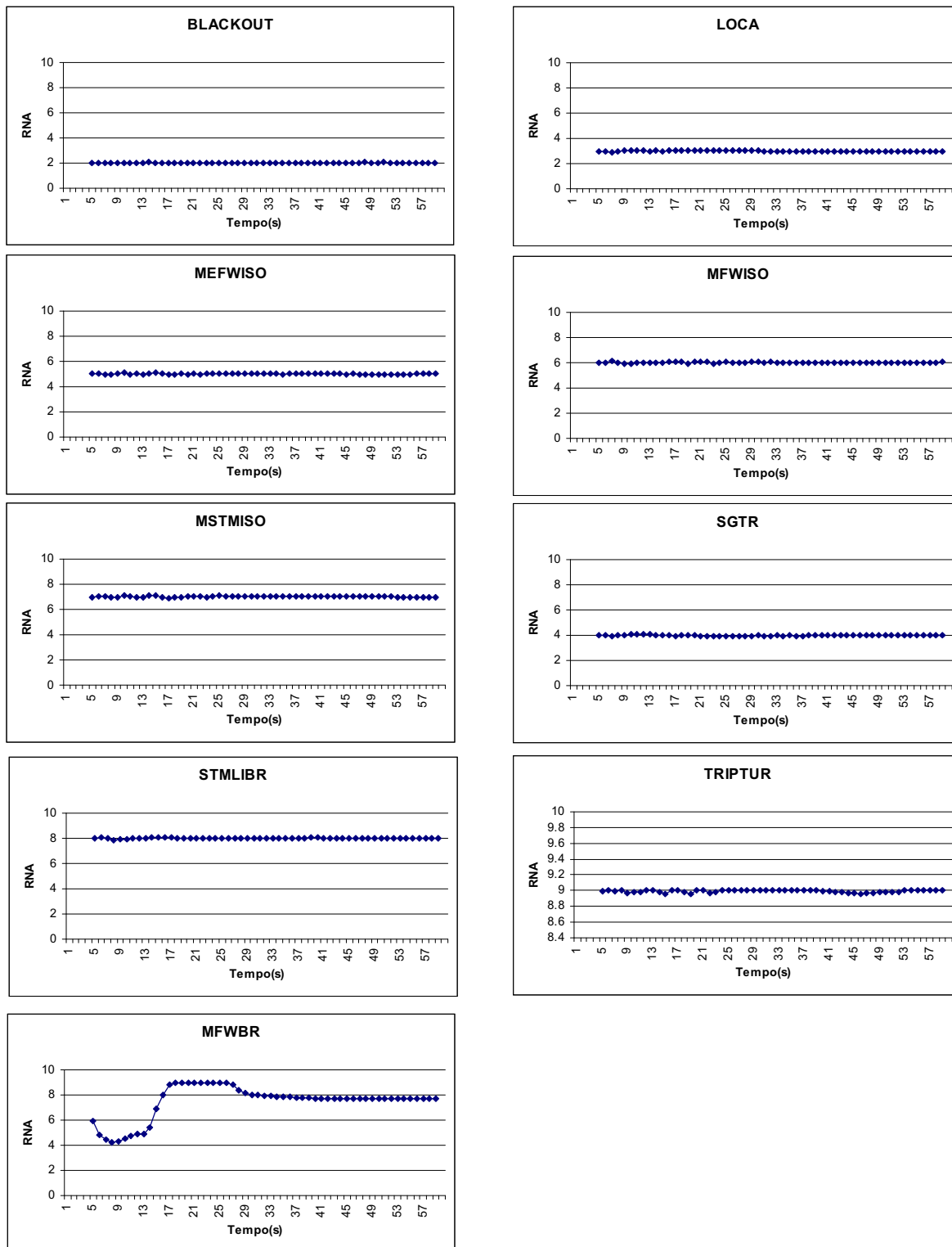


Figura 5.1 Respostas da RNA do CIVR para o primeiro grupo

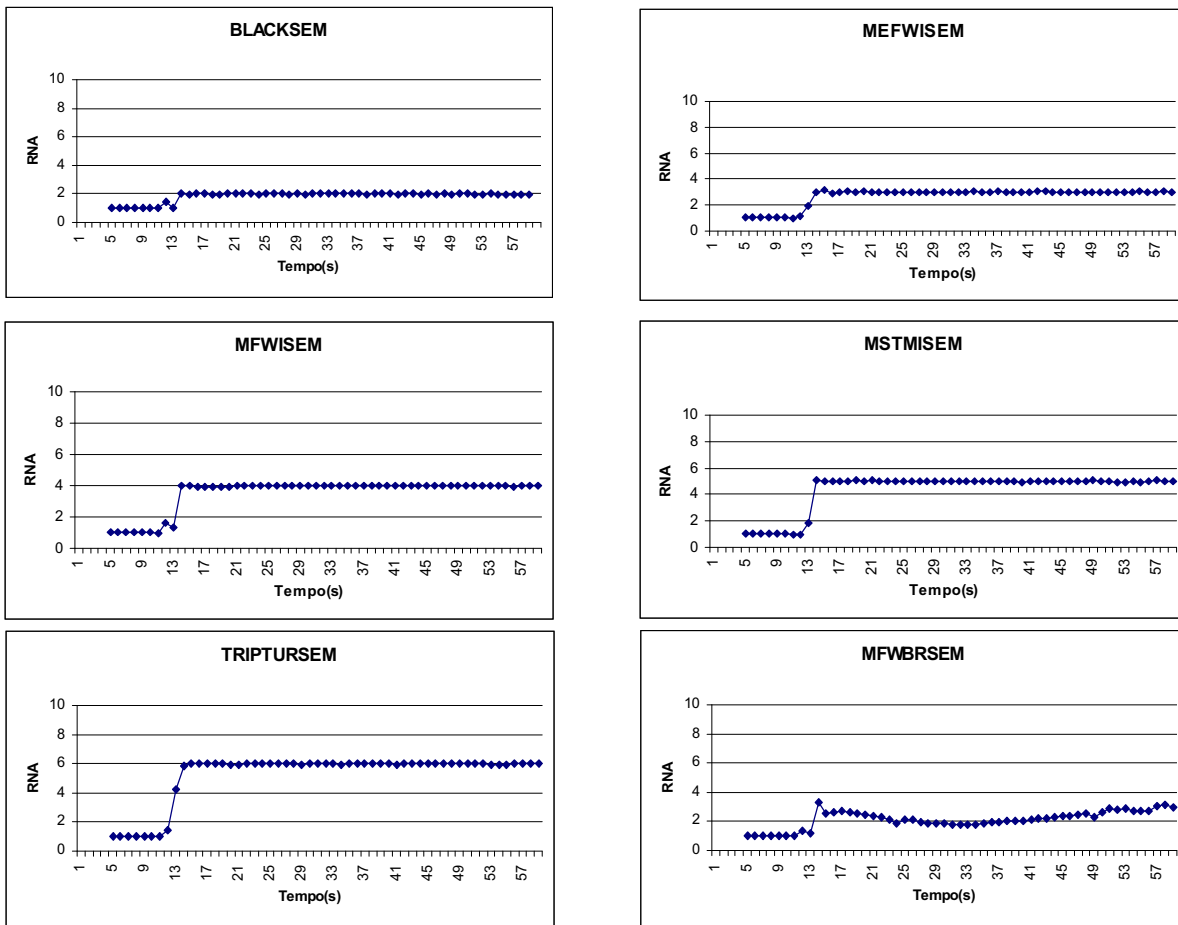


Figura 5.2 Respostas da RNA do CIVR para o segundo grupo

6. CONCLUSÕES

Neste trabalho foi apresentado um sistema para identificação de eventos, baseado RNAs, com o objetivo de auxiliar o operador na tomada de decisão no procedimento a ser seguido em situações de acidentes/transientes em centrais nucleares.

Através dos resultados apresentados nas tabelas 5.6 e 5.7, verificou-se que a RNA do tipo “jump”, com treinamento “backpropagation”, foi capaz de diagnosticar rapidamente todos os acidentes postulados para um reator nuclear PWR, mesmo com inclusão de ruído. Particularmente a tabela 5.6 juntamente com os gráficos apresentados na figura 5.1 demonstraram que a RNA do CIVR foi capaz de identificar todos os acidentes do primeiro grupo, assim que a primeira “janela temporal” foi formada (cinco primeiros segundos do acidente). Para os acidentes do segundo grupo, a tabela 5.7 e os gráficos apresentados na figura 5.2 demonstraram que a RNA do CIVR, também foi capaz de tratar com sucesso, para todos os acidentes do segundo grupo, a transição da condição de normalidade para condição de acidente, identificando o acidente, no pior caso, em 4 segundos após o início do mesmo.

Os resultados das tabelas também demonstraram que o sistema proposto contornou, com sucesso, através do critério de validação das variáveis (CVV), a grande limitação desse tipo rede, que é incapacidade de fornecer uma resposta “não sei” confiável para eventos fora do escopo do treinamento.

Um outro ponto importante verificado neste trabalho, é que o sistema se mostrou independente de um sinal que indique o início do evento tornando-o, assim, robusto em relação a esta limitação.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] **NUREG 737 Supplement 1, 1983**. Clarification of TMI Action Plan Requirements; Requirements for Emergency Response Capability.
- [2] **NUREG 711, 1994**. Human Factors Engineering Program Review Model
- [3] **NUREG 800, Draft Revision 0, 1996**. Standard Review Plan for The Review of Safety Analysis Reports for Nuclear Power Plants
- [4] S. W. CHEON AND S. CHANG, “Aplication of Neural Networks to a Connectionist Expert System for Transient Identification in Nuclear Power Plants”, **Nuclear Technology**, v. 102, pp. 177-186, 1993.
- [5] Y. OHGA AND H. SEKI, “Abnormal Event Identification in Nuclear Power Plants Using Neural Network and Knowledge Processing”, **Nuclear Technology**, v. 101, pp. 159-16., Feb, 1993
- [6] A. G. PARLOS, et al, “Nonlinear Identification of Process Dynamics Using Neural Networks”, **Nuclear Technology**, v 97, pp 79-94, Jan. 1992.