



COPPE/UFRJ

**SISTEMA COGNITIVO DE SUPORTE AO OPERADOR DE UMA USINA
NUCLEAR BASEADO EM MÓDULOS CONTEXTUAIS**

Herculano Vieira Soares

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Nuclear, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Ciências em Engenharia Nuclear.

Orientador: Roberto Schirru

Rio de Janeiro

Março/2009

SISTEMA COGNITIVO DE SUPORTE AO OPERADOR DE UMA USINA NUCLEAR
BASEADO EM MÓDULOS CONTEXTUAIS

Herculano Vieira Soares

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ
COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA
UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM
ENGENHARIA NUCLEAR.

Aprovada por:

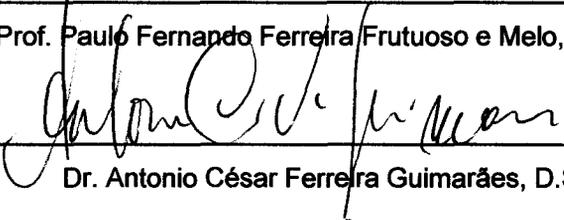


Prof. Roberto Schirru, D.Sc.

Prof. José Antonio Carlos Canedo Medeiros, D.Sc.



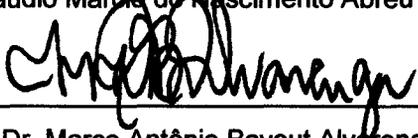
Prof. Paulo Fernando Ferreira Frutuoso e Melo, D.Sc.



Dr. Antonio César Ferreira Guimarães, D.Sc.



Dr. Cláudio Márcio do Nascimento Abreu Pereira, D.Sc.



Dr. Marco Antônio Bayout Alvarenga, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

MARÇO DE 2009

Soares, Herculano Vieira

Sistema Cognitivo de Suporte ao Operador de uma Usina Nuclear Baseado em Módulos Contextuais/Herculano Vieira Soares. – Rio de Janeiro: UFRJ/COPPE, 2009.

XI, 202 p:il.;29,7 cm.

Orientador: Roberto Schirru

Tese (Doutorado) – UFRJ/COPPE/Programa de Engenharia Nuclear, 2009.

Referências Bibliográficas: p. 93-96

1. Sistemas de suporte híbridos. 2. Sistemas especialistas. 3. Sistemas neuro-nebulosos. I. Schirru, Roberto. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Nuclear. III. Título

AGRADECIMENTOS

Ao meu orientador acadêmico Professor Roberto Schirru, mestre e professor.

Ao Doutor Marco Antônio Bayout Alvarenga. Sem sua participação este trabalho teria sido impossível.

Ao meu amigo José Ribeiro de Souza pela paciência e longas horas gastas no simulador para a obtenção dos dados dos acidentes.

A minha amiga Regina Célia Abossamra Alvarenga pela paciência e compreensão.

A meus amigos e colegas da CNEN pelo incentivo e apoio.

A minha esposa Keli pelo incentivo e apoio.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências em Engenharia Nuclear (D.Sc.)

**SISTEMA COGNITIVO DE SUPORTE AO OPERADOR DE UMA USINA NUCLEAR
BASEADO EM MÓDULOS CONTEXTUAIS**

Herculano Vieira Soares

Março/2009

Orientador: Roberto Schirru

Programa: Engenharia Nuclear

Este trabalho apresenta um conceito de sistema de suporte à operação de uma usina nuclear, composto de dois módulos, referentes aos contextos familiares e não familiares ao operador. Este sistema é construído sobre um modelo cognitivo simplificado do operador. A parte referente aos contextos familiares ao operador é constituída de um sistema especialista simbólico que contém as regras tipo SE-ENTÃO extraídas dos procedimentos de operação em emergência. É adotado o conceito de Tarefas de Segurança que reúne as ações sequenciais dos procedimentos que possuem um objetivo em comum. A parte referente aos contextos não familiares ao operador é constituída de um sistema especialista híbrido neuro-nebuloso. Este módulo é treinado com dados referentes a sete acidentes típicos de uma usina tipo PWR obtidos por um simulador gráfico. Um protótipo do sistema, integrando esses dois módulos, é construído num ambiente computacional especialmente projetado para dar suporte à modelagem cognitiva, denominado COGENT. O sistema é testado e os resultados obtidos permitem concluir pela sua viabilidade.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requisites for the degree of Doctor of Sciences in Nuclear Engineering (D.Sc.)

**COGNITIVE NUCLEAR POWER PLANT OPERATOR SUPPORT SYSTEM BASED
ON CONTEXTUAL MODULES**

Herculano Vieira Soares

March/2009

Advisor: Roberto Schirru

Department: Nuclear Engineering

This work presents a concept of a nuclear power plant operator support system composed of two modules relative to operator familiar and non familiar contexts. The system is built over a simplified operator cognitive model. The familiar context module comprises a symbolic expert system containing IF-THEN type rules obtained from the emergency operating procedures. It is adopted the concept of Safety Tasks which gathers the sequential instructions which have a common safety objective. The non familiar context module comprises a hybrid neuro-fuzzy expert system. This module is trained with data from seven PWR nuclear power plant typical accidents, obtained from a graphical simulator. A prototype of the system, integrating these two modules, is built in a computational environment designed to support cognitive modeling, named COGENT. The system is tested and the results allow us to conclude for its viability.

INDICE

1.	INTRODUÇÃO	
1.1	Apresentação	1
1.2	Conceito de Tarefas do Operador Durante um Acidente	6
1.3	Sistemas de Suporte	9
1.4	Avaliação da Bibliografia sobre o Tema	11
1.5	Objetivos e Organização do Trabalho	14
2.	FUNDAMENTOS TEÓRICOS	
2.1	Sistemas Especialistas	17
2.2	Modelos Cognitivos	20
2.3	Ambiente Computacional COGENT	23
2.4	Lógica Nebulosa	25
2.5	Redes Neurais Artificiais	29
3.	METODOLOGIA	33
3.1	Modelo Cognitivo Simplificado	37
3.2	Construção do Sistema de Suporte no COGENT	38
3.2.1	Subsistema Controlado por Regras	38
3.2.2	Subsistema Controlado pela Rede Neuro-Nebulosa	43
3.2.3	Topologia da Rede Neural Utilizada	43
3.2.4	Equacionamento da Rede Neural Utilizada	44
3.2.5	Treinamento da Rede Neural Utilizada	50
3.2.6	Programação da Rede Neural na Linguagem MATLAB	55
3.2.7	Programação da Rede Neural no COGENT	57
3.2.8	Acidentes Treinados pelo Sistema Neuro-nebuloso	63

INDICE

4.	TESTES E ANÁLISE DOS RESULTADOS	59
4.1	Testes do Sistema Neuro-Nebuloso	68
4.1.1	Comparação dos Resultados Obtidos com os Dados de Treinamento	68
4.1.2	Análise dos Resultados Detalhados para o Acidente 1	70
4.1.3	Análise dos Resultados dos Testes de Robustez do Sistema	76
4.1.4	Análise dos Resultados do Teste de Capacidade de Generalização	80
4.2	Testes do Sistema Especialista Clássico e do Sistema Integrado	82
5.	CONCLUSÕES E RECOMENDAÇÕES	89
6.	REFERÊNCIAS	93
	APÊNDICE 1 - Regras das Tarefas de Segurança	97
	APÊNDICE 2 - Programas em Linguagem MATLAB	102
	APÊNDICE 3 - Tabelas com Resultados dos Acidentes 2 a 8	182

INDICE DE FIGURAS

Figura 1 – Arvore de Eventos do Controle das Ações do Operador	5
Figura 2 – A organização do Conhecimento na Memória	20
Figura 3 – Esquema do Sistema de Suporte	32
Figura 4 – Modelo Cognitivo Simplificado	37
Figura 5 – Exemplo de uma Regra do Sistema Especialista	38
Figura 6 – Sistema Controlado por Regras	39
Figura 7 – Vetor de Entrada de Dados	40
Figura 8 – Regras do Processador BASE DE REGRAS	41
Figura 9 – Exemplo de saída de Tarefa de Segurança	42
Figura 10 – Topologia da Rede	43
Figura 11 – Conjuntos Nebulosos do Parâmetro Pressão do Sistema de Refrigeração do Reator para as Tarefas 1, 2, 3, 4	46
Figura 12 – Fatoração dos Operadores OU nas Regras	47
Figura 13 – Arranjo dos Componentes da Rede no COGENT	57
Figura 14 – Regras de Normalização dos Parâmetros	58
Figura 15 – Regras de Separação dos Parâmetros em Conjuntos Nebulosos	59
Figura 16 – Exemplo de Processamento de uma Regra	60
Figura 17 – Exemplo do Cálculo da Camada 4	61
Figura 18 – Exemplo de Regras da Camada 5	62
Figura 19 – Exemplo de Saída das Tarefas de Segurança	63
Figura 20 – Parâmetros da Contenção no Acidente 1	70
Figura 21 – Parâmetros do Secundário no Acidente 1	71
Figura 22 – Saída do Nodo da Camada 4 Referente à Tarefa 20	74

INDICE DE FIGURAS

Figura 23 – Saída do Nodo Referente às Tarefas 1,2,3 e 4	75
Figura 24 – Saída do Nodo Referente à Tarefa 11	75
Figura 25 – Saída do Nodo Referente às Tarefas 8 e 9	76
Figura 26 – Saída do Nodo Referente à Tarefa 21	77
Figura 27 – Exemplo de um Vetor de Entrada	88
Figura 28 – Exemplo de uma Saída do Sistema Especialista Simbólico	89
Figura 29 – Exemplo de uma Saída do Sistema Especialista Conexcionista	90

ÍNDICE DE TABELAS

Tabela 1 – Tarefas de Segurança por Acidente	8
Tabela 2 – Parâmetros de Entrada do Sistema	34
Tabela 3 – Composição da Tarefa “Injeção de Alta Pressão”	35
Tabela 4 – Tarefas de Segurança	36
Tabela 5 – Valores da Média dos Conjuntos Nebulosos da Camada 2	55
Tabela 6 – Valores dos Desvios Padrão dos Conjuntos Nebulosos da Camada 2	56
Tabela 7 – Taxas de Acertos da Rede em Relação ao Conjunto de Regras	69
Tabela 8 – Tarefas Ativadas no Acidente 1	72
Tabela 9 – Taxas de Acertos da Rede em Relação ao Conjunto de Regras para um Desvio na Pressão do Sistema de Refrigeração do Reator	76
Tabela 10 – Tarefas Ativadas no Acidente 1 com o Valor da Pressão do Sistema de Refrigeração do Reator Fixo em 109 kgf/cm ²	77
Tabela 11 – Taxas de Acertos da Rede em Relação ao Conjunto de Regras	78
Tabela 12 – Tarefas Ativadas no Acidente 1 com o Valor da Pressão no Gerador de Vapor Fixo em 22 kgf/cm ²	79
Tabela 13 – Tarefas Ativadas no Acidente 8	81
Tabela 14 – Dados para Teste do Acidente 1	82
Tabela 15 – Dados para Teste do Acidente 3	83
Tabela 16 – Dados para Teste do Acidente 5	83
Tabela 17 – Dados para Teste do Acidente 7	84
Tabela 18 – Resultados Obtidos com os Dados de Teste	85

1. INTRODUÇÃO

1.1 Apresentação

No Brasil, toda pessoa física designada por uma Organização Operadora de um Reator ou Reatores Nucleares, para exercer as atividades de (1) manipular os controles de um reator, função denominada de Operador de Reator, ou (2) dirigir as atividades de Operadores de Reator, função denominada de Operador Sênior de Reator, deve ser licenciada pela Comissão Nacional de Energia Nuclear (CNEN) para exercer suas atividades.

O licenciamento é feito de acordo com a Norma CNEN – NN 1.01 [1] por meio de um processo que culmina em um exame de qualificação. A Norma estabelece os requisitos de pré-qualificação dos candidatos, os procedimentos do processo de licenciamento, as condições da licença, os critérios para renovação ou revogação das licenças, as especificações do exame de licenciamento e as obrigações básicas dos licenciados em serviço.

Nas exigências referentes à pré-qualificação dos candidatos são especificados os tempos de experiência técnica, os níveis de educação formal e os requisitos de treinamento específicos para cada atividade. Para a função de Operador Sênior de Reator, por exemplo, são requeridos 30 meses de experiência técnica global, térmica e nuclear, e a graduação em nível superior em campo tecnológico apropriado, a critério da CNEN. O treinamento específico exige cerca de dois anos, incluindo aulas em classe e em simulador.

O exame de qualificação é aplicado por uma banca examinadora composta de 5 componentes, designada para este fim pela Presidência da CNEN. Este exame é composto de um exame escrito, de um exame operacional oral nas instalações da central e de um exame em simulador, por ora aplicado pela banca apenas para a usina Angra 2, a qual possui um simulador de escopo total, no Brasil. O exame em simulador para Angra 1 é feito por uma empresa contratada, às expensas da ELETRONUCLEAR.

A equipe de operação da Sala de Controle é composta por cinco membros. O Supervisor e o Encarregado de Turno possuem licença de Operador Sênior de Reator. Um Operador de Reator e um Operador do Secundário (balanço da planta), licenciados como Operadores de Reator, e um Operador de Painéis de Retaguarda, não licenciado, geralmente um operador em preparação para a licença.

Após o acidente na Unidade 2 da Central Nuclear de Three Mille Island, em 1979, nos Estados Unidos, iniciado por problemas mecânicos e exacerbado por uma combinação de erros humanos que ocorreram quando da resposta a esses problemas, a indústria nuclear e os órgãos regulatórios resolveram, entre outras providências, reavaliar os requisitos de qualificação dos Operadores e Operadores Sênior de Reator.

Nas exigências do relatório da força-tarefa [2] criada pelo governo americano para avaliar este evento, foi feita a recomendação 1.16, a qual especifica a necessidade de melhoria da qualificação teórica dos Operadores Sênior de Reator. Como a Comissão Regulatória Nuclear (NRC) americana requeria apenas nível de segundo grau para esta função, a força-tarefa recomendou que fosse criada a função de Assessor Técnico do Turno, a qual requeria graduação superior em área tecnológica apropriada. A CNEN preferiu requerer esta graduação para os Operadores Sênior de Reator que exercem a função de Chefe de Turno. Foram, também, incluídos no exame de qualificação questões relativas à fluidodinâmica e à termodinâmica, necessários quando uma tomada de decisão deva ser baseada no conhecimento.

Como parte do processo de licenciamento da Usina Nuclear, por parte da CNEN, é exigida da organização operadora a elaboração de um documento denominado Relatório Final de Avaliação de Segurança [3]. No Capítulo 15 desse documento são avaliados transitórios operacionais, com probabilidade maior que 10^{-3} por ano, e eventos denominados acidentes postulados, cuja probabilidade é calculada entre 10^{-3} e 10^{-6} por ano. Na avaliação desses acidentes postulados, considera-se o critério de falha única, segundo o qual uma das redundâncias dos sistemas requeridos para combate do acidente específico está indisponível. Mais recentemente, vem sendo requerido que seja, também, dada consideração a acidentes de muito baixa probabilidade, denominados acidentes severos e além das bases de projeto, por exemplo, aqueles que consideram falhas múltiplas de componentes ou sistemas.

Para combater os acidentes bases de projeto são preparados procedimentos operacionais contendo as ações sequenciais requeridas por parte dos operadores. Os eventos considerados acidentes severos (além das bases de projeto) não possuem procedimentos detalhados como aqueles referentes a acidentes postulados. São disponibilizados apenas guias genéricos para o gerenciamento desses tipos de acidente.

Os procedimentos da usina são organizados no Manual de Operação da Usina (MOU) [4], composto de 17 partes, que vão desde procedimentos administrativos gerais, até procedimentos técnicos específicos para cada área de atuação. Os procedimentos relativos à operação estão contidos na parte 2 do MOU. Estes são organizados conforme segue:

1. PO-S - Procedimentos de operação específicos de sistemas;
2. PO-G – Procedimentos Gerais (incluem partida, parada, variações de carga);
3. PO-A – Procedimentos de operação anormal (falhas e maus funcionamentos de componentes);
4. PO-E – Procedimentos de operação em emergência básicos (diagnóstico, perda de refrigerante, ruptura de tubo de Gerador de Vapor);
5. PO-ES – Procedimento de operação em emergência suplementares (falhas adicionais);
6. PO-ECA – Procedimentos de operação em emergência eventuais (complicações durante a emergência, como perda de alimentação elétrica, por exemplo);
7. F – Funções críticas de segurança (árvores de estado);
8. RF – Restauradores de função crítica de segurança;
9. ALB – Livro de alarmes.

A equipe de operação, na Sala de Controle da usina, tem suas ações controladas por regras. Essas regras, expressas nos procedimentos do Manual de Operação da Usina, são aprendidas durante o treinamento específico, sendo amplamente discutidas em classe e exaustivamente treinadas em simulador.

O controle por regras, entretanto, por ter as condições de entrada nos procedimentos regida por valores determinados dos parâmetros, tem uma fragilidade

intrínseca quando confrontado com dados com “ruído”, que contêm valores inesperados ou dados incompletos.

Sistemas baseados em regras de produções (Productions Rule Systems) têm sido utilizados com sucesso para modelar atividades humanas como resolução de problemas e comportamento adaptativo, onde o modo clássico de representar o conhecimento é na forma de regras “SE – ENTÃO” [5]. O atendimento do antecedente resulta na execução do conseqüente, isto é, uma ação é executada. A abordagem clássica da representação do conhecimento é baseada na lógica bivalente, a qual é associada a uma séria limitação: sua inabilidade em lidar com os problemas de incertezas e imprecisões [5].

Quando, durante um evento operacional, os valores dos parâmetros importantes indicados no Painel de Controle não determinam uma regra a ser aplicada, o operador se encontrará numa situação de incerteza e, neste caso, deverá recorrer a seus conhecimentos da tecnologia e ciências básicas para produzir uma regra e aplicá-la à situação. Incerteza, neste caso, refere-se à falta de informações corretas e adequadas para subsidiar uma tomada de decisão.

A experiência mostra que, tomemos o acidente de Three Mille Island -2 como exemplo, erros humanos geralmente ocorrerão em situações inesperadas, cujo contexto não está plenamente compreendido pelo operador. Há que se considerar, também, o elevado grau de estresse inerente a essas situações.

A Figura 1 apresenta esses conceitos na forma de uma árvore de eventos [6]. Nesta figura a ação transcorre de cima para baixo. Listados à direita estão os eventos. Quando um evento é atendido a linha passa na vertical e, no caso oposto, ela se desloca para a esquerda. Verificamos, então, que os ramos 1, 2 e 3, à esquerda, descritos abaixo, referem-se aos casos em que a ação do operador é controlada com base no conhecimento.

1. Operação não rotineira; a situação não é perfeitamente compreendida pela equipe.
2. Operação não rotineira; situação perfeitamente compreendida pela equipe; caso não coberto por procedimento.

1.2 Conceito de Tarefas do Operador Durante um Acidente

O conjunto de ações para controle de um acidente em uma usina nuclear distingue entre gerenciamento orientado para o evento e gerenciamento orientado para os sintomas [7].

Por definição, gerenciamento de acidente orientado para o evento considera o tratamento específico para o evento que causou o acidente. A conexão direta entre a causa e o acidente permite que o guia para gerenciamento do acidente seja escrito na forma de procedimentos fixos, compostos de tarefas de segurança sequenciais.

Um diagnóstico válido do acidente é um requisito essencial para o gerenciamento orientado para o evento. Isto deve ter sido previamente efetuado pela equipe. Após o diagnóstico ser feito, isto é, se o evento que causou o acidente foi determinado, será possível prever o curso do acidente baseado no conhecimento da instalação e na experiência ganha em extensivas simulações por computador. Somente então se deve reagir aos possíveis perigos para a instalação, realizando ações manuais no tempo correto.

Sob o conceito de gerenciamento orientado para o evento, o diagnóstico do acidente é suficiente para se escolher um procedimento específico (conjunto de tarefas de segurança) para levar a instalação à condição de Desligada Fria.

Devido às múltiplas interações físicas entre os sistemas e a possibilidade de combinações de acidentes, os sintomas observáveis do acidente nem sempre permitem um reconhecimento preciso da causa. Conseqüentemente, o possível reconhecimento do acidente baseado nos sintomas será possível, somente após todos os outros acidentes que poderiam causar os mesmos sintomas terem sido descartados. Além disso, para os casos em que um diagnóstico não possa ser feito, ou que um diagnóstico incorreto tenha sido feito, um outro conceito de gerenciamento do acidente deverá estar disponível. Isto é buscado pelo gerenciamento orientado para os sintomas, com a utilização dos procedimentos restauradores de funções críticas de segurança [7].

O atendimento às funções críticas de segurança tem o objetivo de que, durante um acidente, a ameaça às condições de segurança da instalação seja reduzida, enquanto um diagnóstico não possa ter, ainda, sido feito.

O conceito de gerenciamento *orientado para os sintomas* é dirigido para assegurar que as funções críticas de segurança sejam mantidas. Evidências da manutenção destas são garantidas por critérios bem definidos e verificados, continuamente monitorados. Em caso de uma função crítica ser ameaçada ou violada, o pessoal de operação é solicitado a realizar ações manuais. As funções de segurança estarão satisfeitas se nenhum dos seus respectivos critérios tiver sido ameaçado ou violado.

Os procedimentos de emergência e restauradores de função crítica de segurança podem ser divididos em tarefas de segurança, sendo que para cada acidente existe um conjunto de tarefas especificado. Cada tarefa é composta de um conjunto de ações especificadas.

A Tabela 1 apresenta um arranjo das tarefas de segurança específico para usinas PWR.

Contudo, tanto os procedimentos de emergência, quanto os destinados a restaurar funções críticas de segurança, têm seus valores de parâmetros de entrada rigidamente especificados. Suas regras de aplicação têm como condições um conjunto fixo de valores rígidos desses parâmetros, estando, portanto, também, sujeitos aos problemas de dados com “ruído” ou incompletos.

Tabela 1 – Tarefas de Segurança por Acidente

Acidente	Pequena Perda de Refrig.	Média/ Grande Perda de Refrig.	Perda de Refrig. Secundário na Contenção	Perda de Refrig. Secundário Fora Contenção	Ruptura Tubo Gerador Vapor	Falha Dispos. Desarme Reator
Tarefa						
Desarme do Reator	█	█	█	█	█	█
Desarme da turbina	█	█	█	█	█	█
Injeção água borada						
Injeção alta pressão	█	█	█	█	█	
Injeção baixa pressão		█				
Injeção do acumulador		█				
Recirculação do poço		█				
Isolação da contenção	█	█	█			
Isolação ventilação contenção	█	█	█			█
Resfriam. contenção	█	█	█			
Aspersão contenção		█				
Última fonte fria	█	█	█	█	█	
Realinhamento ventilação	█	█				
Gerador Diesel	█	█	█	█	█	
Água de alimentação auxiliar	█		█	█	█	█
Desligamento das Bombas Refrigeração do Reator	█	█			█	
Isolamento água alimentação	█		█	█	█	
Resfriamento rápido do Sistema de Refrigeração do Reator					█	
Despressurização rápida do Sistema de Refrigeração do Reator					█	
Isolamento do acumulador dos Acumuladores	█		█	█	█	
Resfriamento para o Sistema de Remoção de Calor Residual	█					
Alinhamento do Sistema de Remoção de Calor Residual	█		█	█	█	
Fechamento das Válvulas de Isolamento das Linhas de Vapor Principal			█	█	█	█
Isolamento do Gerador de Vapor Falho			█	█	█	█
Resfriamento por drenagem/alimentação						

1.3 Sistemas de Suporte

As unidades 1 e 2 da Central Nuclear de Angra compartilham um sistema denominado Sistema Integrado de Computadores de Angra (SICA), desenvolvido pelo Laboratório de Monitoração de Processos (LMP) do Instituto Alberto Luiz Coimbra de Pós-graduação e Pesquisa em Engenharia (COPPE) da Universidade Federal do Rio de Janeiro (UFRJ). Nesse sistema estão incluídos vários aplicativos para dar suporte à operação das usinas, tanto em operação normal, quanto em acidente. Um deles, desenvolvido como uma das melhorias propostas pela empresa americana projetista e fornecedora de Sistemas Nucleares de Suprimento de Vapor (NSSS) para usinas nucleares Westinghouse, após o acidente ocorrido em 1979 na usina Three Mile Island, unidade 2, na Pensilvânia, Estados Unidos [2], é o denominado Sistema de Monitoração de Parâmetros de Segurança.

Este sistema tem a finalidade de apresentar em telas convenientemente dispostas os valores dos parâmetros operacionais de processo da usina, escolhidos pelo operador para cada situação operacional, podendo, inclusive, mostrar tendências de comportamento dos referidos parâmetros. Este sistema oferece, também, a facilidade de monitorar continuamente as árvores de estado das funções críticas de segurança, tanto em operação normal quanto em situação de transitório operacional ou acidente, proporcionando ao operador mais tempo para dedicar-se ao combate a um eventual transitório.

A monitoração dessas árvores de estado é requerida após um desarme do reator, quando é requerida a entrada no procedimento de operação em emergência PO-E 0 – Desarme do Reator ou Injeção de Segurança.

Este trabalho apresenta uma proposta de um sistema voltado para dar suporte ao operador após um desarme do reator em um transitório operacional ou acidente. São utilizados dois conceitos principais: Tarefas de Segurança e Contexto Familiar e Não Familiar ao operador.

As instruções seqüenciais dos Procedimentos de Operação em Emergência, relacionadas a um objetivo específico, são agrupadas em *pacotes* que recebem a denominação de Tarefas de Segurança. Cada uma dessas tarefas de segurança é aplicável a determinada situação da usina durante um transitório operacional ou

acidente, situação esta definida por um conjunto de valores dos parâmetros de segurança. Desta maneira, o sistema de suporte proposto poderá indicar as ações requeridas pela situação, independentemente de que tenha sido identificado o Procedimento de Operação em Emergência especificado para o evento.

Consideramos Contexto Familiar ao operador uma situação em que a condição da planta é perfeitamente compreendida, todos os valores e o comportamento dos parâmetros operacionais estão conforme previsto durante o seu treinamento e a evolução do evento ocorre conforme antecipado no projeto [8]. O contexto familiar não significa unicamente operação normal. Mesmo durante um grave evento, se tudo corre como antecipado no treinamento o operador estará em um contexto que lhe é familiar e suas ações serão controladas por habilidades ou regras.

Contexto Não Familiar é o oposto. Nessa situação, a condição da planta não é perfeitamente compreendida e os valores dos parâmetros indicados no painel de controle deixam dúvidas sobre a regra a ser utilizada. Nesta situação, as ações do operador serão controladas pelo seu conhecimento.

O sistema proposto nesta tese funciona da seguinte maneira: Dado um conjunto de valores dos parâmetros selecionados, o sistema indicará a tarefa, ou tarefas de segurança especificadas para a situação.

O sistema é híbrido, inteligente, combinando características simbólicas e conexionistas, com um controle por regras nos contextos familiares ao operador e por uma rede neuro-nebulosa fora destes. Este arranjo permite a integração do conhecimento do especialista (conhecimento explícito) com a capacidade de aprender por exemplos (conhecimento implícito). A lógica nebulosa provê um mecanismo de inferência sob incerteza, enquanto que a rede neural oferece vantagens de aprendizado, adaptação, tolerância a falhas, paralelismo e generalização.

No contexto familiar ao operador, o sistema de suporte proposto é comandado por um sistema especialista, com uma memória procedimental composta de regras SE-ENTÃO, que contêm as condições de entrada com valores rígidos dos parâmetros na parte se, e a tarefa de segurança correspondente na parte ENTÃO.

Quando o operador se encontra numa situação em que o conjunto de valores dessas variáveis cai num contexto não familiar, nenhuma regra na memória procedimental terá suas condições completamente atendidas. Nessa situação passa a ser preponderante o módulo conexionista da arquitetura. Esta parte faz a vez da base de conhecimentos teóricos do operador que seria usada nesta situação. Utilizamos uma Rede Neural Artificial tipo Radial Basis Network (neuro-nebulosa) [9], para classificar o conjunto de parâmetros e escolher uma regra que melhor se adapte ao conjunto de valores dos parâmetros. A rede é treinada com conjuntos contendo os valores de parâmetros de entrada nos procedimentos de emergência e a tarefa de segurança indicada para a situação.

Os dois módulos funcionam simultaneamente. Dado um conjunto de valores dos parâmetros especificados, o módulo controlado por regras fornecerá a tarefa ou tarefas de segurança especificadas, o mesmo ocorrendo com o módulo controlado pela rede neural. As indicações dos módulos servem como alternativas consistentes que deverão ser consideradas pelo operador na sua tomada de decisão. Se os parâmetros indicam diretamente uma regra a ser aplicada, haverá concordância entre o resultado dos dois módulos, o que confere um grande peso à tarefa indicada. Se os parâmetros não indicam diretamente uma regra a ser aplicada, o módulo da rede neural indicará uma tarefa baseado no treinamento recebido. Se as indicações dos dois módulos forem diferentes, o operador deverá considerá-las como alternativas possíveis.

1.4 Avaliação da Bibliografia sobre o Tema

Métodos da inteligência artificial têm sido grandemente utilizados em diagnóstico de falhas [10], predição e validação de sinais [11], controle de sistemas [12] e estimativa de erros de diagnóstico [13]. Na área nuclear esses métodos são utilizados em sistemas especialistas para suporte à operação de usinas nucleares, principalmente na monitoração das condições da instalação e na tomada de decisões. Para nossa análise vamos separar esses sistemas em aqueles que fazem o diagnóstico de falhas e acidentes, e aqueles que indicam as ações a serem tomadas em cada caso, como propõe o sistema apresentado neste trabalho.

Com relação ao diagnóstico de falhas, Alvarenga, Martinez e Schirru [10] utilizaram redes neurais, conjuntos nebulosos e algoritmos genéticos para identificar acidentes em uma usina nuclear; Nabeshima et al. [14] utilizaram uma rede neural auto associativa de três camadas para detectar anomalias na operação de uma usina nuclear por meio da verificação do desvio de sinais de processo medidos, dos previstos pela modelagem teórica; Uhrig, Hines e Nelson [15] construíram, operaram e testaram um protótipo de sistema integrando várias técnicas de inteligência artificial, tais como sistemas especialistas, redes neurais, lógica nebulosa e algoritmos genéticos, para prover informações aos operados da usina sobre seu estado operacional; Marseguerra et al. [16] propuseram um método baseado na lógica nebulosa para identificar o início de um transitório, com regras inferidas de dados de entrada e saída, providos por um código de cinética de reatores; Lee et al. [17] propuseram um sistema especialista para monitoração de procedimentos de operação durante a partida da usina; Lee e Seong [18] desenvolveram um sistema de aconselhamento para diagnóstico de acidentes baseado em uma rede neural; Santosh et al. [19] analisaram vários tipos de redes neurais para verificar a que melhor se adapta para o diagnóstico de transitórios em usinas nucleares; Shi, Xie e Wang [20] utilizaram uma rede neural com função de base radial (RBFNN) para desenvolver um sistema automático de diagnóstico de falha para usinas nucleares; Chen, Narita e Yamada [21] utilizaram uma rede neural treinada por algoritmo genético para desenvolver um sistema de diagnóstico para usinas nucleares; Mo, Lee e Seong [22] propuseram um modelo de diagnóstico de transitórios em usinas utilizando duas redes neurais e um processo de agregação dinâmica de redes neurais (DNNA).

Com relação ao diagnóstico com indicação de ações, Yang e Chang [23] apresentaram um sistema especialista para diagnóstico de transitórios em usinas nucleares baseado no conhecimento híbrido, estrutural, funcional, comportamental e heurístico, com a indicação das ações recomendadas aos operadores baseadas nesse diagnóstico; Cheon, Chang e Chung [24] apresentaram as estratégias para o desenvolvimento de um sistema especialista para processamento de múltiplos alarmes e diagnóstico em usinas nucleares com a indicação de ações de emergência baseados nos procedimentos da planta; Varde, Sankar e Verma [25] apresentaram um sistema especialista para suporte aos operadores de um reator de pesquisa para diagnóstico de falhas por uma rede neural e uma base de conhecimentos extraídos da Análise Probabilística de Segurança; Vinod et al. [26] desenvolveram um sistema de

suporte para diagnóstico de usinas nucleares por redes neurais utilizando como entrada os valores dos parâmetros da instalação e indicando os procedimentos a serem seguidos baseados nesse diagnóstico; Santosh, Saraf e Ghosh [27] apresentaram um sistema de suporte à operação baseado em redes neurais para diagnóstico de transitórios e indicação de procedimentos para correção baseados no diagnóstico.

O artigo de Lee e Seong [28], contemporâneo ao do projeto deste trabalho de tese [29], será analisado com mais detalhe. Nesse artigo os autores desenvolveram um sistema integrado de suporte à decisão para auxiliar as atividades cognitivas dos operadores durante situações anormais. As atividades cognitivas referidas são aquelas apresentadas no método ATHEANA [30], a saber: (1) Monitoração e detecção, (2) Análise da situação, (3) Planejamento da resposta e (4) Implementação da resposta. Para cada uma dessas atividades cognitivas é apresentado um sistema de suporte.

A atividade de planejamento da resposta é claramente sujeita a erros na tomada de decisão. Esta atividade compreende o modelo do estado da usina mentalizado pelo operador, a identificação de objetivos, a geração de planos de resposta alternativos, a avaliação dos planos de resposta e a escolha do plano de resposta mais apropriado para o modelo da situação [28]. Nessa situação, os autores ponderam que nos procedimentos escritos a informação pode ser, às vezes, excessiva, dificultando o contínuo gerenciamento dos passos requeridos. Nessa atividade os autores propõem como suporte um sistema de sugestão de operação adequada, utilizando um sistema computadorizado de procedimentos.

O sistema de suporte proposto neste trabalho de tese tem função semelhante: apresentar ao operador ações em resposta ao conjunto de valores dos parâmetros da usina. Essas ações funcionam como sugestões de alternativas a serem consideradas na formulação do plano de resposta. Ao prover informações na forma de tarefas de segurança, ao invés de instruções sequenciais, o sistema facilita a definição de objetivos e a seleção do plano de resposta, ao fornecer informação de grande conteúdo em forma reduzida.

1.5 Objetivos e Organização do Trabalho

Este trabalho propõe um conceito de sistema de suporte ao operador de uma usina nuclear, conforme o apresentado no item anterior, e, por intermédio da construção e teste de um protótipo, visa demonstrar que este sistema é realizável.

Foram selecionadas apenas tarefas de segurança dos procedimentos básicos de operação em emergência da unidade 1 da Central Nuclear de Angra (Angra 1), e escolhidos apenas parâmetros da instalação necessários para a ativação dessas tarefas. O treinamento do sistema neuro-nebuloso foi feito com dados referentes a 7 acidentes bases de projeto, escolhidos por serem eventos representativos da avaliação de segurança da usina. Os dados foram obtidos por intermédio do simulador gráfico do Centro de Treinamento da ELETRONUCLEAR. Este simulador é da usina de Krsko, na República da Eslovênia, fabricada pela Westinghouse, de dois circuitos de refrigeração do núcleo, muito similar a Angra 1.

Este trabalho é estruturado da forma descrita a seguir.

Neste capítulo apresentamos o conceito de tarefas do operador de uma usina nuclear durante um transitório operacional e discutimos como é fundamentada a tomada de decisões por parte do ser humano, nos contextos familiares e não familiares possíveis de ser encontrados ao longo do desenvolvimento do evento. Baseado nesta discussão foi apresentado um conceito de sistema de suporte ao operador que leva em consideração essas situações, sendo, também, explicada a motivação e a relevância desta pesquisa. Por intermédio de uma avaliação da bibliografia encontrada sobre este tema procuramos mostrar que o sistema proposto é inédito.

O Capítulo 2 apresenta os fundamentos teóricos aplicados no desenvolvimento do sistema. Esses fundamentos são:

1. **Sistemas especialistas.** Este fundamento é utilizado no desenvolvimento geral do sistema, que é um sistema especialista híbrido. A parte referente aos sistemas especialistas clássicos é utilizada no módulo referente aos contextos familiares ao operador e a parte referente à lógica nebulosa e às

redes neurais, utilizada no módulo referente aos contextos não familiares ao operador.

2. Modelos cognitivos. Este fundamento também é utilizado no desenvolvimento geral do sistema, pois este é construído sobre um modelo cognitivo. Este tipo de construção tem o objetivo de que este, ao ser implementado, permita uma maior facilidade de interação com o usuário, conforme será discutido no texto deste trabalho.
3. Ambiente computacional COGENT. Nesse item é apresentado o ambiente computacional no qual o sistema é construído.
4. Lógica Nebulosa. Neste item são apresentados fundamentos sobre a lógica nebulosa e sobre sistemas baseados em regras nebulosas que são utilizados no módulo referente aos contextos não familiares ao operador.
5. Redes neurais. Os fundamentos apresentados neste item são utilizados para fazer o ajuste do sistema baseado em regras nebulosas, referente aos contextos não familiares ao operador, com a utilização dos dados obtidos no simulador.

O Capítulo 3 descreve a metodologia utilizada para a aplicação desses fundamentos na solução do problema de engenharia que é a construção de um protótipo do sistema de suporte ao operador. Inicialmente é feito um detalhamento do sistema, com a sua descrição, dados de entrada e saída, bem como é explicado o conceito de tarefa de segurança e o tipo de regra utilizado pelo sistema. A seguir é explicada a implementação do sistema no ambiente computacional COGENT.

O Capítulo 4 apresenta os testes aplicados ao protótipo de sistema para verificar se este atende aos objetivos propostos no projeto conceitual. São apresentados os resultados obtidos nos testes, os quais demonstram que o protótipo alcança os objetivos buscados.

O Capítulo 5 faz um sumário dos objetivos da pesquisa e apresenta suas conclusões, baseando-as nos resultados obtidos no Capítulo 4. São, também, apresentadas recomendações para a continuação da pesquisa e desenvolvimento do sistema proposto.

O Apêndice 1 mostra as regras utilizadas pelo sistema, extraídas dos Procedimentos de Operação em Emergência da Unidade 1 da Central Nuclear de Angra.

O Apêndice 2 contém os dois programas em linguagem MATLAB para o treinamento da rede neural utilizada no módulo referente aos contextos não familiares ao operador e para teste do sistema híbrido neuro-nebuloso.

O Apêndice 3 mostra os resultados obtidos nos testes do sistema nos cenários referentes aos acidentes 2 a 8. Para simplificação, no texto foram apresentados somente os resultados dos testes para o acidente 1.

2. FUNDAMENTOS TEÓRICOS

2.1 Sistemas Especialistas

Sistemas especialistas são sistemas baseados no conhecimento que provêm especialização em uma área restrita de aplicação. Eles são compostos dos seguintes blocos principais: base de conhecimentos, base de dados, motor de inferência, módulo de explicação, módulo de aquisição de conhecimento e a interface com usuário [31].

Na base de conhecimentos reside o conhecimento sobre o problema. Ela pode ser uma memória de produções, uma rede conexionista ou um conjunto de regras nebulosas.

Sistema de produções são atualmente as linguagens e métodos mais comuns de representação de conhecimento em sistemas especialistas. As regras de produção têm dois lados: um lado esquerdo, que expressa as condições ou premissas, e um lado direito, que define que ações devem ser executadas, quando a regra é satisfeita.

A base de dados contém fatos correntes ou passados, Ela é a *memória de trabalho* nas linguagens de produções. Em algumas arquiteturas o módulo contendo dados passados, pode ser usado como adições de conhecimento da base de conhecimentos.

O motor de inferência é um programa que controla o funcionamento do sistema como um todo. Ele contém o mecanismo de inferência que pode ter encadeamento para frente, para trás ou uma combinação destes. O processo de casamento das condições das regras pode ser *exato* ou *parcial*. O casamento exato significa que os fatos a serem comparados coincidem exatamente com os elementos das condições do lado esquerdo das regras. Um casamento parcial não requer igualdade entre os fatos e o lado esquerdo das regras. Nestes casos somente uma aproximação é necessária. Mas, o grande problema associado ao casamento parcial é quão aproximadamente os fatos casam com as condições, para que a regra seja disparada. Casamento parcial é típico de sistemas nebulosos e redes neurais.

O motor de inferência em um sistema de produções executa a inferência em um encadeamento avante, da seguinte maneira:

1. Compara todos os fatos da memória de trabalho com as condições de todas as produções e define quais delas são satisfeitas. Toda combinação de fatos que satisfaz uma produção é denominada INSTANCIAMENTO. Todas as produções que são satisfeitas pela comparação corrente formam um CONJUNTO DE CONFLITO, ou AGENDA.

2. Seleciona uma regra da agenda com a utilização de uma estratégia de seleção. Algumas das estratégias de seleção utilizadas em sistemas de produção são as seguintes:
 - a) RECENTIDADE. A regra “instanciada” pelos fatos mais recentes é selecionada.

 - b) ESPECIFICIDADE. As regras com mais elementos de condição têm prioridade maior. Isto restringe o “instanciamento”, pois estas regras devem ser “instanciadas” um menor número de vezes.

 - c) REFRAÇÃO. Uma vez que a regra é ativada para um “instanciamento” particular, ela não será novamente ativada no mesmo “instanciamento”, podendo ser apenas por alguns ciclos, ou para sempre.

 - d) SALIÊNCIA. Um número de prioridade pode ser associado a cada regra e este será o critério para seleção de uma regra da agenda.

 - e) SELEÇÃO ALEATÓRIA. Uma regra é selecionada aleatoriamente da agenda.

3. Executa a regra selecionada, efetuando as ações especificadas.

O motor de inferência atua até que não haja mais produções na agenda, ou é compulsoriamente desligado.

O módulo de explicação rastreia a execução do sistema especialista e acumula informação sobre o curso do processo de raciocínio. Ele então transfere a informação para o usuário.

A interface com o usuário comunica-se com o meio ambiente e interage com o usuário.

O módulo de aquisição de conhecimentos é projetado para acumular conhecimentos para construir a base de conhecimentos.

Na inteligência artificial três abordagens distintas são adotadas para sistemas especialistas, a simbólica, às vezes chamada clássica, a conexionista e a híbrida.

Na abordagem clássica os sistemas especialistas são baseados em regras do tipo SE – ENTÃO, às vezes associadas a uma estimativa de probabilidade. Durante sua execução, um MOTOR DE INFERÊNCIA examina a base de regras buscando verificar a coincidência das condições correntes com as condições de alguma dessas regras, para selecionar aquelas possíveis de serem DISPARADAS. As regras devem ser obtidas de especialistas no domínio considerado e adicionadas durante a fase de projeto. A obtenção dessas regras é frequentemente a tarefa mais difícil na construção desse tipo de sistema [32]. As regras podem, também, ser construídas com a utilização de lógica nebulosa, contendo valores linguísticos, tais como: grande, frio ou baixo.

Na abordagem conexionista as perspectivas de AQUISIÇÃO DE DADOS, INFERÊNCIA e ANALISE podem ser efetuadas por uma rede neural.

Redes conexionistas podem aliviar algumas das dificuldades associadas à construção de sistemas especialistas, tais como:

1. Redes neurais podem extrair regras automaticamente de exemplos, sem depender de serem explicitadas a priori por especialistas no domínio.

2. A representação distribuída inerente a redes de multicamadas provê uma base para a generalização entre as categorias observadas, mesmo que exemplos específicos não tenham sido mostrados.
3. Redes neurais oferecem execução em paralelo e, com a implementação apropriada, podem ser mais efetivas para a inferência do que regras em série como nos sistemas clássicos.

Sistemas híbridos combinam essas duas abordagens de forma complementar. Estes podem ser modulares ou integrados. Nos sistemas modulares as duas abordagens são usadas separadamente, na forma de módulos. Nos sistemas integrados a informação simbólica é inserida de alguma forma na rede neural pelo especialista no domínio. Depois que o conhecimento do domínio tem uma representação neural, exemplos são usados para treinamento, na intenção de refinar o conhecimento inicial. O grande desafio na construção de um sistema híbrido está na colocação do conhecimento do especialista na rede neural.

2.2 Modelos Cognitivos

Um modelo cognitivo de comportamento tenta descrever, explicar e prever o comportamento humano em um domínio, modelando que informações as pessoas coletam, como as adquirem, como são representadas internamente e como são usadas para guiar o comportamento [33].

Existem duas linhas principais de arquiteturas cognitivas, a simbólica e a subsimbólica. A linha subsimbólica leva aos modelos conexionistas que são representados por redes de grande quantidade de *unidades* interconectadas, cada uma com certa quantidade de ativação, que se espalha para as outras unidades interconectadas por intermédio de pesos associados. A linha simbólica inicia-se com modelos paramétricos, como o MHP – Model Human Processor [33], que usa alguns parâmetros para caracterizar a arquitetura, evoluindo para os modelos integrativos, que apresentam como característica mecanismos cognitivos detalhados interagindo entre si [34].

Arquiteturas integrativas, como seu nome sugere, integram em um único e coerente arcabouço, vários submodelos do comportamento humano (sensação, percepção, cognição e assim por diante) [33]. As arquiteturas integrativas, de algum modo, refletem uma teoria do comportamento humano unificada. Nestas, submodelos relacionados interagem por intermédio de representações, em comum, de resultados intermediários de processamento da informação, nos quais são usadas representações conceituais consistentes e ferramentas similares.

Existem quatro modos gerais principais de desenvolvimento de modelos de desempenho humano [35]: (1) processamento da informação, (2) teoria do controle, (3) rede de tarefas e (4) baseado no conhecimento. O modelo que vamos utilizar é uma arquitetura cuja tomada de decisão é baseada no conhecimento

O modo baseado no conhecimento tem suas raízes na psicologia cognitiva e na ciência da computação (inteligência artificial). O campo da ciência cognitiva que representa a intersecção dessas disciplinas tem como objetivo o desenvolvimento de representações formais dos processos cognitivos humanos, tais como tomada de decisão, resolução de problemas, planejamento ou entendimento. Algumas vezes essas representações são algoritmos; mais frequentemente são expressas na forma de simulações dos processos que se acredita serem utilizados pelos seres humanos. As ferramentas do especialista em inteligência artificial, tais como programação orientada ao objeto, são amplamente utilizadas para implementar essas simulações e são aplicadas para a modelagem de sistemas homem-máquina. Elas suprem as bases para modelos muito flexíveis que podem ser preparados com relativa simplicidade para produzir resultados para a avaliação de desempenho, como, por exemplo, tempo previsto de execução de uma tarefa pelo ser humano, ou ainda com saídas de sumários gráficos.

Modelos de desempenho humano baseados no conhecimento são explicações de como as pessoas decidem o que deve ser feito para resolver um problema. Eles evoluíram das pesquisas de Newell e Simon [36] em simulação por computador da resolução humana de problemas. Esses pesquisadores verificaram que programas de computador podem ser vistos como manipulação de símbolos em vez de execução de cálculos aritméticos. Eles arguíram que o pensamento humano é também um exemplo

de manipulação de símbolos e desse modo podem ser modelados por programas de computador.

A ideia básica, por trás da simulação por computador, é que o conhecimento pode ser representado por estruturas de símbolos e regras para operação nestas. A ideia de que o pensamento pode ser modelado por programas de computador de nenhum modo implica que a maquinaria do cérebro humano seja logicamente similar aos dispositivos digitais modernos. Como mostrado na Figura 2, o conhecimento, nos modelos baseados no conhecimento, é organizado em duas classes distintas: informação na memória de trabalho (ou de curta duração) e informação na memória de longa duração.

A representação do problema é armazenada na memória de trabalho, como um conjunto de proposições. Essas proposições podem se referir a conhecimentos sobre o problema a ser resolvido.

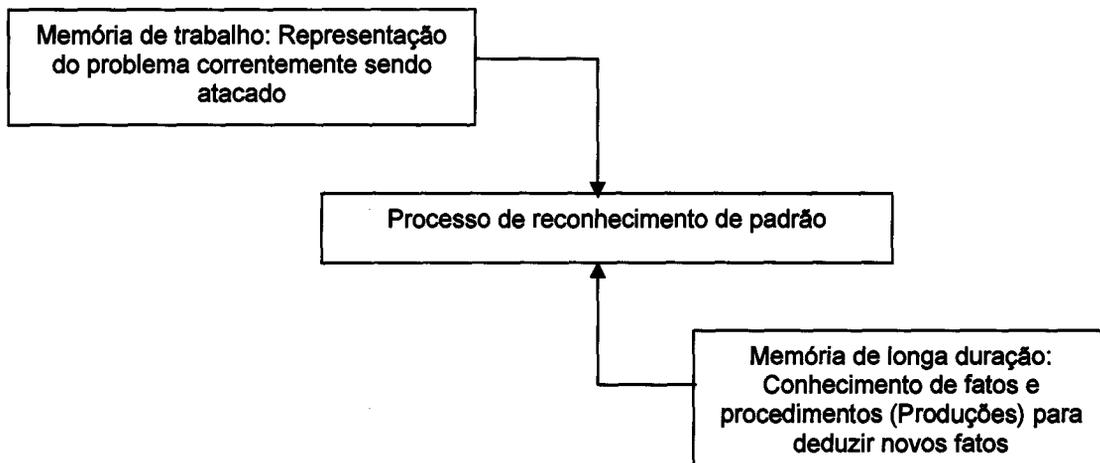


Figura 2 - A organização do Conhecimento na Memória

Assume-se que a informação sobre como resolver problemas é residente na memória de longo termo. Os fatos e métodos são chamados de informação declarativa sobre resolução de problemas. Eles estão armazenados na forma de regras SE-ENTÃO, chamadas produções.

Estes componentes constituem-se em:

- **Processos baseados em regras:** Manipulam a informação de acordo com as regras simbólicas especificadas pelo usuário.
- **Memórias de armazenamento:** Memórias são dispositivos de armazenamento de informações em geral. Elas podem ser usadas para memória de curta e longa duração e diferentes subtipos podem armazenar dados em diferentes formatos (proposicional, tabular e analógico). O comportamento detalhado de qualquer memória é determinado pelas suas propriedades, as quais especificam características tais como limitação de capacidade, parâmetros de decaimento e restrições de acesso.
- **Redes connexionistas:** O processo permite interface com alguns objetos connexionistas o que o torna útil para aplicações de modelagem híbrida. São incluídos dois tipos de redes: bidirecional e multicamadas com associações mútuas.
- **Fontes e drenos de dados:** Componentes especializados de fonte de dados permitem que estes sejam alimentados aos componentes de uma maneira controlada. Da mesma forma os *drenos* de dados permitem a coleta de dados de componentes durante a execução do modelo.
- **Conexões de comunicação entre módulos:** A comunicação entre módulos é feita pelas setas. Há dois tipos de setas: setas de leitura e setas de enviar/registrar.

A linguagem de modelagem baseada em regras do COGENT permite que processos complexos sejam especificados em termos de regras de produção. Cada regra consiste em um conjunto de condições, um conjunto de ações e, opcionalmente, um disparador. As condições incluem operadores lógicos, cujo resultado pode ser verdadeiro ou falso, tais como o “casamento” de um elemento em uma memória ou o

teste da igualdade de elementos de dados. As ações resultam em mensagens de várias formas sendo enviadas a outras “caixas”.

2.4 Lógica Nebulosa

Na lógica nebulosa uma afirmação é verdadeira ou falsa somente em certo grau. Objetos pertencem a um conjunto nebuloso somente em certo grau. Regras nebulosas associam a parte SE de conjuntos nebulosos à parte ENTÃO de conjuntos nebulosos. Sistemas Baseados em Regras Nebulosas (SBRN) são conjuntos de regras nebulosas que mapeiam uma entrada a uma saída. [9].

A lógica nebulosa pode ser vista como uma extensão da lógica clássica, provendo um arcabouço conceitual eficiente para lidar com o problema de representação do conhecimento, em um ambiente de incerteza ou de imprecisão. Como seu próprio nome sugere, ela é uma forma de lógica cujos modos de realização são aproximados, em vez de exatos. Sua importância decorre do fato de que a maioria dos modos de raciocínio humano, especialmente o raciocínio de senso comum, é aproximada em natureza [5].

O cérebro humano interpreta a informação sensorial incompleta e imprecisa recolhida pelos órgãos perceptivos. A teoria dos conjuntos nebulosos provê um cálculo sistemático para lidar com essa informação linguisticamente, efetuando um cálculo numérico com a utilização de valores estipulados pelas funções de pertinência. A função de pertinência mapeia cada elemento de um conjunto a um grau de pertinência a este conjunto, ou seja, um valor de pertinência, usualmente entre 0 e 1.

Assim, a definição de conjunto nebuloso é apenas uma extensão da definição de conjunto clássico, onde a função de pertinência é permitido assumir um valor entre 0 e 1. Se a função de pertinência for permitido apenas os valores 0 e 1, este se tornará um conjunto clássico.

Na maioria das utilizações de Sistemas Baseados em Regras Nebulosas as funções de pertinência são determinadas por especialistas. Essas funções podem ou não ser precisas o suficiente para a aplicação desejada. Nestes casos, existem

técnicas de otimização para ajuste fino das funções de pertinência, visando melhorar o desempenho do sistema. Essa é uma importante aplicação das redes neurais, formando assim os chamados Sistemas Neuro-Nebulosos.

Por exemplo, se X é uma coleção de objetos denotados genericamente por x , então um conjunto nebuloso A em X é definido como um conjunto de pares ordenados tais que:

$$A = \{(x, \mu_A(x)) \mid x \in X\} \quad (2.1)$$

onde $\mu_A(x)$ é chamada função de pertinência do conjunto nebuloso A . A função de pertinência mapeia cada elemento de X a um valor de pertinência entre 0 e 1.

A construção de um conjunto nebuloso depende de duas coisas: a escolha de um universo adequado e a especificação da função de pertinência apropriada. Funções de pertinência podem ser definidas em uma ou mais dimensões. Para os conjuntos nebulosos são aplicáveis as mesmas operações dos conjuntos clássicos, tais como união, intersecção, etc.

Regras nebulosas e raciocínio nebuloso formam a espinha dorsal de sistemas de inferência nebulosos. SBRNs têm sido aplicados a uma grande variedade de áreas, tais como controle automático, sistemas especialistas, reconhecimento de padrões, predições em séries temporais e classificação de dados.

As regras nebulosas assumem a seguinte forma:

$$\text{Se } x \text{ é } A \text{ então } y \text{ é } B, \text{ ou } A \rightarrow B \quad (2.2)$$

onde A e B são valores linguísticos definidos por conjuntos nebulosos nos universos X e Y , respectivamente. Geralmente " x é A " é chamado de antecedente, ou premissa, enquanto que " y é B " é chamado de conseqüente ou conclusão.

Raciocínio nebuloso, também conhecido como raciocínio aproximado, é um procedimento de inferência que deriva conclusões de um conjunto de regras nebulosas SE-ENTÃO e um conjunto de fatos conhecidos. A regra básica de

inferência na tradicional lógica de dois valores é o MODUS PONENS, segundo a qual, podemos inferir a verdade de uma proposição B , a partir da verdade de uma proposição A e da implicação $A \rightarrow B$ [36].

O sistema de inferência nebuloso é um arcabouço de cálculo baseado nos conceitos da teoria dos conjuntos nebulosos, das regras nebulosas SE-ENTÃO e do raciocínio nebuloso. A estrutura básica desse sistema compõe-se de três componentes conceituais: uma base de regras, que contém o conjunto de regras nebulosas selecionadas, uma base de dados, que define as funções de pertinência usadas nas regras nebulosas, e um mecanismo de raciocínio, que executa o procedimento de inferência sobre as regras e fatos dados para derivar uma conclusão ou saída racional.

Às vezes é necessário um valor rígido de saída, especialmente em situações em que o sistema de inferência é usado como um controlador. Deste modo, haverá necessidade de um método de transformação de um conjunto nebuloso em um valor rígido que melhor represente este conjunto nebuloso. Este processo é chamado “defuzzyficação”.

Dois métodos de efetuar esta transformação de um conjunto nebuloso em um valor rígido são mais utilizados [38]: método do Centro de Gravidade, ou Centróide da Área, e método Média dos Máximos.

No método do centro de gravidade, para “defuzzyficar” um conjunto nebuloso A , num universo Z , a seguinte equação se aplica:

$$CG = \frac{\int \mu_A(z)zdz}{\int \mu_A(z)dz} \quad (2.3)$$

onde $\mu_A(z)$ é função de pertinência.

O método Média dos Máximos encontra o valor que é a média dos valores máximos de z nos quais a função de pertinência atinge o máximo μ^* . A seguinte equação se aplica:

$$MM = \frac{\int_{Z'} zdz}{\int_{Z'} dz} \quad (2.4)$$

onde $Z' = \{z \mid \mu_A(z) = \mu^*\}$. Em particular, se $\mu_A(z)$ tem um único máximo em $z = z^*$, então $MM = z^*$.

Existem três tipos principais de sistemas de inferência nebulosos utilizados nas mais variadas aplicações: Mamdani, Sugeno e Tsukamoto [5].

O sistema Mamdani é conhecido como o *controlador de lógica nebulosa* cuja estrutura genérica compreende uma base de conhecimentos que armazena os conhecimentos disponíveis sobre o problema, na forma de regras nebulosas SE-ENTÃO. Os outros três componentes são: a interface de “*fuzzyficação*”, que estabelece uma relação entre os valores rígidos no domínio de entrada e os conjuntos nebulosos no mesmo domínio, o sistema de inferência e a interface de “*defuzzyficação*” que realiza a operação de ligar conjuntos nebulosos no domínio de saída a valores rígidos no mesmo domínio.

O sistema Sugeno foi desenvolvido num esforço para sistematizar a geração de regras nebulosas de um dado conjunto de dados ENTRADA-SAÍDA. As regras desse sistema são da forma:

$$\text{Se } x \text{ é } A \text{ e } y \text{ é } B \text{ então } z = f(x,y) \quad (2.5)$$

onde A e B são conjuntos nebulosos no antecedente, enquanto $z = f(x,y)$ é uma função rígida no conseqüente da regra.

No sistema Tsukamoto, o conseqüente de cada regra nebulosa é representado por um conjunto nebuloso com uma função de pertinência monotônica. Como resultado, a saída inferida de cada regra é definida como um valor rígido induzido pela força de disparo da regra. A saída total é a média ponderada da saída de cada regra. Como cada regra infere uma saída rígida, este modelo agrega a saída de cada regra pelo método de média ponderada, evitando o demorado processo de “*defuzzyficação*”.

As principais aplicações de sistemas de regras nebulosas são: a modelagem de funções, controladores de processo e classificação de padrões.

De um modo genérico, o objetivo de um classificador de padrões, que utilizaremos nesta pesquisa, é designar uma classe C_j de um conjunto de classes $C = \{C_1, C_2, C_3, \dots, C_n\}$ a um objeto, baseado em seu vetor característico $\{x_1, \dots, x_n\} \in R^n$. O problema de construir este classificador é encontrar um mapeamento:

$$D: R^n \rightarrow C \quad (2.6)$$

que seja ótimo no sentido de que ele maximize uma certa medida de desempenho $\delta(D)$. O algoritmo de aprendizado começa com um conjunto de exemplos de treinamento corretamente classificados com o objetivo de minimizar o erro no espaço especificado. O desempenho do classificador é avaliado por meio de um conjunto de exemplos de teste, não previamente vistos, para se obter uma estimativa do erro de classificação. O classificador pode ser implementado com exemplos como um sistema nebuloso, uma rede neural, uma árvore de decisão ou uma rede bayesiana [39].

2.5 Redes Neurais Artificiais

Redes Neurais artificiais são modelos computacionais compostos de unidades de processamento interconectadas que executam processamento em paralelo. São implementações de sistemas não lineares estáticos ou dinâmicos. Uma característica muito importante dessas redes é sua natureza adaptativa onde “aprender por exemplos” substitui a “programação” para a solução de problemas. Essa característica torna esses modelos computacionais muito atraentes para domínios onde se tem pouco entendimento ou entendimento incompleto do problema a ser resolvido, porém estão facilmente disponíveis exemplos para treinamento [40].

Existem varias aplicações nas quais esses modelos computacionais são viáveis. Nestas aplicações incluem-se: classificações de padrões, reconhecimento e síntese vocais, interfaces adaptativas entre humanos e sistemas complexos, aproximação de funções, compressão de imagens, memória associativa, formação de conjuntos, organização combinatória, modelagem de sistemas não lineares e controle.

Redes neurais são modelos biologicamente inspirados e consistem de elementos processadores (chamados neurônios, ou nodos), conexões entre eles com coeficientes (pesos) ligados às conexões, os quais constituem a estrutura da rede neural, e algoritmos de treinamento e recuperação ligados à estrutura. Elas são chamadas de modelos conexionistas devido ao papel importante das suas conexões. Os pesos das conexões constituem a memória do sistema [31].

As características principais dessas redes são: aprendizado, adaptação, generalização, paralelismo massivo, robustez, armazenamento associativo de informação e processamento de informação espacial e temporal.

Em geral, um modelo de neurônio artificial é baseado nos seguintes parâmetros, que o descrevem:

- Conexões de entrada do neurônio u_j : x_1, x_2, \dots, x_n . Cada uma com seu peso ligado: $w_{1j}, w_{2j}, \dots, w_{nj}$.
- Função de entrada f . Calcula o sinal líquido agregado do neurônio $u_j = f(x, w_{ij})$, onde x e w são os vetores de entrada e de pesos, usualmente f é a somatória:

$$U_j = \sum_{i=1,n} x_i * w_{ij} \quad (2.7)$$

- Função de ativação s : Calcula o nível de ativação do neurônio

$$A_j = s(u_j) \quad (2.8)$$

- Função de saída: calcula o valor do sinal de saída emitido pelo neurônio. O sinal de saída é também chamado de nível de ativação do neurônio.

As funções geralmente utilizadas como funções de ativação são:

1. Função de ativação limitada (função degrau). Se a entrada líquida para o neurônio é superior a certo limite o neurônio se torna ativo, caso contrário, este fica inativo.
2. Função de ativação linear. O valor de ativação aumenta linearmente com o aumento da entrada líquida, porém após um determinado limite este se estabiliza.
3. Função de ativação sigmoide. É qualquer função de transformação não linear em forma de S, com as características de: crescimento monotônico, ligada e contínua.
4. Função de ativação gaussiana.

Embora um único neurônio possa efetuar a função de processamento da informação, o poder da computação neural vem da conexão destes em uma rede. O tipo de conexão dos neurônios em uma rede define sua topologia. Neurônios em uma rede podem ser completamente conectados, isto é, cada neurônio é conectado a todos os outros neurônios, ou parcialmente conectado, onde, por exemplo, cada neurônio é conectado somente com neurônios de outras camadas.

Em uma rede, de acordo com a presença ou ausência de conexões de realimentação, há dois tipos de arquiteturas:

1. Arquitetura de alimentação para frente (*feedforward*), na qual não há conexões vindas dos neurônios da saída para os da entrada. A rede não mantém registro de suas saídas anteriores e dos estados de ativação de seus neurônios.
2. Arquiteturas recorrentes (*feedback*), nas quais há conexões vindas dos neurônios de saída para os de entrada. Tais redes mantêm uma memória dos estados anteriores e suas saídas não dependem somente das entradas, porém também dos seus estados anteriores.

A característica mais atrativa das redes neurais é a sua habilidade de aprender. O aprendizado torna possível a modificação de comportamento em resposta ao ambiente. A rede é treinada de modo que um conjunto X de vetores de entrada produz o desejado (ou ao menos consistente) conjunto de saídas Y , ou a rede aprende sobre características internas ou estrutura de dados de X . O conjunto X usado para treinar a rede é chamado de conjunto de treinamento. Os elementos x deste conjunto X são chamados exemplos de treinamento. O processo de treinamento é refletido na mudança dos pesos das conexões da rede. Durante o treinamento, os pesos devem gradualmente convergir de modo que cada exemplo x do conjunto de treinamento cause a saída desejada y produzida pela rede.

A habilidade de aprender da rede é alcançada pela aplicação de um ALGORITMO DE APRENDIZADO, classificado, principalmente, em três grupos:

1. Supervisionado. Os exemplos de treinamento contêm os vetores de entrada x e os desejados vetores de saída y . O treinamento é realizado até que a rede aprenda a associar cada vetor de entrada x ao seu correspondente e desejado vetor de saída y .
2. Não supervisionado. Apenas vetores de entrada x são fornecidos. A rede aprende apenas algumas características internas de todo o conjunto de vetores de entrada fornecidos a ela.
3. Aprendizado de prêmio-penalização. Este tipo de aprendizado é baseado em apresentar-se um conjunto de vetores de entrada x a uma rede neural e se verificar os vetores de saída. Se o vetor de saída é considerado "bom", então um "prêmio" é dado à rede, no sentido de que os pesos das conexões existentes são aumentados, caso contrário, a rede é "punida" e os pesos das conexões são reduzidos, por ser considerado um conjunto não apropriado.

O aprendizado não é uma habilidade individual de um único neurônio. É um processo coletivo de toda a rede neural e o resultado de um processo de treinamento. A matrix W dos pesos das conexões tem seu significado como um padrão global. Ela representa o "conhecimento" do seu todo.

3. METODOLOGIA

Neste capítulo vamos aplicar os fundamentos teóricos apresentados no Capítulo 2 à solução de um problema de engenharia, que é a construção de um protótipo do sistema de suporte ao operador, cujo conceito foi apresentado no Capítulo 1. O sistema é construído sobre um modelo cognitivo simplificado, conforme será explicado no item 3.1. A estrutura do sistema de suporte é apresentada na Figura 3.

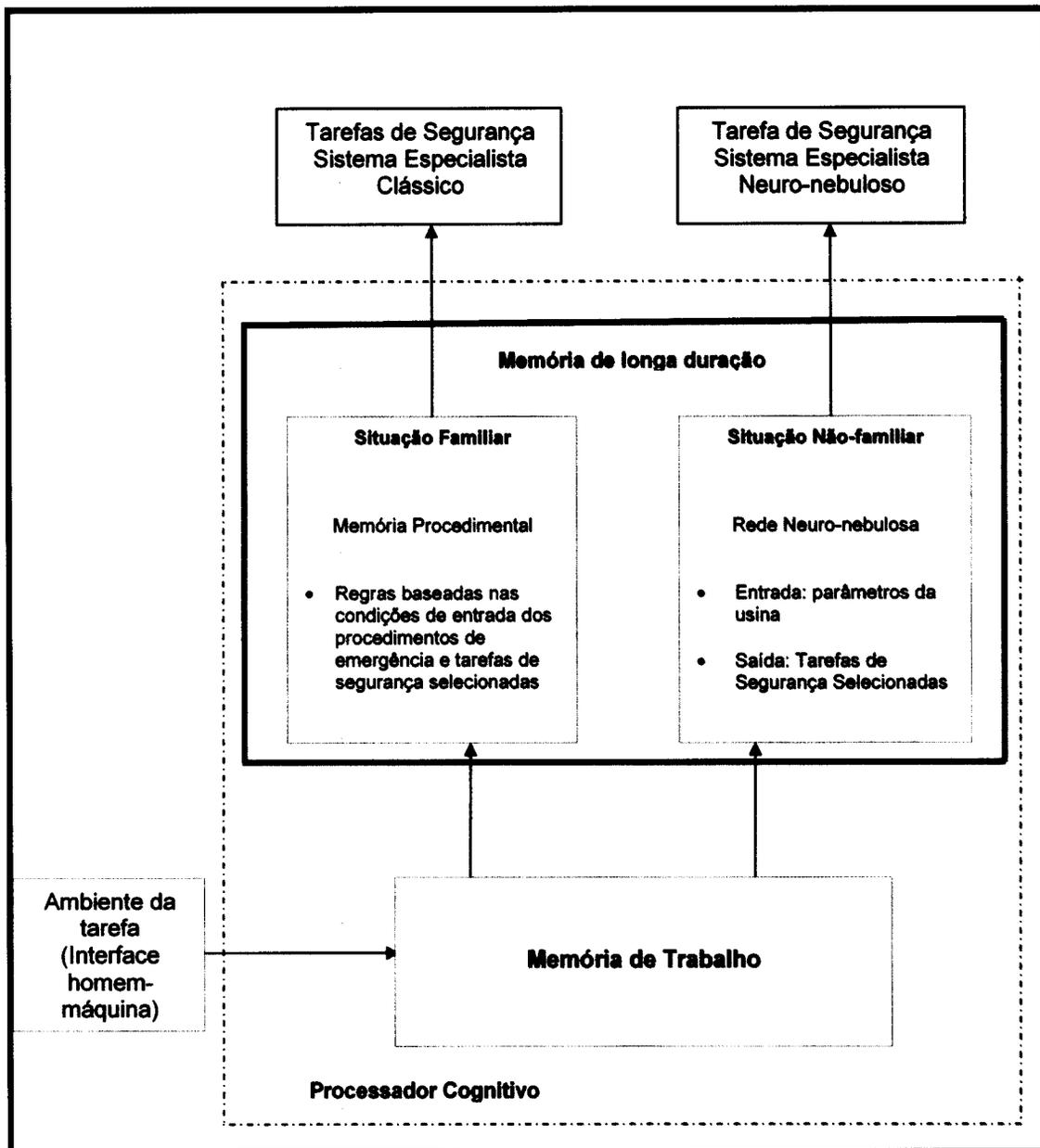


Figura 3 – Esquema do Sistema de Suporte

O sistema proposto tem como entrada 19 parâmetros operacionais de uma usina do tipo Reator a Água Pressurizada (PWR) e, de acordo com os valores assumidos por esses parâmetros durante um transitório operacional ou um acidente, deverá indicar uma tarefa de segurança a ser executada pelo operador da Sala de Controle. Estes parâmetros de entrada, apresentados na Tabela 2, foram selecionados por serem os que determinam as ações do operador nos Procedimentos de Operação em Emergência.

Tabela 2 – Parâmetros de Entrada do Sistema

Código	Parâmetro
P2	Pressão do Pressurizador
P3	Temperatura média do refrigerante
C4	Pressão no envoltório de contenção
C2	Nível no poço do envoltório de contenção
G4	Pressão no Gerador de Vapor 1
G5	Pressão no Gerador de Vapor 2
G2	Nível no Gerador de Vapor 1
G3	Nível no Gerador de Vapor 2
C5	Nível no Tanque de Água de Carregamento
G6	Vazão de vapor no GV 1
G7	Vazão de vapor no GV 2
N2	Taxa de partida na Faixa da Fonte da instrumentação nuclear
N4	Taxa de partida na Faixa Intermediária da instrumentação nuclear
N6	Fluxo de nêutrons na Faixa de Potência
C6	Vazão de Água de Alimentação Auxiliar GV 1
C7	Vazão de Água de Alimentação Auxiliar GV 2
P5	Grau de sub-resfriamento do refrigerante do reator
R3	Nível de radioatividade no GV 1
R4	Nível de radioatividade no GV 2

Inicialmente, os Procedimentos de Operação em Emergência da usina Angra 1 foram analisados e suas instruções sequenciais agrupadas de acordo com seus

objetivos finais, em unidades denominadas Tarefas de Segurança. A Tabela 3 apresenta como exemplo, a composição da Tarefa de Segurança INJEÇÃO DE ALTA PRESSÃO. As composições das demais tarefas são apresentadas no Apêndice 1.

Tabela 3 – Composição da Tarefa “Injeção de Alta Pressão”

AÇÃO/RESPOSTA ESPERADA	RESPOSTA NÃO OBTIDA
<p>Verifique Se a Injeção De Segurança Foi Atuada:</p> <p>a. Janela "ATUAÇÃO IS" no painel BPLB (permissivos) - ACESA.</p>	<p>a. Verifique se a injeção de segurança é requerida por:</p> <ul style="list-style-type: none"> - Pressão no PZR menor que 123 Kg/cm². <p style="text-align: center;"><u>OU</u></p> <ul style="list-style-type: none"> - Pressão na(s) linha(s) de vapor menor que 42 Kg/cm². <p style="text-align: center;"><u>OU</u></p> <ul style="list-style-type: none"> - Pressão na contenção maior que 0,28 Kg/cm². <p><u>SE</u> for requerida,</p> <p><u>ENTÃO</u> atue-a manualmente.</p>
<p>b. Bombas de Injeção de Segurança – AMBAS OPERANDO</p>	<p>b. Atue manualmente a injeção de segurança</p>
<p>Confirme o Alinhamento Correto dos Sistemas Atuados Pelo Sinal de Injeção de Segurança:</p> <p>Grupos 3 e 4 - Sinais S/T - ACESOS.</p> <p>Grupos 1,2 e 5 – Sinais S/T -APAGADOS.</p>	<p>Faça o seguinte:</p> <ul style="list-style-type: none"> a. Atue manualmente o Sinal de Isolação da Contenção Fase A. b. Se necessário, alinhe manualmente os equipamentos.
<p>Confirme Fluxo de Injeção de segurança:</p> <p>Confirme indicação de fluxo pelos FI-926 <u>E</u> FI-927 (neste, após a abertura das válvulas 8815A/8815B).</p>	<p>a. Parta manualmente as bombas de injeção de segurança</p> <p style="text-align: center;"><u>E</u></p> <p>Alinhe o sistema para injeção nas pernas frias e no vaso do reator.</p>

As tarefas de segurança de saída do sistema são mostradas na Tabela 4. Essas tarefas são as requeridas para se combater os acidentes analisados no Capítulo 15 do Relatório Final de Análise de Segurança de Angra 1 [3] que trata da análise de acidentes.

Tabela 4 – Tarefas de Segurança

Tarefa	Descrição
1	Injeção de Segurança de Alta Pressão
2	Última Fonte Fria
3	Ventilação da Área Controlada em Modo de Emergência
4	Ventilação da Sala de Controle em Modo de Emergência
5	Injeção de Água Borada
6	Injeção do Acumulador
7	Injeção de Segurança de Baixa Pressão
8	Recirculação do Poço da Contenção
9	Resfriamento Rápido do Sistema de Refrigeração do Reator
10	Despressurização Rápida do Sistema de Refrigeração do Reator
11	Água de Alimentação Auxiliar
12	Isolamento da Contenção
13	Isolamento da Ventilação da Contenção
14	Resfriamento da Contenção
15	Aspersão da Contenção
16	Isolamento da Água de Alimentação
17	Isolamento das Linhas de Vapor Principal
18	Resfriamento do Reator por Drenagem e Alimentação
19	Isolamento de Gerador de Vapor Falho
20	Desligamento das Bombas de Refrigeração do Reator – Regra 1
21	Desligamento das Bombas de Refrigeração do Reator – Regra 2

3.1 Modelo Cognitivo Simplificado

O modelo cognitivo simplificado sobre o qual o sistema de suporte é construído é composto de um Dispositivo de Entrada de Dados que faz a função do Painel de Controle, disponibilizando ao sistema os valores dos diversos parâmetros da instalação. O Processador Perceptual faz a coleta da informação necessária à análise da situação operacional e envia os valores dos parâmetros à Memória de Trabalho, no qual esta informação é armazenada enquanto o sistema busca, na sua Memória de Longo Prazo, a regra que se aplica à situação operacional corrente.

A Memória de Longo Prazo do nosso modelo é composta de dois módulos, um referente às situações familiares ao operador, composto de um sistema especialista clássico, e outro referente às situações não familiares, composto de um sistema neuro-nebuloso. Cada módulo tem uma memória de saída para onde a tarefa de segurança selecionada é enviada.

A Figura 4 mostra o arranjo dos componentes do modelo cognitivo simplificado.

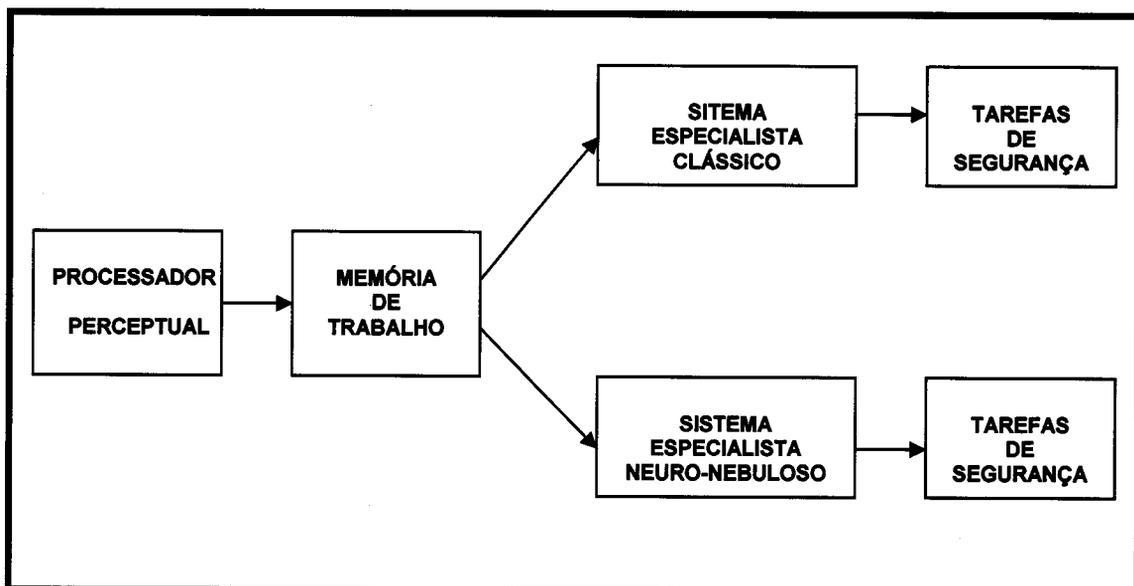


Figura 4 – Modelo cognitivo simplificado

3.2 Construção do Sistema de Suporte no COGENT

Para a implementação do projeto conceitual do sistema utilizamos um *software* denominado COGENT [37], que oferece grande facilidade para a construção de sistemas controlados por regras do tipo SE-ENTÃO, como o que é utilizado pelo módulo do sistema de suporte correspondente aos contextos familiares ao operador.

O COGENT é um ambiente computacional projetado especificamente para dar suporte ao desenvolvimento de modelagem cognitiva e é gratuito para uso acadêmico. O sistema foi desenvolvido como um modelo cognitivo simplificado do operador. Este desenvolvimento permite que o sistema possa, também, ser utilizado em outras pesquisas, nas quais este modelo cognitivo seja necessário. As memórias do COGENT permitem a especificação de parâmetros internos, tais como uma constante de decaimento da informação armazenada e um limite na capacidade de armazenamento da informação. Por meio da especificação desses parâmetros pode-se simular o operador humano, para pesquisa, por exemplo. As memórias, para nossa aplicação, serão de capacidade ilimitada e sem provisão de decaimento da informação armazenada. Além disso, este tipo de desenvolvimento poderá facilitar a comunicação com o usuário, por exemplo, apresentando na interface do sistema não somente a tarefa ativada, mas, também, o que gerou esta conclusão, ou seja, o conteúdo da Memória de Trabalho.

3.2.1 Subsistema Controlado por Regras

Na parte do modelo referente ao controle por regras SE-ENTÃO, correspondente aos CONTEXTOS FAMILIARES, as regras com as condições de entrada especificadas com valores rígidos dos parâmetros foram programadas no processador denominado BASE DE REGRAS. A Figura 5 mostra a Regra Injeção de Alta Pressão, como exemplo.

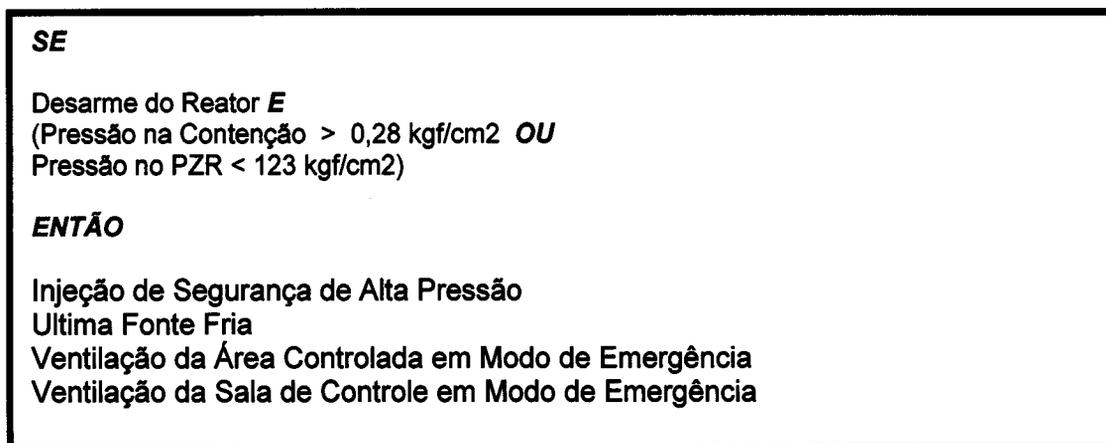


Figura 5 – Exemplo de uma Regra do Sistema Especialista

A Figura 6 mostra o arranjo dos componentes do subsistema controlado por regras construído no COGENT. A parte referente à ENTRADA DE DADOS e ao PROCESSADOR PERCEPTUAL é comum a ambos os módulos da MEMÓRIA DE LONGO PRAZO.

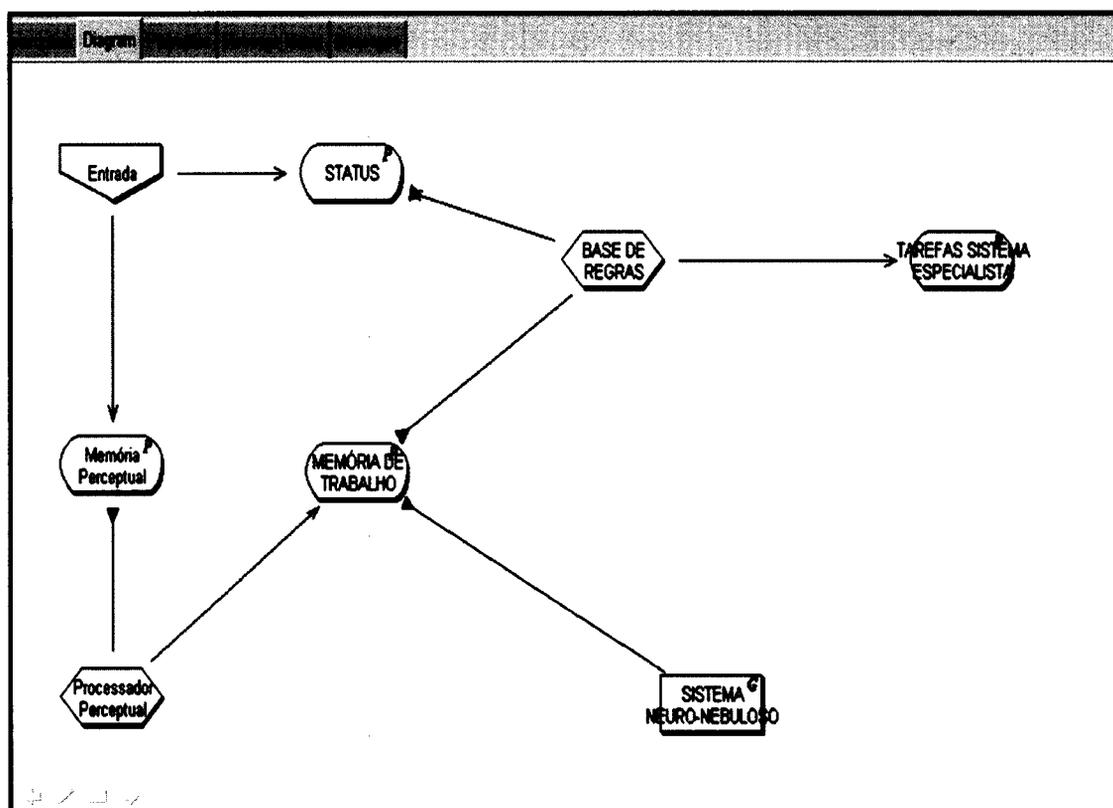


Figura 6 – Sistema Controlado por Regras

Ao ser acionado o sistema, os valores dos parâmetros de entrada coletados pelo PROCESSADOR PERCEPTUAL são transferidos para a MEMÓRIA DE TRABALHO, bem como a mensagem DESARME DO REATOR é enviada à memória STATUS. Esta mensagem inicia o funcionamento sistema, ela é colocada nessa memória para evitar que tenha que ser repetida em cada regra. O processador BASE DE REGRAS é acionado quando os dados são armazenados na MEMÓRIA DE TRABALHO. Este, então, verifica na base de regras se os valores das variáveis coincidem com as condições de alguma regra ali armazenada e, caso positivo, envia a Tarefa de Segurança constante da parte ENTÃO desta regra para a memória TAREFAS DO SISTEMA ESPECIALISTA.

A Figura 7 mostra um exemplo do vetor de entrada de dados que contém os valores dos parâmetros para a determinada situação da usina, conforme programado no dispositivo de entrada do COGENT.

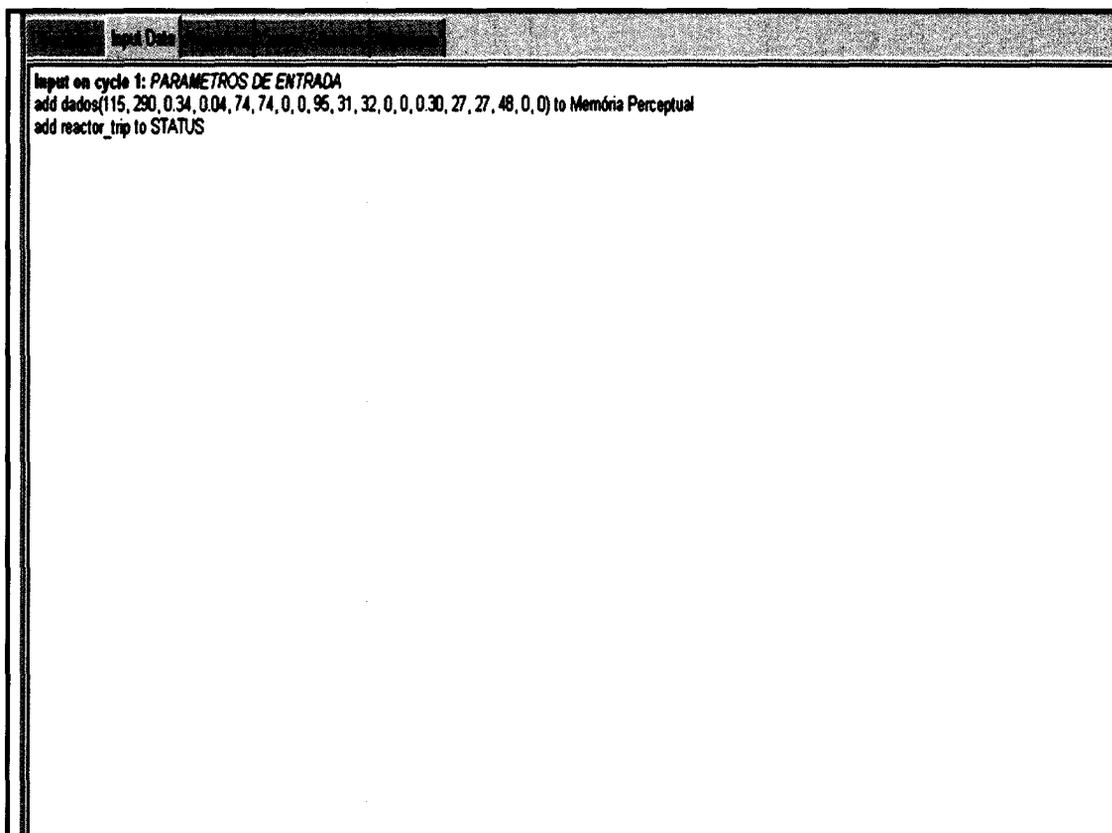


Figura 7 – Vetor de Entrada de Dados

A Figura 8 mostra uma tela do COGENT, apresentando, como exemplo, duas das regras extraídas dos Procedimentos de Operação em Emergência, para mostrar como estas são programadas no processador BASE DE REGRAS.

```
Rules & Condition Definitions

Rule 1 (refractor): IS bx pressao pzz - R-16
IF: dados(P2, P3, C4, C2, G4, G5, G2, G3, C5, G6, G7, N2, N4, N6, C6, C7, P5, R3, R4) is in MEMÓRIA DE TRABALHO
reactor_trip is in STATUS
P2 is less than 123
THEN: add injecao_alta_pressao to TAREFAS SISTEMA ESPECIALISTA
add ultima_fonte_fria to TAREFAS SISTEMA ESPECIALISTA
add ventilacao_area_controlada_modos_emergencia to TAREFAS SISTEMA ESPECIALISTA
add ventilacao_sala_controle_modos_emergencia to TAREFAS SISTEMA ESPECIALISTA
add safety_injection to STATUS

Rule 2 (refractor): IS bx pressao LVP - R-01
IF: dados(P2, P3, C4, C2, G4, G5, G2, G3, C5, G6, G7, N2, N4, N6, C6, C7, P5, R3, R4) is in MEMÓRIA DE TRABALHO
reactor_trip is in STATUS
G4 is less than 42
THEN: add injecao_alta_pressao to TAREFAS SISTEMA ESPECIALISTA
add ultima_fonte_fria to TAREFAS SISTEMA ESPECIALISTA
add ventilacao_area_controlada_modos_emergencia to TAREFAS SISTEMA ESPECIALISTA
add ventilacao_sala_controle_modos_emergencia to TAREFAS SISTEMA ESPECIALISTA
add safety_injection to STATUS
```

Figura 8 – Regras do Processador BASE DE REGRAS

A Figura 9 mostra um exemplo de tela do COGENT com as mensagens recebidas pela memória TAREFAS DO SISTEMA ESPECIALISTA, referentes a 5 tarefas de segurança ativadas pelo processador BASE DE REGRAS, baseado na coincidência das condições dessas regras com os valores das variáveis de entrada.

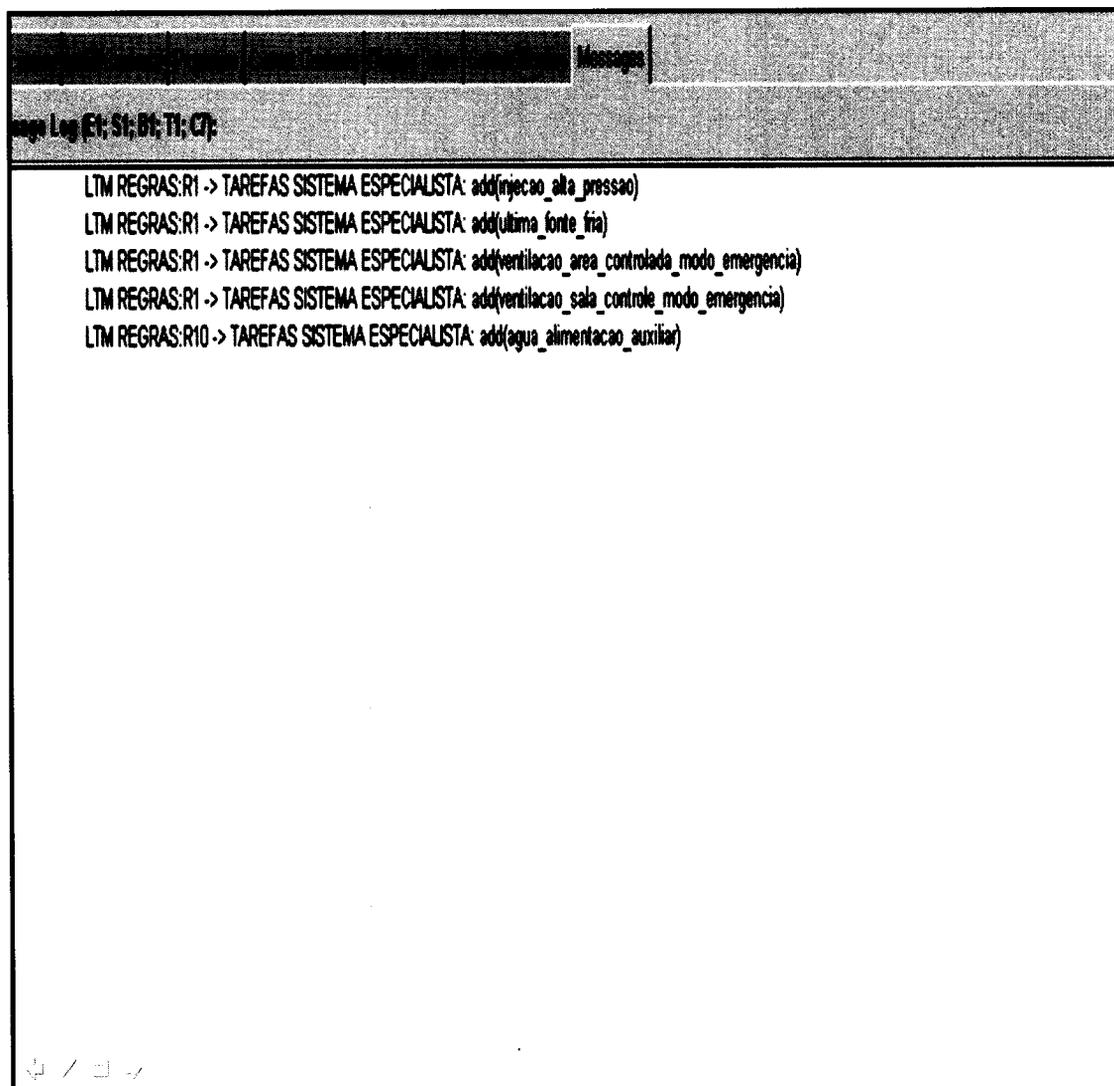


Figura 9 – Exemplo de saída de Tarefa de Segurança

3.2.2 Subsistema Controlado pela Rede Neuro-Nebulosa

Neste módulo, correspondente às situações *não familiares* ao operador, a determinação das Tarefas de Segurança aplicáveis ao conjunto de valores dos parâmetros de entrada coletados pelo PROCESSADOR PERCEPTUAL e enviados à MEMÓRIA DE TRABALHO, será feita por uma rede neuro-nebulosa. Em oposição ao módulo controlado por regras, que trabalha com valores rígidos dos parâmetros, este módulo busca tirar proveito das propriedades teóricas previstas para os Sistemas Nebulosos e Redes Neurais, principalmente aqueles relacionados à operação com valores linguísticos, robustez e generalização.

3.2.3 Topologia da Rede Neural Utilizada

A Figura 10 mostra a topologia da rede a ser utilizada nesta parte referente aos contextos *não familiares*. A rede é do tipo Radial Basis Function (RBF), *feedforward*, de cinco camadas. Ela é encontrada na referência [9] onde é usada para uma função de controle.

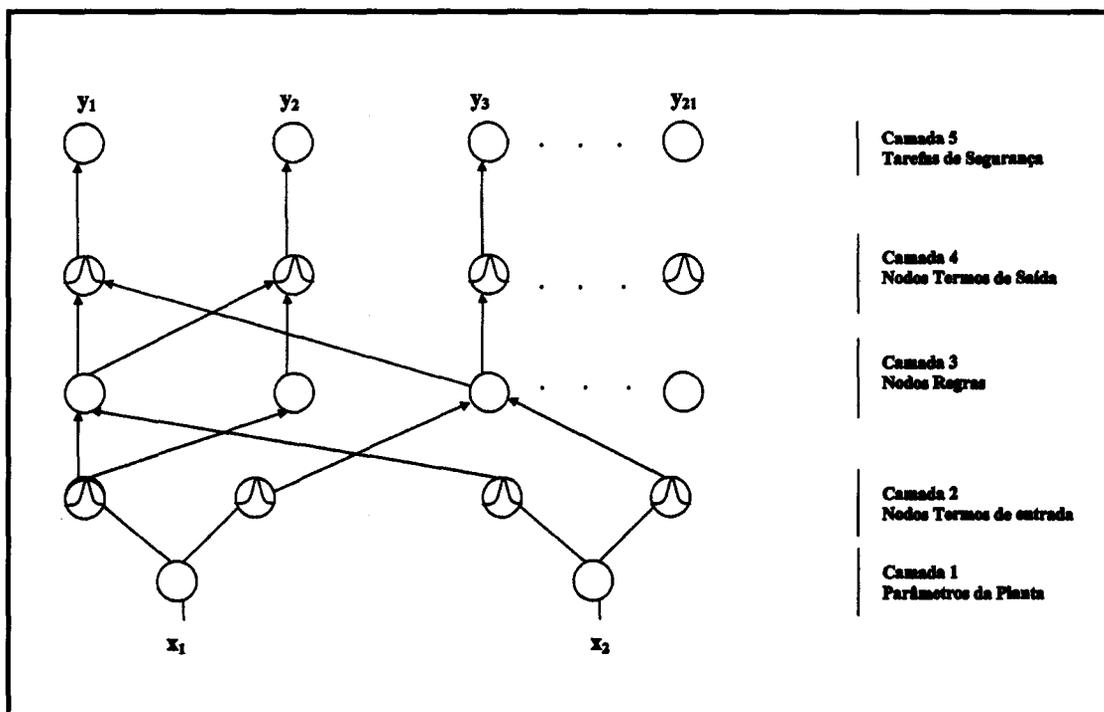


Figura 10 – Topologia da Rede

Para nossa utilização ela foi modificada na sua arquitetura original para ser utilizada na função de classificação. Nesta aplicação, as funções básicas das camadas terão funções similares às da referência. Os nodos da primeira camada representam as variáveis de entrada. Os nodos da camada 2 representam as partições em conjuntos nebulosos de cada variável de entrada. Cada um dos nodos da camada 3 define uma regra nebulosa, portanto esta camada representa a base de regras. As conexões da camada 3 definem os termos SE das regras nebulosas. As conexões da camada 4 definem o ENTÃO das regras nebulosas. Cada nodo dessa camada possui um conjunto nebuloso que se conecta a um nodo da camada 5 que representa um grupo de tarefas de segurança com as mesmas condições, portanto esta camada representa o conjunto de tarefas de segurança. Durante a operação o sinal flui na direção da camada 5. A “defuzzyficação” é feita na camada 5, conforme explicado adiante.

A rede foi testada com duas arquiteturas diferentes, a primeira com dois conjuntos nebulosos na camada 2, para cada variável, e outra com quatro conjuntos nebulosos para cada variável. Verificamos que para algumas tarefas obtinham-se melhores resultados, em termos de acertos e erros, com a utilização de dois conjuntos. Para os parâmetros de entrada que contribuíam para o disparo de dois grupos de tarefas de segurança, cada um com um ponto de ajuste diferente, os resultados eram melhores com a utilização de quatro conjuntos nebulosos, sendo dois para cada grupo. Também utilizamos esta estratégia para parâmetros que apresentavam oscilações no comportamento da variável. Optamos, então, por utilizar quatro conjuntos para os parâmetros: Pressão do Sistema de Refrigeração do Reator, Pressão do Gerador de Vapor 1, Pressão do Gerador de Vapor 2, Vazão de Vapor Gerador de Vapor 1, Vazão de Vapor Gerador de Vapor 2 e Grau de Sub-resfriamento de Sistema de Refrigeração do Reator. Para os demais parâmetros foram utilizados dois conjuntos na camada 2 para cada um.

3.2.4 Equacionamento da Rede Neural Utilizada

A rede que utilizaremos tem a arquitetura que passamos a descrever.

Os sinais de entrada na camada 1 são os dezenove parâmetros da usina apresentados na Tabela 2. Os nodos da camada 1 transmitem o sinal diretamente

para a camada 2. Deste modo os pesos das conexões dos nodos da camada 1 com os nodos da camada 2, w_{12} , são iguais a 1.

$$w_{12}(i,j) = 1 \quad (3.1)$$

onde i corresponde ao parâmetro e j ao conjunto nebuloso.

Os nodos da camada 2 fazem a "fuzzyficação" dos parâmetros em dois ou quatro conjuntos nebulosos, tendo como função de ativação uma gaussiana.

A saída $a_2(i,j)$ de cada nodo da camada 2 é:

$$a_2(i,j) = e^f \quad (3.2)$$

com

$$f = - (x_i - m_{i,j})^2 / \sigma_{i,j}^2 \quad (3.3)$$

onde $m_{i,j}$ e $\sigma_{i,j}$ são, respectivamente, o centro e a largura da gaussiana do conjunto nebuloso j da variável x_i

A rede foi testada com todos os nodos da camada 2 interconectados com todos os nodos da camada 3 no intuito de obter uma maior robustez. Verificamos que esta arquitetura não convergia para a saída desejada. Deste modo, modificamos a rede e atribuímos o valor dos pesos das conexões entre as camadas 2 e 3 a um e zero, quando o parâmetro contribui para a ativação da regra ou não, respectivamente. Quando o parâmetro participa da regra de ativação da tarefa o nodo correspondente da camada 2 transmite o sinal diretamente para a camada 3, com um valor de $w_{23}(i,j)$ igual a um. Caso contrário o valor da conexão é zero. Isto pode ser justificado pelo fato de que, diferentemente da aplicação da referência, já são conhecidas de antemão as regras que ativam as tarefas de segurança, extraídas dos Procedimentos de Operação em Emergência.

A Figura 11 mostra os conjuntos nebulosos do parâmetro Pressão do Sistema de Refrigeração do Reator para as tarefas de segurança 1, 2, 3 e 4, conforme Tabela 4.

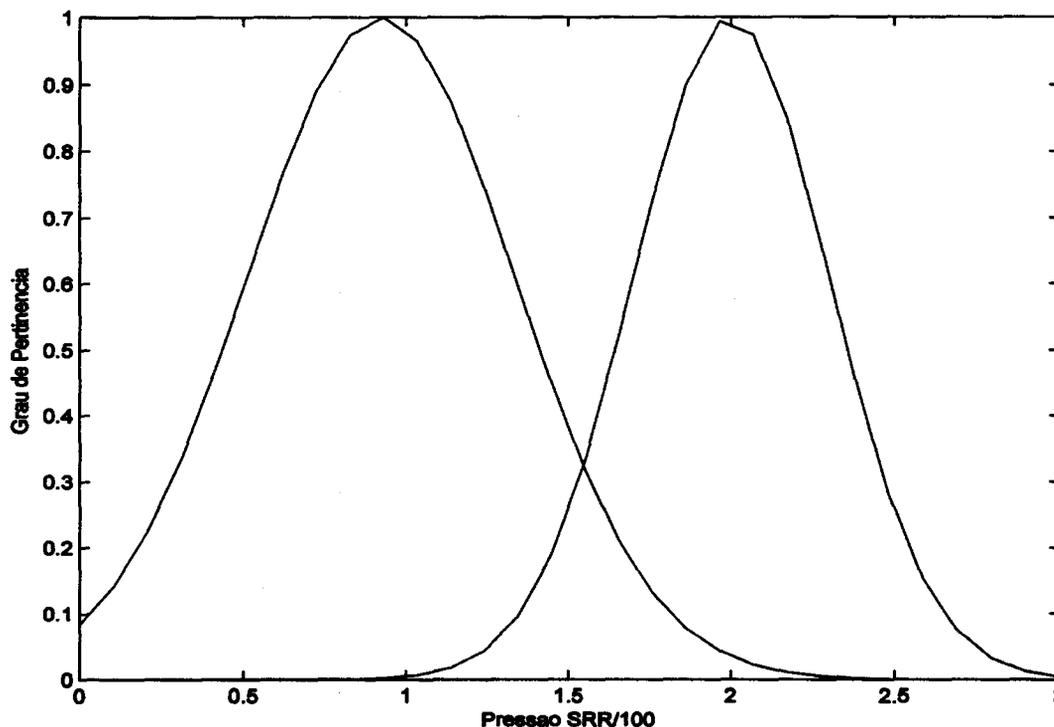


Figura 11 – Conjuntos Nebulosos do Parâmetro Pressão do Sistema de Refrigeração do Reator para as Tarefas 1, 2, 3, 4

As conexões w_{23} da camada 2 com a camada 3 são as partes SE das regras. Na aplicação da referência os nodos da camada 3 fazem a função do operador E tendo sido escolhida pelo autor como função de ativação uma das funções MÍNIMO ou PRODUTO. A aplicação da referência [9] tem a função de controle, com parâmetros de entrada contínuos, como na nossa aplicação, porém com saída de uma variável contínua diferentemente da nossa aplicação que é discreta (0 ou 1).

No nosso caso, na parte SE, as regras são compostas pelos valores dos parâmetros relacionados por um operador E, OU, ou ambos. Após vários testes, verificamos que a rede apresentou melhores resultados com as regras rearranjadas fatorando-se os operadores OU, conforme a Figura 12. Esta fatoração permitiu que regras simples fossem criadas, sem operadores lógicos na parte SE, deixando-se o

restante da regra original com uma combinação de operadores lógicos E e OU complexa. Isto é mais coerente com as funções lógicas MÍNIMO ou PRODUTO, já que estas funções são equivalentes ao operador lógico E.

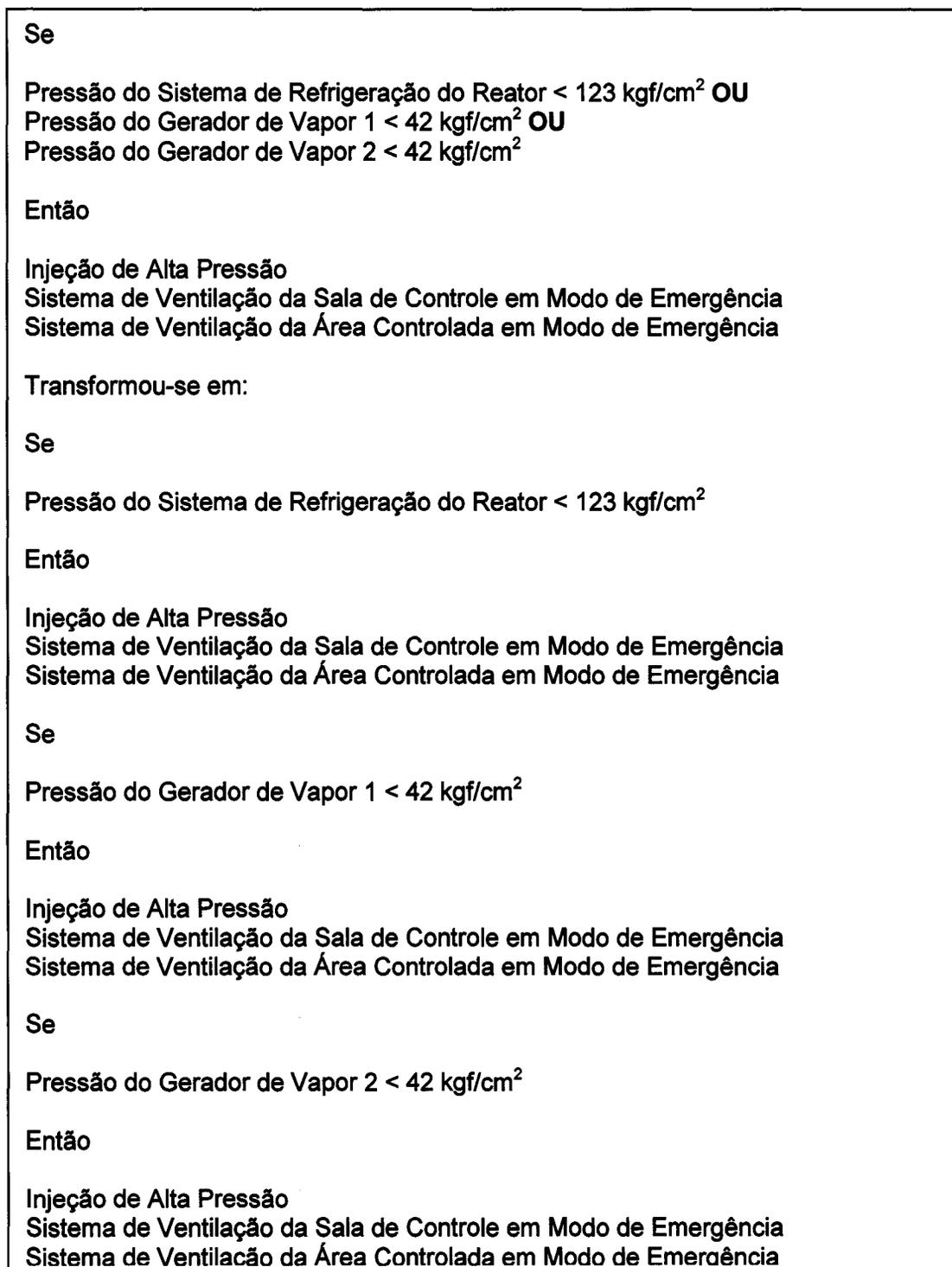


Figura 12 – Fatoração dos Operadores OU nas Regras

Verificamos também, que durante o treinamento com retro propagação, nas regras contendo o operador E, seria necessário utilizar a função de ativação MÍNIMO DOS MÁXIMOS dos dois ou quatro conjuntos nebulosos referentes a cada parâmetro que contribui para a ativação da regra. Isto foi feito porque ao utilizarmos a função MÍNIMO ou PRODUTO, conforme a referência, um valor muito pequeno de saída da camada 2 sempre prevalecia sobre os demais, prejudicando o funcionamento do nodo. Quando utilizamos a rede para diagnóstico, verificamos que seria necessário utilizar como função de ativação o PRODUTO DOS MÁXIMOS para evitar esta prevalência. Entretanto, na retro propagação o que se busca é exatamente identificar a variável e o conjunto nebuloso que contribuem mais para o erro a fim de diminuí-lo.

Essas modificações constituem a interferência do especialista no domínio para a adaptação da ferramenta para a solução deste problema de engenharia.

A função de ativação da camada 3 para o algoritmo de retro propagação é calculada:

$$a_3(n) = \min\{\max[a_2(i,j) * \omega_{23}(i,j)]\} \quad (3.4)$$

onde o MÁXIMO se refere ao valor máximo de saída dos conjuntos nebulosos de cada variável e o MÍNIMO ao mínimo desses máximos.

Para o diagnóstico utilizamos a equação:

$$a_3(n) = \Pi\{\max[a_2(i,j) * \omega_{23}(i,j)]\} \quad (3.5)$$

ou seja, o produto de todos os máximos, conforme definido anteriormente, sendo que i corresponde aos parâmetros que compõem a regra n e j ao conjunto nebuloso de cada parâmetro i .

Na camada 4 a função de ativação dos nodos é igual à da referência:

$$a_{34}(k) = \sum_n [a_3(n)] \quad (3.6)$$

$$a_4(k) = \min(a_{34}(k), 1) \quad (3.7)$$

onde $a_{34}(k)$ corresponde ao somatório das saídas (para o operador lógico OU) de todos os nodos n da camada 3 que estão conectadas ao nodo k correspondendo ao grupo de tarefas de segurança específico. Como existe uma regra para cada grupo de tarefas de segurança, cada grupo específico tem as mesmas condições em comum que compõem a regra de origem. Neste caso, os termos da diagonal da matriz $w_{34}(n,k)$ são unitários e fora da diagonal são nulos durante o treinamento não supervisionado.

Durante o treinamento não supervisionado competitivo, o cálculo da matriz $w_{34}(n,k)$ é executado supondo que todas as regras influenciam todos os grupos de tarefas de segurança. Assim, além do grupo disparado pela sua regra específica, esta também poderia capturar a correlação estatística entre o grupo de tarefas de segurança disparada e outros grupos de tarefas que fossem disparadas por outras regras no mesmo instante do tempo [41]. Isto poderia servir como um elemento de robustez quando as variáveis que compõem uma determinada regra apresentassem valores errôneos devido à instrumentação, já que outras regras disparadas poderiam induzir o grupo de tarefa pertencente ao grupo da regra que falhou. Entretanto, este uso da matriz $w_{34}(n,k)$ necessita de uma adaptação da rede neural proposta, que funciona bem para sistemas de controle, para os problemas de identificação, neste caso, de tarefas de segurança. O uso da $w_{34}(n,k)$ neste trabalho é explicado mais adiante.

A camada 5 executa a “defuzzyficação”. Cada nodo da camada 5 representa um grupo de tarefas de segurança e recebe o sinal de saída a_4 da camada 4. Foi verificado por meio de testes que quando utilizado o valor de 0,79 na função de ativação dos nodos da camada 5, a rede apresentava melhor desempenho. Assim, quando o sinal a_4 for maior que 0,79 a saída do nodo da camada 5 será igual a 1 e, quando a_4 for menor que 0,79, a saída será 0.

A função de ativação dos nodos da camada 5 é:

$$a_4 \geq 0,79 \rightarrow a_5 = 1 \quad (3.8)$$

$$a_4 < 0,79 \rightarrow a_5 = 0 \quad (3.9)$$

Deste modo, a tarefa será ativada se seu nodo correspondente na camada 5 for igual a 1 e, opostamente, não será ativada se o valor da saída for 0.

No cálculo da w_{34} utilizamos o sentido inverso do diagnóstico, ou seja da camada 5 para a camada 3, conforme Figura 10, com um conjunto nebuloso gaussiano, cujos m_i e σ_{ij} são pré-ajustados no treinamento não supervisionado e não competitivo. Isto será explicado mais adiante.

3.2.5 Treinamento da Rede Neural Utilizada

Para o treinamento da rede, os dados dos acidentes obtidos no simulador foram aplicados às regras do sistema especialista para definir as tarefas de segurança referentes a cada linha de dados de entrada. Deste modo, foram obtidos os pares de exemplos ENTRADA – SAÍDA para treinamento.

A rede foi inicialmente submetida a um treinamento não supervisionado e não competitivo que ajusta a estatística definindo os valores iniciais de $m(i,j)$ (valor médio de cada gaussiana) e $\sigma(i,j)$ (desvio padrão) para o treinamento não supervisionado competitivo e treinamento supervisionado com retropropagação de erro, feitos a seguir. Isto garante uma inicialização adequada do algoritmo de retro propagação, a fim de evitar instabilidades na convergência numérica e permitir uma convergência mais rápida aos resultados finais.

No treinamento não supervisionado competitivo são determinados os valores dos pesos das conexões ω_{34} entre as camadas 3 e 4, com a finalidade de impedir que tarefas não ativadas no acidente sejam treinadas pela rede, o que prejudica o treinamento das tarefas ativadas. Para impedir o treinamento das tarefas não ativadas o programa verifica se a norma da matriz ω_{34} composta pelos pesos das conexões entre as camadas 3 e 4, é igual a zero para essas tarefas.

Finalmente é feito o treinamento supervisionado com retro propagação do erro.

As equações aplicadas no treinamento da rede foram as utilizadas para este fim na referência [9]. O treinamento em retro propagação é feito pelo processo de redução do gradiente [9], minimizando-se a função de custo C :

$$C = \frac{1}{2} [y(t) - y'(t)]^2 \quad (3.10)$$

onde $y(t)$ é a saída desejada e $y'(t)$ é saída corrente. Inicia-se o treinamento de cada conjunto de dados nos nodos de entrada. Um percurso para a frente computa o nível de atividade de todos os nodos da rede. Então, partindo-se dos nodos de saída, um percurso para trás computa o gradiente $\delta C/\delta y'$ para todos os nodos intermediários. Se m é o parâmetro ajustável em um nodo (como o valor médio de uma função de ativação) então a regra geral de aprendizado é a redução do gradiente:

$$\Delta m \propto -\delta C/\delta m \quad (3.11)$$

$$m(t+1) = m(t) + \eta(-\delta C/\delta m) \quad (3.12)$$

onde η é a taxa de aprendizado e:

$$\delta C/\delta m = (\delta C/\delta a) * (\delta a/\delta m) \quad (3.13)$$

onde a é o sinal que chega ao nodo. Deste modo, $\delta C/\delta y'$ é então calculado camada por camada, começando-se pelos nodos de saída. Usamos o valor médio m_{ij} e a largura σ_{ij} da gaussiana como parâmetros de ajuste.

Na camada 5 a regra adaptativa do valor médio m_j é:

$$\delta C/\delta m_j = (\delta C/\delta a^{(5)}) * (\delta a^{(5)}/\delta m_j) = - [y(t) - y'(t)] \quad (3.14)$$

Assim, o valor atualizado esperado do parâmetro m é:

$$\delta m_j(t) = \eta [y(t) - y'(t)] \quad (3.15)$$

Da mesma forma as equações adaptativas de σ_{ij} são:

$$\delta C / \delta \sigma_j = (\delta C / \delta a^{(5)}) * (\delta a^{(5)} / \delta \sigma_j) = - [y(t) - y'(t)] \quad (3.16)$$

e o valor atualizado esperado do parâmetro σ_j é:

$$\delta \sigma_j(t) = \eta [y(t) - y'(t)] \quad (3.17)$$

Então, o erro propagado para a camada anterior é:

$$\delta^{(5)} = - \delta C / \delta a^{(5)} = y(t) - y'(t) \quad (3.18)$$

No modo “de baixo para cima” não há parâmetros a ajustar na camada 4 e computamos apenas o sinal de erro $\delta^{(4)}$.

$$\delta_i^{(4)} = - \delta C / \delta a_i^{(4)} = [y(t) - y'(t)] \quad (3.19)$$

Na camada 3, como na camada 4, computamos apenas o sinal de erro δ_4 :

$$\delta_i^{(3)} = - \delta C / \delta a_i^{(3)} = \delta C * \delta a_i^{(4)} / \delta a_i^{(4)} * \delta a_i^{(3)} = - \delta^{(4)} (\delta a_i^{(4)} / \delta x_i^{(4)}) = - \delta_i^{(4)} \quad (3.20)$$

onde $X_i^{(4)}$ é a saída da camada 5 calculada, entrando na camada 4.

Na camada 2 as equações (2), (3) e (13) dão a regra adaptativa dos m_{ij} :

$$\delta C / m_{ij} = (\delta C / \delta a_i^{(2)}) * e^f * 2(x_i^{(2)} - m_{ij}) / \sigma_{ij}^2 \quad (3.21)$$

onde de (3.20) :

$$\delta C / \delta a_j^{(2)} = \sum_k [(\delta C / \delta a_k^{(3)}) * (\delta a_k^{(3)} / \delta a_j^{(2)})] \quad (3.22)$$

$$\delta C / \delta a_k^{(3)} = - \delta_k^{(3)} \quad (3.23)$$

e, de (3.4):

$$\delta a_k^{(3)} / \delta a_i^{(2)} = \delta a_k^{(3)} / \delta x_i^{(3)} = z \quad (3.24)$$

$z = 1$, se $x_i^{(3)} = \min(\text{valor máximo das entradas do nodo da regra } k)$

$z = 0$, nos demais casos

Deste modo:

$$\delta C / \delta a_i^{(2)} = \sum_i q_k \quad (3.25)$$

onde o somatório é feito sobre os nodos das regras nas quais $a_i^{(2)}$ é recebido, e:

$$q_k = -\delta_k^{(3)} \text{ se } a_i^{(2)} \text{ é o mínimo das entradas na regra } k, \text{ ou} \quad (3.26)$$

$$q_k = 0 \text{ caso contrário.} \quad (3.27)$$

Deste modo, a regra adaptativa de m_{ij} é:

$$m_{ij}(t+1) = m_{ij}(t) - \eta(\delta C / \delta a_i^{(2)}) * e^{f(i)} * 2(x_i^{(2)} - m_{ij}) / \sigma_{ij}^2 \quad (3.28)$$

Similarmente, a regra adaptativa de σ_{ij} será:

$$\sigma_{ij}(t+1) = \sigma_{ij}(t) - \eta(\delta C / \delta a_i^{(2)}) * e^{f(i)} * 2(x_i^{(2)} - m_{ij})^2 / \sigma_{ij}^3 \quad (3.29)$$

No treinamento não supervisionado, fornecidos os dados de entrada de treinamento $x_i(t)$, os valores desejados de saída $y_i(t)$, as partições nebulosas $|T(x)$ e $|T(y)|$ e as formas desejadas das funções de ativação, o problema consiste em ajustar os m_i e σ_{ij} das funções de ativação e as regras nebulosas. Os nodos e conexões da camada 4 estão no modo de transmissão de cima para baixo, de modo que os dados de treinamento são alimentados pelos dois lados da rede. Nesta direção os nodos da camada 4 têm como função de ativação uma gaussiana e quando são utilizados para diagnóstico, de baixo para cima, as funções são determinadas pelas equações (3.6) e (3.7).

Inicialmente, selecionamos os valores médios e as larguras dos conjuntos nebulosos das funções de ativação por meio de um agrupamento estatístico. Esta forma de agrupamento espaço-produto move as funções de ativação para aquelas regiões do espaço ENTRADA-SAIDA em que os dados estão presentes.

$$\|x(t) - m_{\text{mais próximo}}(t)\| = \min \{\|x(t) - m_i(t)\|\} \quad (3.30)$$

$$m_{\text{mais próximo}}(t+1) = m_{\text{mais próximo}}(t) + \alpha(t)[x(t) - m_{\text{mais próximo}}(t)] \quad (3.31)$$

$$m_i(t+1) = m_i(t) \text{ para } m_i \neq m_{\text{mais próximo}} \quad (3.32)$$

onde $\alpha(t)$ é uma taxa de aprendizado escalar monotonicamente decrescente. Esta formulação adaptativa corre independentemente para cada parâmetro de entrada e de saída, supondo uma gaussiana para cada tarefa. A largura da gaussiana é determinada por:

$$\sigma_i = (m_i - m_{\text{mais próximo}}) / r \quad (3.33)$$

onde r é um parâmetro de superposição.

Na fase do treinamento não supervisionado competitivo as conexões da camada 4 estão completamente interconectadas. Denotaremos o peso entre o i -ésimo nodo de regra e o j -ésimo nodo termo de saída, referente a um grupo de tarefas, por $w_{34}(i,j)$. A seguinte equação realiza o aprendizado competitivo, atualizando esses pesos, iterativamente, para cada conjunto de dados de treinamento referente a um dado instante de tempo:

$$\delta w_{34}(i,j) / \delta t = a_j^{(4)} (a_i^{(3)} - w_{34}(i,j)) \quad (3.34)$$

Nesta equação, $a_i^{(3)}$ é calculada conforme as equações (3.4) e (3.5) e $a_j^{(4)}$ conforme a gaussiana correspondente à tarefa j . Observar que os m_i e $\sigma_{i,j}$ destas duas camadas (3 e 4) foram pré-ajustadas conforme o treinamento não-supervisionado e não-competitivo descrito acima.

3.2.6 Programação da Rede Neural na Linguagem MATLAB

A rede foi programada em MATLAB, um *software* produzido pela The MathWorks, Inc. Após o treinamento e determinação dos parâmetros foram extraídos os valores referentes aos pesos das conexões entre as camadas, e os valores médios e desvios padrão dos conjuntos nebulosos das camadas 2 e 4. Os valores obtidos para o valor médio $m_2(i,j)$ e para o desvio padrão $\sigma_2(i,j)$, onde i corresponde ao parâmetro e j ao conjunto nebuloso correspondente, estão nas Tabelas 5 e 6. A coluna PARÂMETRO corresponde à numeração da Tabela 2.

Tabela 5 – Valores da Média dos Conjuntos Nebulosos da Camada 2

Parâmetro	$m_2(i,1)$	$m_2(i,2)$	$m_2(i,3)$	$m_2(i,4)$
P2	0,5000	0,9238	0,9326	2,0000
P3	0,5000	0,7000	1,3861	2,0000
C4	0,5000	3,4560	0	0
C2	0,9986	2,1249	0	0
G4	-0,0199	5,0000	0	0
G5	-0,0199	5,0000	0	0
G2	0,0000	1,0000	2,0000	5,0000
G3	1,0000	0,0000	2,0000	5,0000
C5	1,0478	1,6991	0	0
G6	0,9992	7,1078	0	0
G7	-2,5301	-38,2819	0	0
N2	0,5000	2,0000	0	0
N4	0,5000	5,0000	0	0
N6	-0,4505	2,2625	0	0
C6	0,5826	0,7000	1,0000	2,0000
C7	0,5867	0,7000	1,0000	2,0000
P5	0,2522	0,0236	0,9403	0,8920
R3	1,0000	5,0000	0	0
R4	0,0757	5,0000	0	0

Tabela 6 – Valores dos Desvios Padrão dos Conjuntos Nebulosos da Camada 2

Parâmetro	$\sigma_2(i,1)$	$\sigma_2(i,2)$	$\sigma_2(i,3)$	$\sigma_2(i,4)$
P2	0,0794	0,7090	0,5866	0,4279
P3	0,1111	0,1111	0,4121	0,3417
C4	0,8333	1,8387	0	0
C2	0,3772	0,5498	0	0
G4	1,4343	2,5000	0	0
G5	0,5923	1,5236	0	0
G2	0,5556	0,5556	0,5556	1,6667
G3	0,5556	0,5556	1,1111	1,6667
C5	0,6272	0,1726	0	0
G6	4,9987	1,2004	0	0
G7	0,9466	0,0252	0	0
N2	0,8333	0,8333	0	0
N4	2,5000	2,5000	0	0
N6	0,1524	0,8648	0	0
C6	0,4140	0,0610	0,1667	0,5556
C7	0,1961	0,0610	0,1667	0,5556
P5	0,2653	0,4136	0,6165	0,6219
R3	2,2222	2,2222	0	0
R4	2,5980	2,5728	0	0

Para os parâmetros P2, P3, G2, G3, C6, C7 e P5 foram adotados 4 conjuntos nebulosos e para os demais 2 conjuntos para cada um. A razão desta escolha está detalhada no Capítulo TESTES E RESULTADOS.

O Apêndice 2 mostra em detalhe o programa de treinamento da rede.

3.2.7 Programação da Rede Neural no COGENT

Para a construção deste módulo no COGENT, cada camada da rede foi transformada em uma memória de armazenamento e um processador. Este processador lê os dados de entrada na memória e executa uma operação conforme as equações da rede, colocando o resultado na memória de armazenamento da camada seguinte, até a saída da camada 5 com a Tarefa de Segurança correspondente aos dados de entrada.

A Figura 13 mostra o arranjo das memórias e processadores no COGENT.

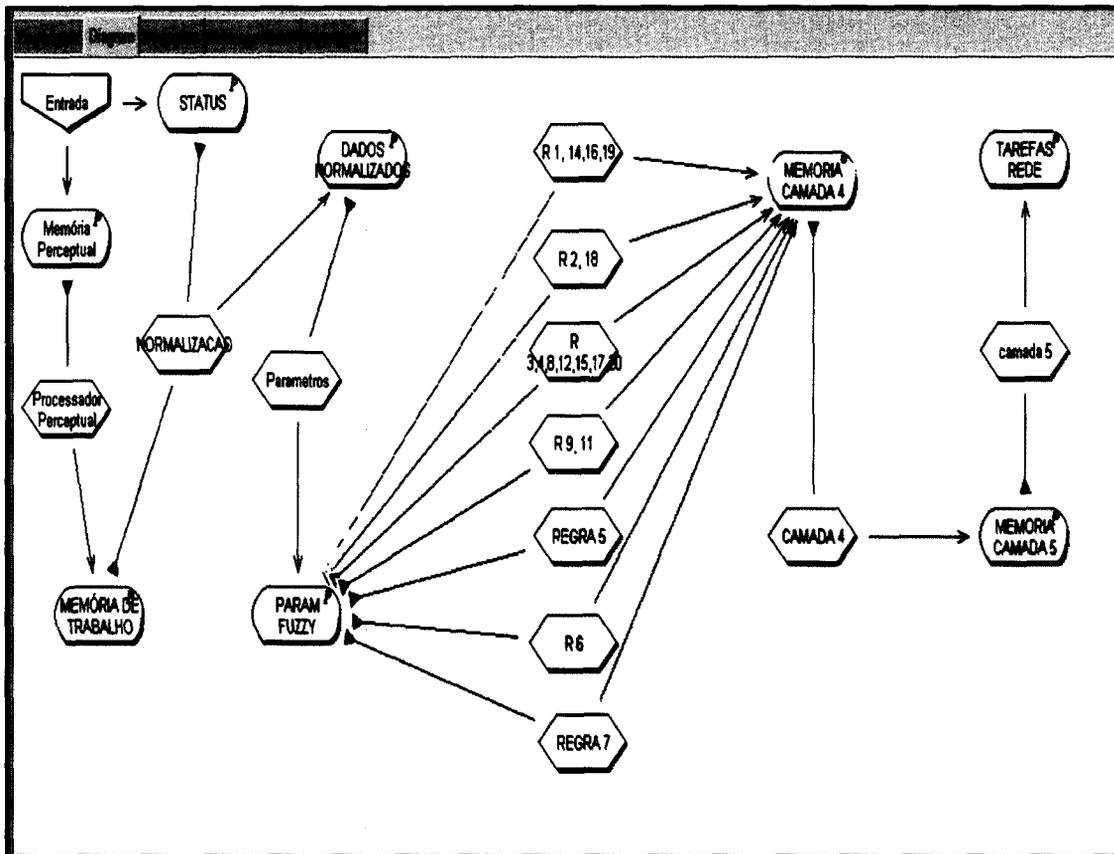


Figura 13 – Arranjo dos Componentes da Rede no COGENT

Quando os dados de entrada são depositados na MEMÓRIA DE TRABALHO, é acionado o processador NORMALIZAÇÃO, que executa a normalização dos dados de entrada e envia estes valores para a memória DADOS NORMALIZADOS, conforme mostrado na Figura 14. A normalização dos valores foi necessária porque há dados com valores muito pequenos, como Pressão na Contenção (0 a 1,6 kgf/cm²) e outros com valores muito altos, como Vazão de Vapor (0 a 2300 ton/h). Estas diferenças prejudicavam a convergência da rede.

```
(refract@): NORMALIZACAO
dados(P2, P3, C4, C2, G4, G5, G2, G3, C5, G6, G7, N2, N4, N6, C6, C7, P5, R3, R4) is in MEMÓRIA DE TRABALHO
reactor Trip is in STATUS
P2N is (P2) / (100)
P3N is (P3) / (200)
C2N is (C2) / (0.20)
C4N is (C4) * (1)
G2N is (G2) / (50)
G3N is (G3) / (50)
G4N is (G4) / (50)
G5N is (G5) / (50)
C5N is (C5) / (30)
G6N is (G6) / (10)
G7N is (G7) / (10)
N2N is (N2) / (0.20)
N4N is (N4) / (0.20)
N6N is (N6) / (10)
C6N is (C6) / (50)
C7N is (C7) / (50)
P5N is (P5) / (10)
R3N is (R3) / (1)
R4N is (R4) / (1)
add dados(P2N, P3N, C4N, C2N, G4N, G5N, G2N, G3N, C5N, G6N, G7N, N2N, N4N, N6N, C6N, C7N, P5N, R3N, R4N) to DADOS NORMALIZADOS
```

Figura 14 – Regras de Normalização dos Parâmetros

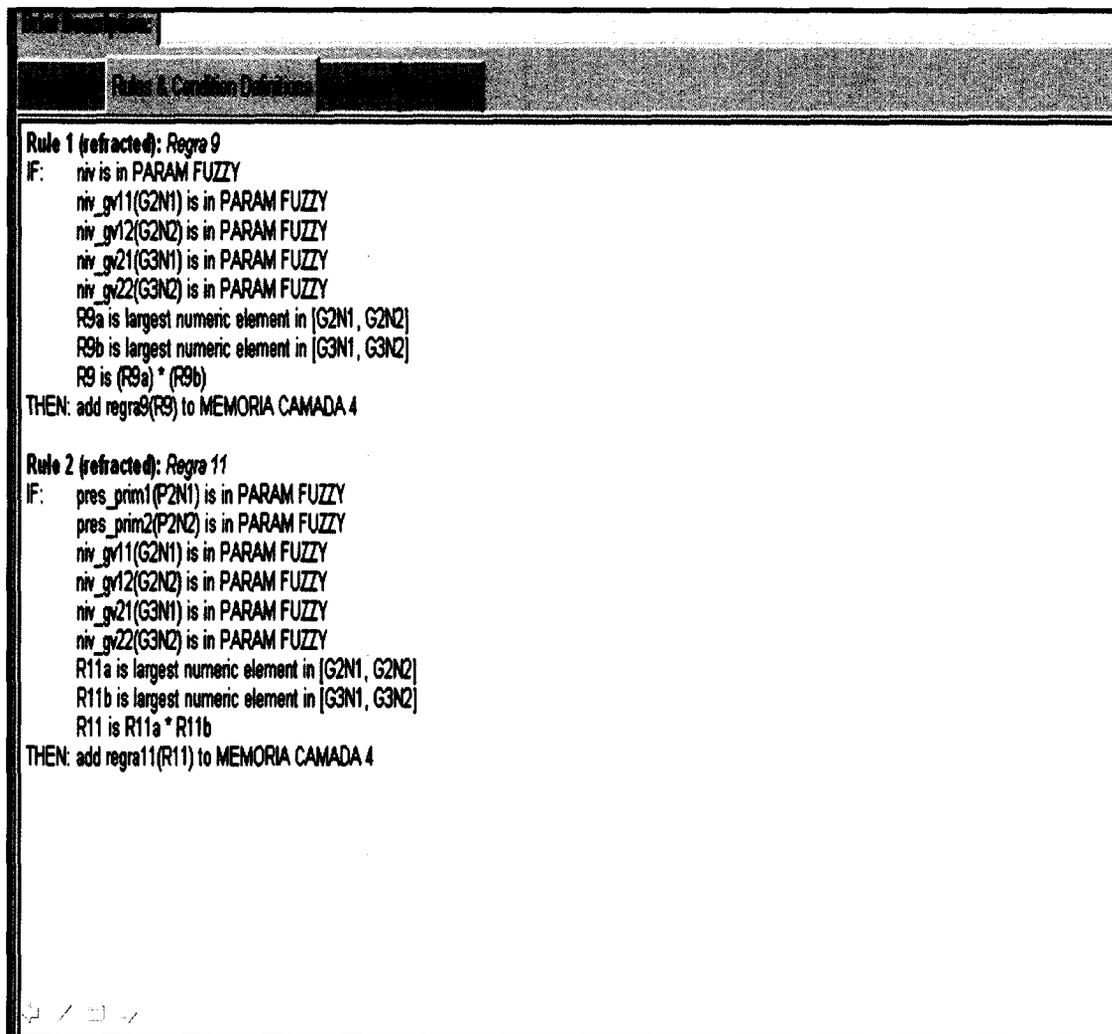
Quando os valores são armazenados na memória DADOS NORMALIZADOS, o processador PARAMETROS faz a “fuzzyficação” desses valores em dois ou quatro conjuntos nebulosos e envia os resultados para a memória PARAMETROS FUZZY, conforme mostrado na tela do COGENT na Figura 15. Esta operação corresponde à camada 2 da rede.

```
Rule 1 (refracted): P1-CN1
IF: dados(P2N, P3N, C4N, C2N, G4N, G5N, G2N, G3N, C5N, G6N, G7N, N2N, N4N, N6N, C6N, C7N, P5N, R3N, R4N) is in DADOS NORMALIZADOS
    A11a is (P2N - 0.87) / (0.61 * 0.61)
    A11b is A11a * A11a
    P2N1 is e to the power - A11b
THEN: add pres_prim1(P2N1) to PARAM FUZZY

Rule 2 (refracted): P1-CN2
IF: dados(P2N, P3N, C4N, C2N, G4N, G5N, G2N, G3N, C5N, G6N, G7N, N2N, N4N, N6N, C6N, C7N, P5N, R3N, R4N) is in DADOS NORMALIZADOS
    A11a is (P2N - 0.93) / (0.61 * 0.61)
    A11b is A11a * A11a
    P2N2 is e to the power - A11b
THEN: add pres_prim2(P2N2) to PARAM FUZZY
```

Figura 15 – Regras de separação dos Parâmetros em conjuntos nebulosos

Quando os valores são armazenados na memória PARAMETROS FUZZY, cada um dos processadores REGRAS lê os valores referentes aos parâmetros que constam na condição de sua regra e faz o PRODUTO destes, enviando os resultados para a MEMÓRIA CAMADA 4. Isto é mostrado na tela do COGENT na Figura 16.



```
Rule 1 (refracted): Regra 9
IF: niv is in PARAM FUZZY
    niv_gv11(G2N1) is in PARAM FUZZY
    niv_gv12(G2N2) is in PARAM FUZZY
    niv_gv21(G3N1) is in PARAM FUZZY
    niv_gv22(G3N2) is in PARAM FUZZY
    R9a is largest numeric element in [G2N1, G2N2]
    R9b is largest numeric element in [G3N1, G3N2]
    R9 is (R9a) * (R9b)
THEN: add regra9(R9) to MEMORIA CAMADA 4

Rule 2 (refracted): Regra 11
IF: pres_prim1(P2N1) is in PARAM FUZZY
    pres_prim2(P2N2) is in PARAM FUZZY
    niv_gv11(G2N1) is in PARAM FUZZY
    niv_gv12(G2N2) is in PARAM FUZZY
    niv_gv21(G3N1) is in PARAM FUZZY
    niv_gv22(G3N2) is in PARAM FUZZY
    R11a is largest numeric element in [G2N1, G2N2]
    R11b is largest numeric element in [G3N1, G3N2]
    R11 is R11a * R11b
THEN: add regra11(R11) to MEMORIA CAMADA 4
```

Figura 16 – Exemplo de Processamento de uma Regra

Quando os valores são armazenados na MEMÓRIA CAMADA 4, o processador CAMADA 4 executa a operação mínimo entre os valores de saída de cada regra e a unidade e envia este valor para a MEMORIA CAMADA 5, conforme mostrado na tela do COGENT na Figura 17.

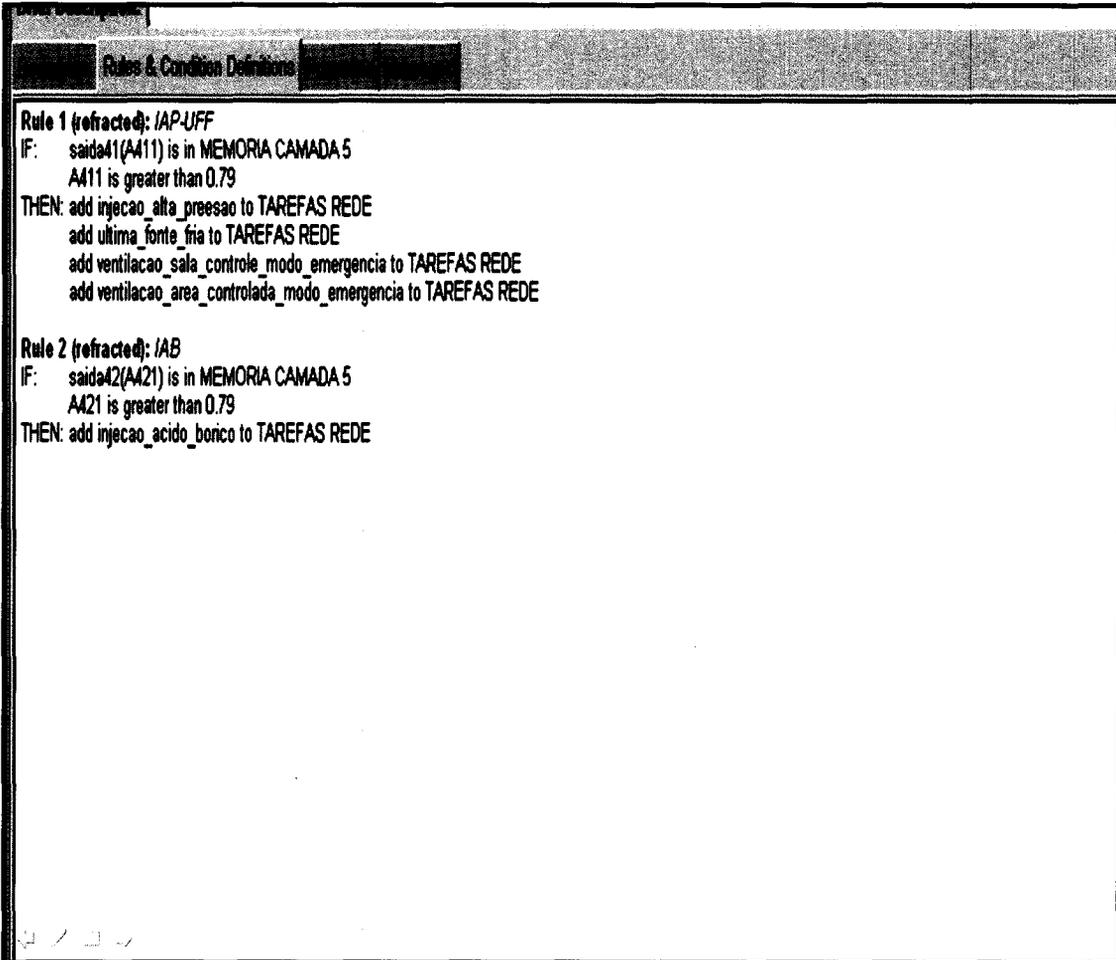
```
Rules & Condition Definitions

Rule 1 (refracted): C4R1
IF:  regra1(R1) is in MEMORIA CAMADA 4
     regra2(R2) is in MEMORIA CAMADA 4
     regra5(R5) is in MEMORIA CAMADA 4
     regra6(R6) is in MEMORIA CAMADA 4
     regra7(R7) is in MEMORIA CAMADA 4
     regra11(R11) is in MEMORIA CAMADA 4
     regra12(R12) is in MEMORIA CAMADA 4
     regra14(R14) is in MEMORIA CAMADA 4
     regra16(R16) is in MEMORIA CAMADA 4
     regra17(R17) is in MEMORIA CAMADA 4
     regra18(R18) is in MEMORIA CAMADA 4
     regra19(R19) is in MEMORIA CAMADA 4
     regra20(R20) is in MEMORIA CAMADA 4
     A411 is smallest numeric element in [R1, 1]
     A421 is smallest numeric element in [R2, 1]
     A451 is smallest numeric element in [R5, 1]
     A461 is smallest numeric element in [R6, 1]
     A471 is smallest numeric element in [R7, 1]
     A4111 is smallest numeric element in [R11, 1]
     A4121 is smallest numeric element in [R12, 1]
     A4141 is smallest numeric element in [R14, 1]
     A4161 is smallest numeric element in [R16, 1]
     A4171 is smallest numeric element in [R17, 1]
     A4181 is smallest numeric element in [R18, 1]
     A4191 is smallest numeric element in [R19, 1]
     A4201 is smallest numeric element in [R20, 1]
THEN: add saida41(A411) to MEMORIA CAMADA 5
      add saida42(A421) to MEMORIA CAMADA 5
      add saida45(A451) to MEMORIA CAMADA 5
      add saida46(A461) to MEMORIA CAMADA 5
      add saida47(A471) to MEMORIA CAMADA 5
      add saida411(A4111) to MEMORIA CAMADA 5
      add saida412(A4121) to MEMORIA CAMADA 5
      add saida414(A4141) to MEMORIA CAMADA 5
      add saida416(A4161) to MEMORIA CAMADA 5
```

Figura 17 – Exemplo do Cálculo da Camada 4

Ao perceber os valores armazenados na MEMORIA CAMADA 5, o processador CAMADA 5 verifica, para cada regra, se seu valor está acima de 0,79 e, caso positivo envia à memória TAREFAS DA REDE a Tarefa de Segurança correspondente. Este valor de corte foi determinado experimentalmente durante a adaptação da rede.

A Figuras 18 mostra um exemplo do processamento no COGENT do processador da CAMADA 5.



```
Rules & Condition Definitions

Rule 1 (refracted): IAP-UFF
IF: saida41(A411) is in MEMORIA CAMADA 5
    A411 is greater than 0.79
THEN: add injecao_alta_preesao to TAREFAS REDE
      add ultima_fonte_fria to TAREFAS REDE
      add ventilacao_sala_controle_modo_emergencia to TAREFAS REDE
      add ventilacao_area_controlada_modo_emergencia to TAREFAS REDE

Rule 2 (refracted): IAB
IF: saida42(A421) is in MEMORIA CAMADA 5
    A421 is greater than 0.79
THEN: add injecao_acido_borico to TAREFAS REDE
```

Figura 18 – Exemplo de Regras da Camada 5

A Figura 19 mostra uma tela do COGENT da memória TAREFAS DA REDE com um exemplo de saída de uma mensagem do processador CAMADA 5 com uma tarefa de segurança.

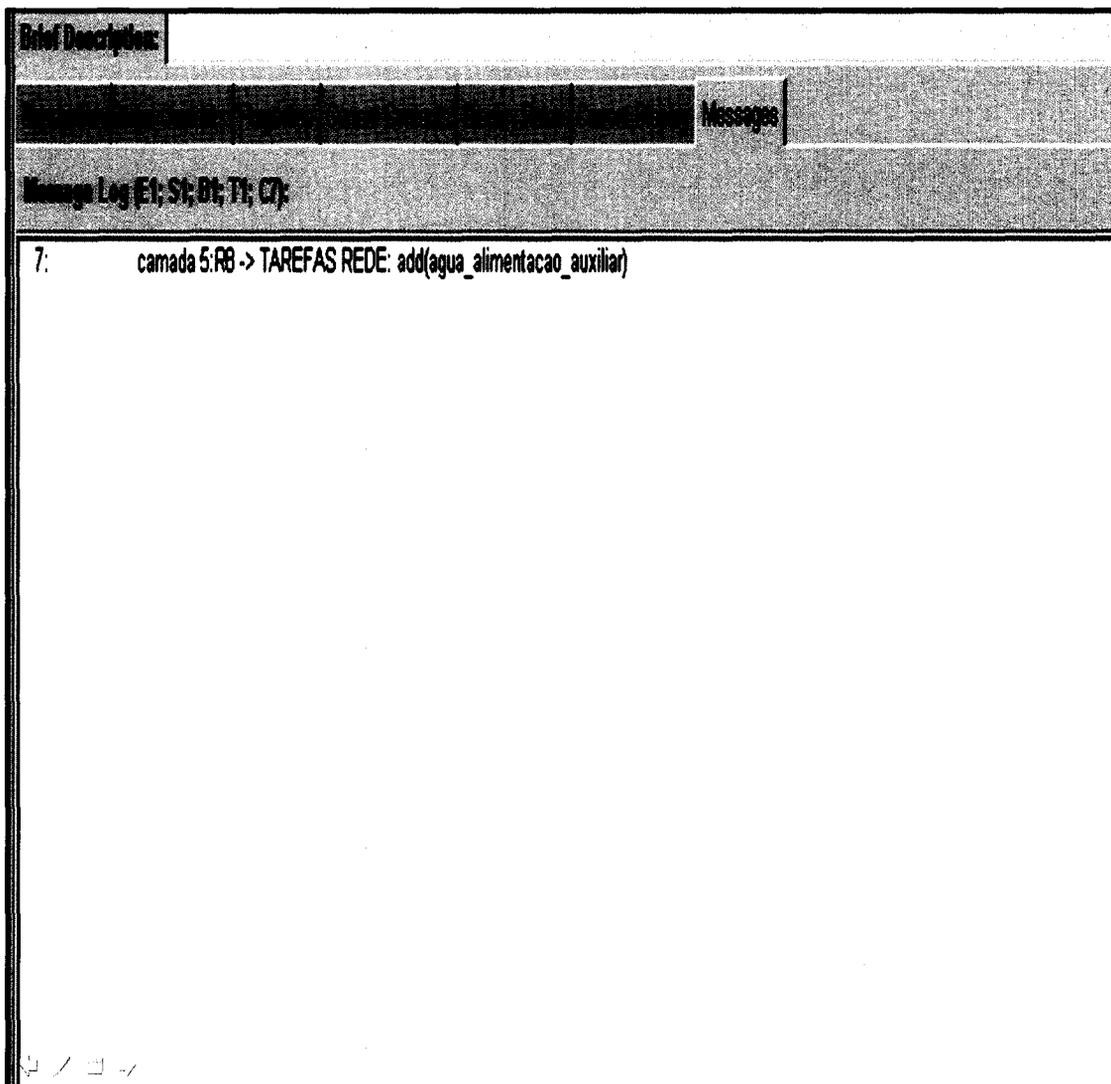


Figura 19 – Exemplo de Saída das Tarefas de Segurança

3.2.8 Acidentes Treinados pelo Sistema Neuro-nebuloso

A rede foi treinada com os dados obtidos no simulador gráfico do Centro de Treinamento da ELETRONUCLEAR para os seguintes acidentes selecionados:

1. Pequena Perda de Refrigerante do Reator Simultaneamente à Ruptura de Tubo de Gerador de Vapor

Neste cenário, ocorre a ruptura de um tubo do Gerador de Vapor, causando despressurização do Sistema de Refrigeração do Reator e aumento do nível de radioatividade no secundário. É acionado o Sistema de Injeção de Segurança de Alta Pressão que mantém o núcleo refrigerado até o isolamento do Gerador de Vapor falho e o controle do acidente. São ativadas as tarefas de segurança Injeção de Segurança de Alta Pressão, Última Fonte Fria, Sistema de Ventilação da Sala de Controle em Modo de Emergência, Sistema de Ventilação da Área Controlada em Modo de Emergência, Isolamento do Gerador de Vapor, Resfriamento e Despressurização Rápidos do Sistema de Refrigeração do Reator.

2. Perda de água de Alimentação Principal com Água de Alimentação Auxiliar

Este cenário apresenta um transitório de perda de Água de Alimentação Principal, devido ao desarme das bombas principais, causando o desarme automático do Reator por baixo nível nos Geradores de Vapor. Neste caso, está disponível o Sistema de Água de Alimentação Auxiliar que provê o resfriamento do refrigerante do reator, mantendo a temperatura do Sistema de Refrigeração do Reator sob controle. É ativada a Tarefa Água de Alimentação Auxiliar.

3. Ruptura de Tubo de Gerador de Vapor a 80%

Neste acidente, foi simulada uma ruptura equivalente ao diâmetro de uma linha de instrumentação do Sistema de Refrigeração do Reator. Neste caso, o Sistema de Injeção de Segurança de Alta Pressão é acionado e é capaz de manter o inventário de refrigerante para garantir o resfriamento do núcleo até a colocação do Sistema de Remoção de Calor Residual em operação, para manutenção do resfriamento em longo prazo. Neste cenário, foram ativadas as tarefas de segurança Injeção de Segurança de Alta Pressão, Última Fonte Fria, Sistema de Ventilação da Sala de Controle em Modo de Emergência, Sistema de Ventilação da Área Controlada em Modo de Emergência e Água de Alimentação Auxiliar.

4. Falha dos Dispositivos de Desarme do Reator a 50%

Neste cenário ocorre a falha do desarme do reator após um transitório em que este é requerido. Ocorre um aumento de pressão do Sistema de Refrigeração do Reator, bem como da Temperatura Média. O transitório é controlado pela injeção de ácido bórico, causando o desligamento do reator. É ativada a tarefa Injeção de Ácido Bórico, Água de Alimentação Auxiliar, Injeção de segurança de Alta Pressão, Última Fonte Fria, Sistema de Ventilação da Área Controlada em Modo de Emergência, Sistema de Ventilação da Sala de Controle em Modo de Emergência e Desligamento das Bombas de Refrigeração do Reator.

5. Perda de Água de Alimentação Auxiliar com falha de um trem do Sistema de Água de Alimentação Auxiliar

Neste cenário ocorre a perda do sistema de Água de Alimentação Principal com deficiência de resfriamento do refrigerante do reator pelo Sistema de Água de Alimentação Auxiliar. Ocorre o aumento de pressão e temperatura média do refrigerante do reator. São ativadas as tarefas Água de Alimentação Auxiliar e Desligamento das Bombas de Refrigeração do Reator.

6. Grande Ruptura de Linha de Vapor e de Tubo do Gerador de Vapor

Este cenário simula uma grande ruptura de uma linha de vapor coincidente com uma ruptura de um tubo do Gerador de Vapor, causando a despressurização do Sistema de Refrigeração do Reator e do Gerador de Vapor 1 e o vazamento de grande quantidade de refrigerante para o Sistema de Vapor Principal, com o consequente aumento da atividade nas linhas de vapor. São ativadas as tarefas de segurança Injeção de Segurança de Alta Pressão, Última Fonte Fria, Sistema de Ventilação da Sala de Controle em Modo de Emergência, Sistema de Ventilação da Área Controlada em Modo de Emergência, Isolamento do Gerador de Vapor, Resfriamento e Despressurização Rápidas do Sistema de Refrigeração do Reator.

7. Grande Perda de Refrigerante do Reator no Interior da Contenção

Neste acidente, foi simulada uma ruptura de grande proporção no Sistema de Refrigeração do Reator. A pressão do refrigerante é bastante reduzida para a injeção do acumulador e o nível do Tanque de Água de Carregamento atinge o valor requerido para iniciar a recirculação do poço da contenção até concluir o resfriamento do refrigerante e a despressurização. Neste cenário, são ativadas as tarefas de segurança Injeção de Segurança de Alta Pressão, Última Fonte Fria, Sistema de Ventilação da Sala de Controle em Modo de Emergência, Sistema de Ventilação da Área Controlada em Modo de Emergência, Água de Alimentação Auxiliar.

8. Pequena Perda de Refrigerante do Reator

Neste cenário ocorre uma pequena perda de refrigerante do reator com queda da pressão do Sistema de Refrigeração do Reator porém passível de ser controlada pelo Sistema de Injeção de Segurança de Alta Pressão. São ativadas as tarefas Água de Alimentação Auxiliar, Injeção de Alta Pressão, Última Fonte Fria, Sistema de Ventilação da Sala de Controle em Modo de Emergência, Sistema de Ventilação da Área Controlada em Modo de Emergência.

O acidente 8 não foi treinado pela rede. Seus dados foram utilizados somente para teste da capacidade de generalização. A propriedade de generalização da rede é sua capacidade de indicar as tarefas de segurança corretas quando submetida a dados para os quais ela não foi treinada. Isto só vale para as tarefas de segurança que foram ativadas pelos acidentes treinados. Uma tarefa de segurança para a qual a rede não foi treinada não será indicada.

Dos 19 parâmetros selecionados, 14 foram efetivamente utilizados pelas regras ativadas nestes acidentes. Os parâmetros utilizados foram: Pressão do Sistema de Refrigeração do Reator, Temperatura Média do Refrigerante, Grau de Sub-resfriamento do Refrigerante do Reator, Pressão do Envoltório de Contenção, Pressão no Gerador de Vapor 1, Pressão no Gerador de Vapor 2, Nível de Radiação no Geradores de Vapor 1, Nível de Radiação no Geradores de Vapor 2, Fluxo de Nêutrons na Faixa de Potência, Taxa de Partida da Faixa Intermediária, Nível no

Geradores de Vapor 1, Nível no Geradores de Vapor 2, Vazão de Água de Alimentação Auxiliar para o Gerador de Vapor 1 e Vazão de Água de Alimentação Auxiliar para o Gerador de Vapor 2.

As Tarefas de Segurança ativadas nestes acidentes foram: Injeção de Segurança de Alta Pressão, Última Fonte Fria, Ventilação da Área Controlada em Modo de Emergência, Ventilação da Sala de Controle em Modo de Emergência, Resfriamento Rápido do Sistema de Refrigeração do Reator, Despressurização Rápida do Sistema de Refrigeração do Reator, Água de Alimentação Auxiliar, Isolamento da Contenção, Isolamento da Ventilação da Contenção, Resfriamento da Contenção, Isolamento de Gerador de Vapor Falho e Desligamento das Bombas de Refrigeração do Reator.

Alguns parâmetros são utilizados em mais de uma Tarefa de Segurança. Por exemplo, a Pressão do Sistema de Refrigeração do Reator ativa as tarefas Injeção de Segurança de Alta Pressão, Injeção dos Acumuladores e Injeção de Baixa Pressão. Nestes casos, a distribuição dos conjuntos nebulosos para uma variável, feita durante o treinamento da rede com os dados dos instantes de tempo de um determinado acidente, será baseada somente nas tarefas ativadas por este parâmetro que são acionadas neste acidente. Deste modo, é necessário impedir que a rede treine com as tarefas relacionadas a este parâmetro que não são ativadas neste acidente. Isto é feito pela matriz w_{34} composta pelos pesos das ligações das camadas 3 e 4 da rede. Quando estiverem disponíveis dados de um acidente no qual outras tarefas dependentes desse parâmetro sejam ativadas, a rede deverá ser treinada novamente para permitir uma nova distribuição dos conjuntos nebulosos.

4. TESTES E ANÁLISE DOS RESULTADOS

No capítulo anterior, mostramos como o sistema de suporte foi construído a partir de um sistema especialista clássico e um sistema neuro-nebuloso, integrados no ambiente computacional COGENT. Neste capítulo, vamos apresentar e analisar os resultados dos testes efetuados para verificar o funcionamento do sistema.

O primeiro teste teve os seguintes objetivos: (1) verificar o funcionamento adequado do sistema neuro-nebuloso, comparando suas taxas de acertos e erros em relação às Tarefas de Segurança indicadas pelo conjunto de regras; (2) verificar a robustez do sistema, ou seja, sua capacidade de operar com dados imprecisos; e (3) verificar a capacidade de generalização do sistema, submetendo-o aos dados do acidente 8, nos quais ele não foi treinado. Os dados de treinamento do acidente 8 correspondem a cerca de 12% do total dos dados de todos os acidentes, um valor aceitável para verificação da capacidade de generalização se comparado, por exemplo, com o Programa NEUROHELL 2 [42], que tem como *default* o valor de 20% dos dados como conjunto de teste.

No segundo teste verificamos o funcionamento do sistema especialista clássico, baseado em regras com valores rígidos dos parâmetros extraídos dos dados de treinamento, juntamente com o funcionamento integrado do sistema de suporte, conforme construído no COGENT.

4.1 Testes do Sistema Neuro-nebuloso

4.1.1 Comparação dos Resultados Obtidos com Dados de Treinamento

O primeiro teste foi feito por intermédio do programa TESTEDAREDEOUT7, incluído no Apêndice 2, programado em MATLAB. Este programa analisa as linhas de dados de instantes de tempo de todos os acidentes, comparando o resultado das tarefas de segurança ativadas pela rede com as ativadas pelo conjunto de regras. Como este teste é feito com os dados de treinamento do sistema as tarefas ativadas pela base de regras deverá ser igual às ativadas pelo sistema neuro-nebuloso.

A Tabela 7 mostra o índice de acertos e erros das tarefas ativadas pela rede em comparação com as ativadas pelo conjunto de regras.

Tabela 7 – Taxas de Acertos da Rede em Relação ao Conjunto de Regras

Acidente	Taxa de Indicação Correta de Tarefas (%)	Taxa de Indicação Errônea de Tarefas (%)
1	90,7	1,3
2	100	0
3	100	3
4	62	2
5	100	0
6	90	9,1
7	96	3
8	100	3

Nesta tabela, a coluna Taxa de Indicação Correta de Tarefas apresenta a porcentagem de linhas de dados do acidente da coluna 1 para as quais as tarefas ativadas pela rede estão de acordo com as ativadas pelo conjunto de regras. Cada linha de dados de instantes de tempo fornecida pelo simulador gráfico corresponde ao valor dos parâmetros a cada 0,05 segundos durante o acidente. Na coluna Taxa de Indicação Errônea de Tarefas, é mostrada a porcentagem de linhas de dados do acidente da coluna 1 em que a rede indicou uma tarefa em desacordo com o conjunto de regras. Note-se que estes não são complementares, cada um corresponde a um tipo de desempenho. Os acidentes da coluna 1 estão de acordo com a Tabela 4.

Os resultados apresentados nessa tabela mostram que a rede tem boa confiabilidade na ativação de tarefas de acordo com o conjunto de regras, exceto pela baixa taxa de indicação correta de tarefas no acidente 4.

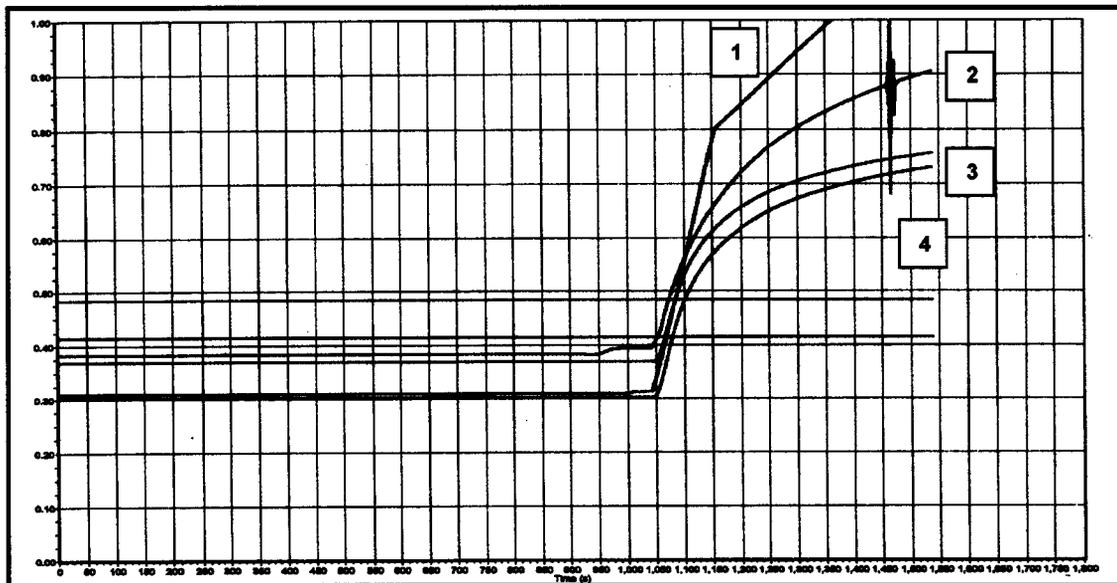
O acidente 4 trata de uma falha do desarme do reator a 50%. Neste acidente, a rede teve dificuldade em identificar o momento do desarme do reator, o que causou a demora em ativar a tarefa 5 – Injeção de Ácido Bórico. Isto se deve a não termos nas

linhas de dados uma variável que indique o momento do desarme e termos sido obrigados a usar a potência elétrica como sinal, o que causou um retardo.

Embora não tenhamos um critério estabelecido a priori, podemos dizer que o desempenho do sistema em relação à não indicação de uma tarefa errada, mostrada na coluna 3, apresentou resultados satisfatórios, exceto pela alta taxa de indicação errônea de tarefas para o acidente 6. Neste acidente houve um retardo na ativação da tarefa 1 e a ativação errônea da tarefa 21.

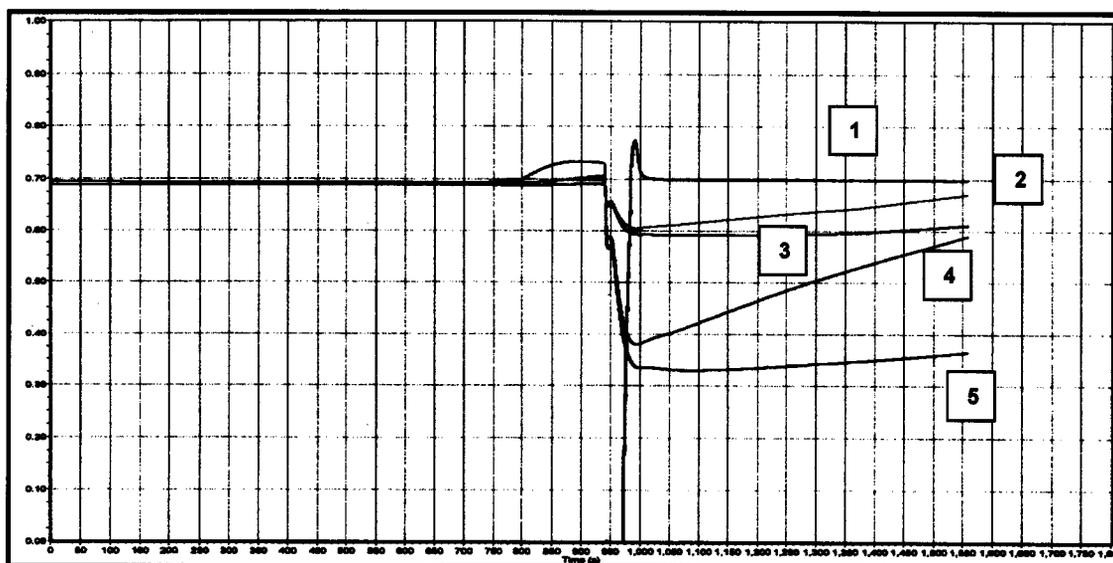
4.1.2 Análise dos Resultados Detalhados para o Acidente 1

As Figuras 20 e 21 mostram exemplos do comportamento de alguns parâmetros no acidente 1.



1. Nível no Poço da Contenção
2. Pressão na Contenção
3. Radiação no Nível da Mesa de Selagem
4. Radiação na Contenção

Figura 20 – Comportamento dos Parâmetros da Contenção no Acidente 1



1. Vazão de Água de Alimentação Auxlliar – Gerador de Vapor 2
2. Nível Faixa Larga – Gerador de Vapor 2
3. Nível Faixa Larga – Gerador de Vapor 1
4. Nível Faixa Estreita – Gerador de Vapor 2
5. Nível Faixa Estreita – Gerador de Vapor 1

Figura 21 – Comportamento dos Parâmetros do Secundário no Acidente 1

A Tabela 8, na próxima página, mostra os resultados obtidos para o acidente 1, com a utilização de dados de treinamento. Como este teste é feito com os dados de treinamento do sistema, as tarefas ativadas pela base de regras deverão ser iguais às ativadas pelo sistema neuro-nebuloso. Nesta tabela, a coluna TCR mostra o número de linhas de dados de instantes do acidente em que o conjunto de regras ativou a tarefa indicada na primeira coluna. A coluna TCA mostra o número de linhas de dados de instantes de tempo do acidente em que a rede ativa uma Tarefa de Segurança de acordo com o conjunto de regras. A coluna TNI mostra o número de linhas de dados de instantes de tempo do acidente em que a rede deixa de ativar uma tarefa que foi ativada pelo conjunto de regras. A coluna TIA mostra o número de linhas de dados de instantes de tempo do acidente em que a rede ativa uma tarefa que não foi ativada pelo conjunto de regras.

Tabela 8 – Tarefas Ativadas no Acidente 1

Tarefa	TCR	TCA	TNI	TIA
1	9026	8790	236	0
2	9026	8790	236	0
3	9026	8790	236	0
4	9026	8790	236	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	9026	9026	0	0
10	9026	9026	0	0
11	9026	9026	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	6624	3084	3540	0
21	7068	7068	0	1958

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

A Figura 22 apresenta a saída do nodo da Camada 4 referente à Tarefa 20. Verifica-se que a esta tarefa é ativada pelo sistema neuro-nebuloso somente em parte do intervalo em que a mesma é ativada pelo conjunto de regras.

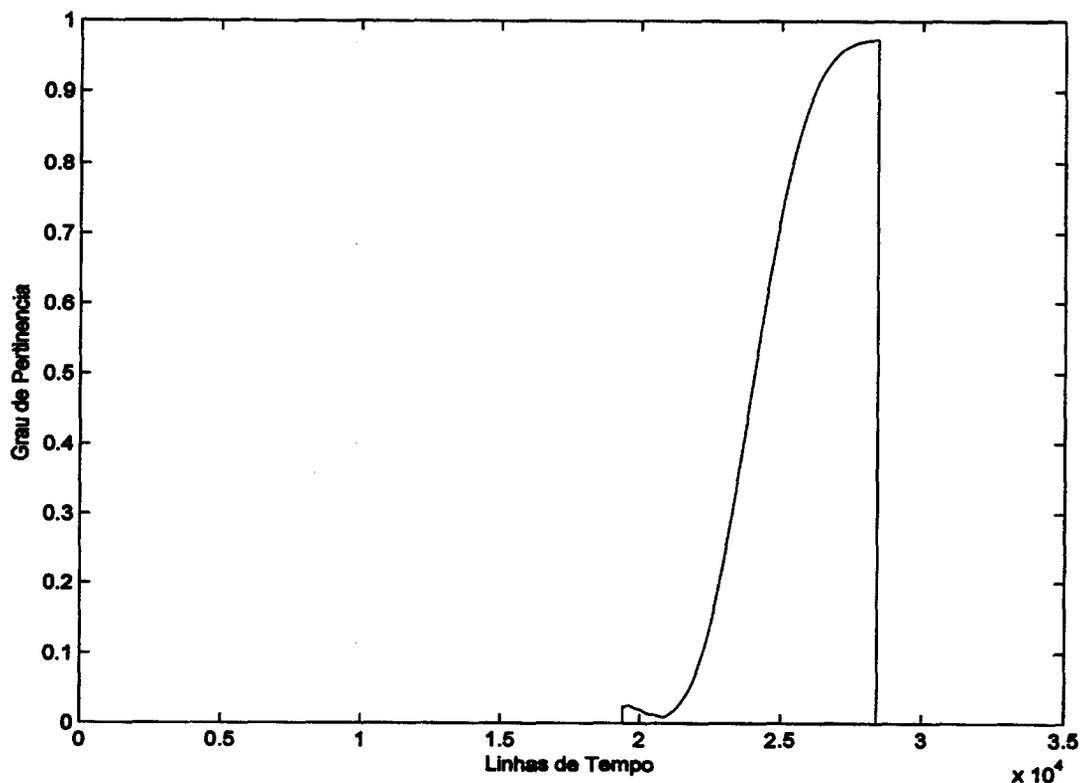


Figura 22 – Saída do Nodo da Camada 4 Referente à Tarefa 20

Nesta figura, vemos que houve um atraso na indicação da Tarefa 20, o que ocorreu somente após a linha de dados 25.000. Neste caso, para que a tarefa seja ativada antes dessa linha, será necessária a redução do valor de corte para a saída da camada 5, especificamente para esta tarefa, para um valor menor que o de 0,79 adotado para as outras tarefas.

As Figuras 23, 24, e 25 mostram as saídas dos nodos referentes às demais tarefas ativadas neste acidente.

As tabelas com as tarefas ativadas nos acidentes 2 a 8 estão no Apêndice 3. As avaliações de seus resultados não diferem da feita para o acidente 1, indicando resultados compatíveis.

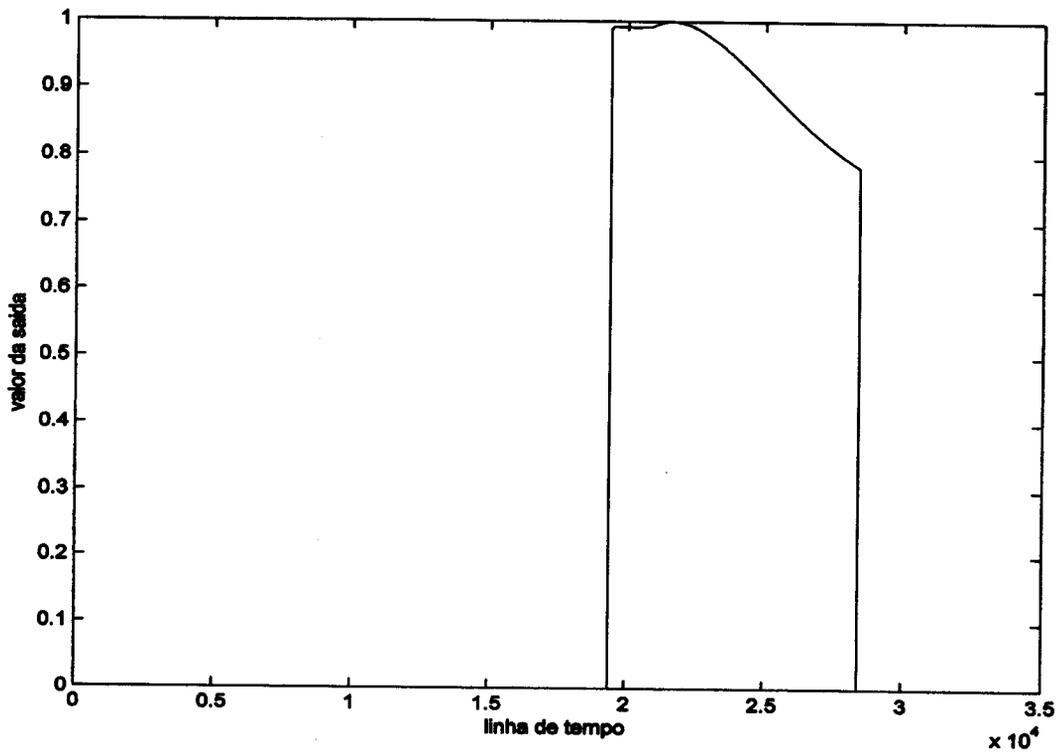


Figura 23 – Saída do Nodo Referente às Tarefas 1,2,3 e 4

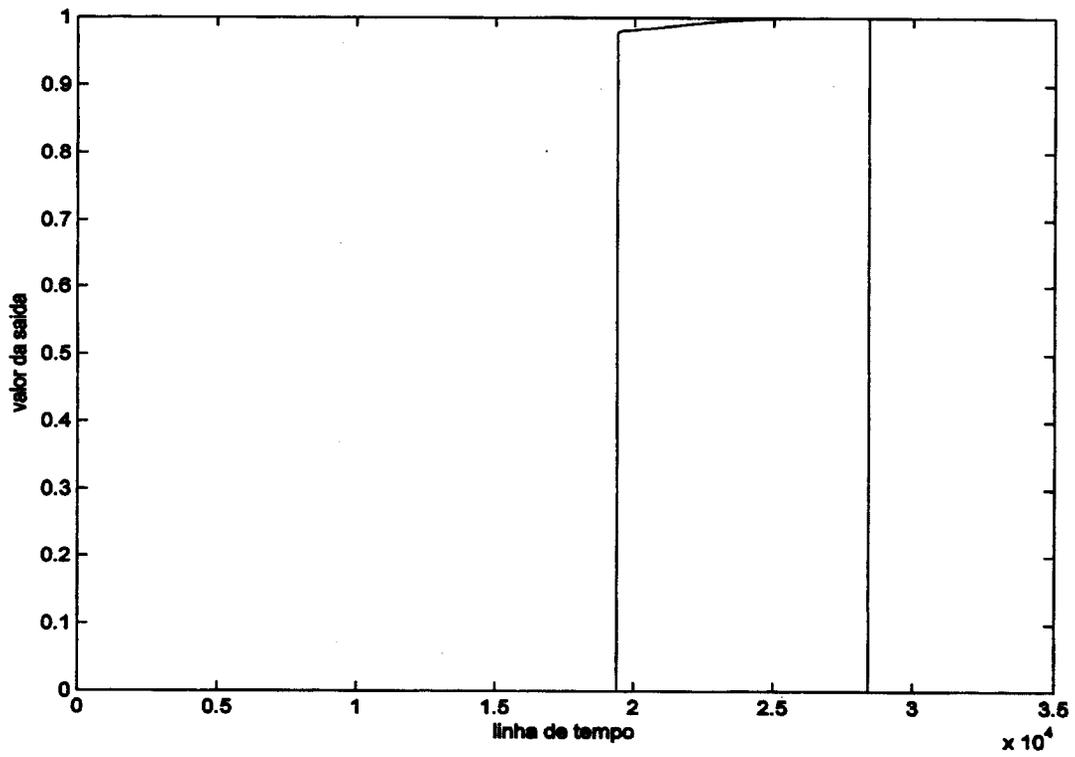


Figura 24 – Saída do Nodo Referente à Tarefa 11

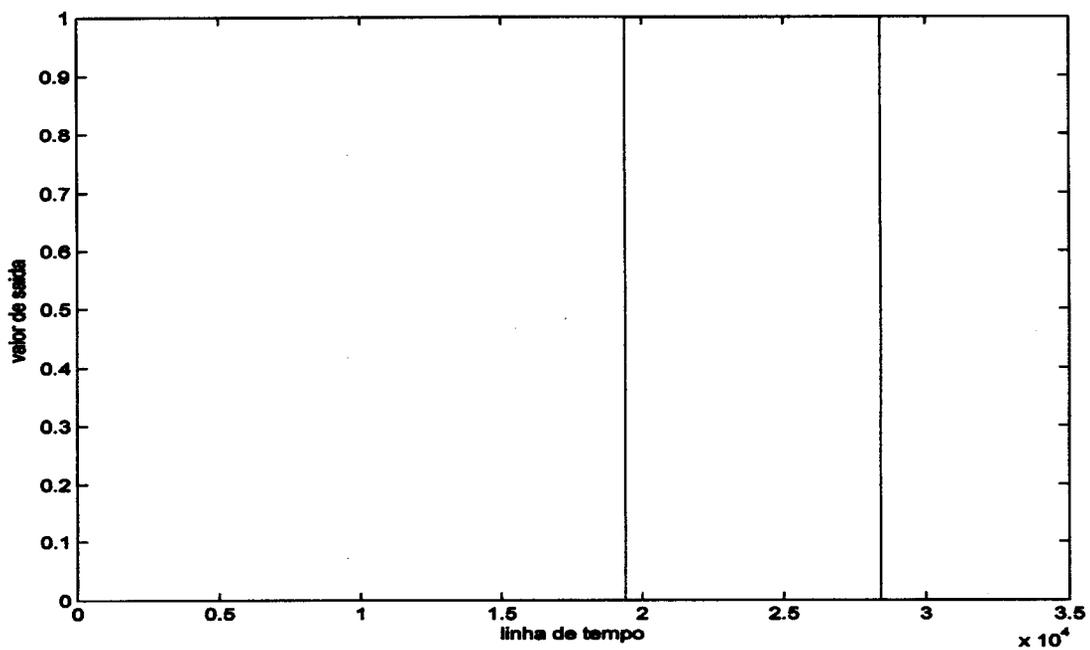


Figura 25 - Saída do Nodo Referente às Tarefas 8 e 9

A Tarefa 21 é ativada um número maior de vezes pela rede que pelo conjunto de regras. Isto também poderia ser corrigido ajustando-se o valor de corte da camada 5. A saída do nodo desta tarefa é mostrada na Figura 26.

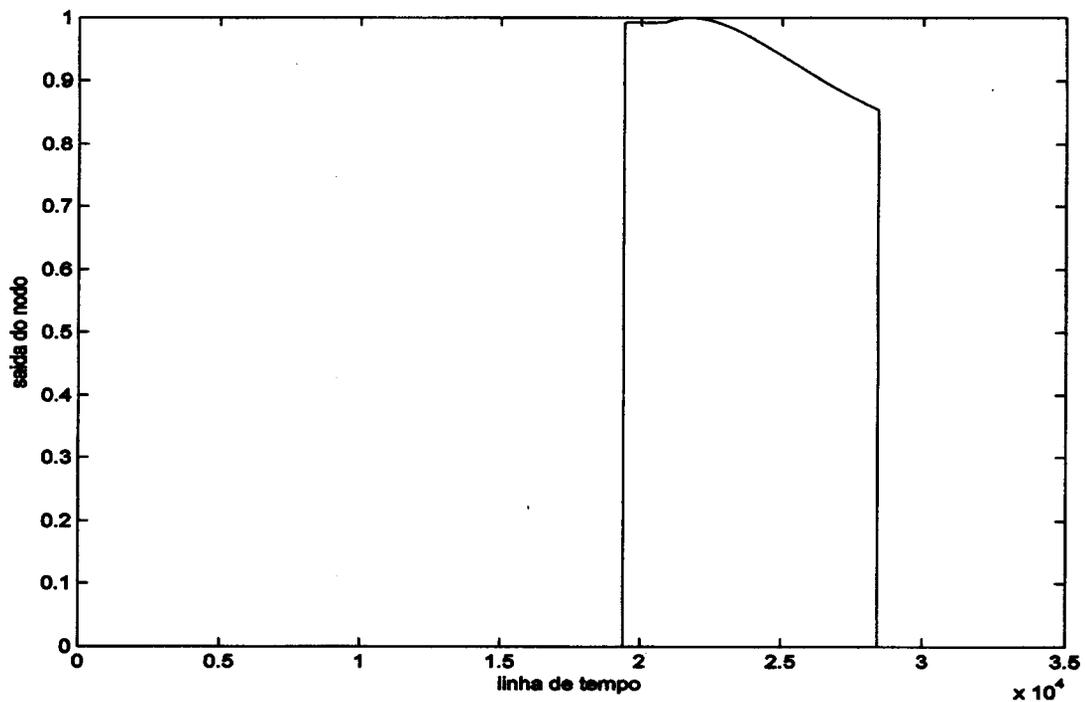


Figura 26 – Saída do Nodo Referente à Tarefa 21

4.1.3 Análise dos Resultados dos Testes de Robustez do Sistema

A seguir, foram feitos dois testes com o mesmo programa TESTEDAREDEOUT7, mantendo-se fixos os valores das duas variáveis mais utilizadas na ativação das tarefas. Estes testes foram feitos com a intenção de verificar a capacidade do sistema de ativar as tarefas corretas, estando uma variável com um valor fixo, simulando uma condição em que um indicador do painel de controle esteja inoperável.

No primeiro teste, manteve-se o valor da variável Pressão do Sistema de Refrigeração do Reator fixo em 109 kgf/cm². Este valor foi escolhido por ser o valor médio entre os pontos de atuação das tarefas 1 (123 kgf/cm²) e 20 (95 kgf/cm²) que são ativadas pelo mesmo parâmetro, sendo então um ponto crítico para o sistema. Os resultados em relação às taxas de acertos são apresentados na Tabela 9.

Tabela 9 – Taxas de Acertos da Rede em Relação ao Conjunto de Regras para um Desvio na Pressão do Sistema de Refrigeração do Reator

Acidente	Taxa de Indicação Correta de Tarefas (%)	Taxa de Indicação Errônea de Tarefas (%)
1	89,5	6
2	100	10,5
3	100	5,5
4	72,8	10,1
5	100	10,1
6	90,2	9,1
7	97	13
8	100	11,4

Na Tabela 10 são apresentados os resultados obtidos para o acidente 1, mantendo-se o valor da Pressão do Sistema de Refrigeração do Reator fixo em 109 kgf/cm². Pelos resultados apresentados, verificamos que não houve diferença entre as ativações de Tarefas de Segurança na condição do teste e as ativações ocorridas com os valores das linhas de dados de treinamento apresentados na Tabela 6. As tabelas

referentes aos resultados dos demais acidentes para estes testes estão apresentados no Apêndice 3.

Tabela 10 – Tarefas Ativadas no Acidente 1 com o Valor da Pressão do Sistema de Refrigeração do Reator Fixo em 109 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	9026	9026	0	0
2	9026	9026	0	0
3	9026	9026	0	0
4	9026	9026	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	9026	9026	0	0
10	9026	9026	0	0
11	9026	9026	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	6624	3048	3540	0
21	0	0	0	9026

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

No segundo teste, foi mantido o valor da variável Pressão no Gerador de Vapor fixo em um valor de 22 kgf/cm². Este valor foi escolhido por ser o valor médio entre os pontos de ativação das tarefas 1 (42 kgf/cm²) e 19 (2 kgf/cm²) ativadas pelo mesmo parâmetro, sendo então um ponto crítico para o sistema. Os resultados em relação às taxas de acertos são apresentados na Tabela 11.

Tabela 11 – Taxas de Acertos da Rede em Relação ao Conjunto de Regras

Acidente	Taxa de Indicação Correta de Tarefas (%)	Taxa de Indicação Errônea de Tarefas (%)
1	92,4	7,9
2	100	10,5
3	100	9,2
4	74,2	12,7
5	100	11,1
6	94,6	14,1
7	95,5	14,5
8	100	15,8

Na Tabela 12 são apresentados os resultados obtidos para o acidente 1. Pelos resultados apresentados verificamos que não houve diferença entre as ativações de Tarefas de Segurança na condição do teste e as ativações ocorridas com os valores das linhas de tempo dos dados de treinamento, apresentados na Tabela 6. Neste caso a tarefa 1 foi ativada também pela Pressão do Gerador de Vapor, além da Pressão do Sistema de Refrigeração do Reator. As tabelas referentes aos resultados dos demais acidentes para estes testes estão apresentados no Apêndice 3.

Tabela 12 – Tarefas Ativadas no Acidente 1 com o Valor da Pressão no Gerador de Vapor Fixo em 22 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	9026	8790	236	0
2	9026	8790	236	0
3	9026	8790	236	0
4	9026	8790	236	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	9026	9026	0	0
10	9026	9026	0	0
11	9026	9026	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	6624	3088	3540	0
21	7068	7068	0	1958

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

4.1.4 Análise dos Resultados do Teste de Capacidade de Generalização

Os dados do acidente 8 não foram utilizados no treinamento da rede, para serem utilizados na verificação da capacidade de generalização desta. A Tabela 13 apresenta os resultados obtidos pela rede e pela base de regras com a utilização desses dados. Pode-se verificar que as tarefas ativadas pela base de regras, apresentados na coluna TCR, estão de acordo com as ativadas pela rede, apresentadas na coluna TCA. Isto mostra que, neste caso, a rede funcionou adequadamente com dados com os quais ela não foi treinada. Observe-se que as tarefas ativadas no acidente 8 foram treinadas nos outros acidentes. Não se espera que a rede venha a ativar uma tarefa para a qual ela não foi treinada.

Tabela 13 – Tarefas Ativadas no Acidente 8

Tarefa	TCR	TCA	TNI	TIA
1	9586	9586	0	0
2	9586	9586	0	0
3	9586	9586	0	0
4	9586	9586	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	9586	9586	0	0
12	7557	7557	0	672
13	7557	7557	0	672
14	7557	7557	0	672
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	4003	4003	0	5583

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

4.2 Testes do Sistema Especialista Clássico e do Sistema Integrado

Para o segundo teste, com o objetivo de verificar o funcionamento do sistema controlado por regras e o funcionamento integrado do sistema, conforme construído no COGENT, foram selecionadas três linhas de tempo, correspondentes a três instantes durante o acidente, de quatro dos acidentes de treinamento. A primeira linha no início, logo após o desarme do reator, a segunda no meio e a terceira no fim de cada acidente. Estes acidentes foram escolhidos pela variedade das Tarefas de Segurança ativadas. Os valores selecionados estão nas Tabelas 14, 15, 16, 17 e 18.

Nestas tabelas a primeira coluna corresponde aos parâmetros conforme Tabela 1. As demais colunas correspondem aos valores desses parâmetros nas linhas de tempo dos acidentes indicadas.

Tabela 14 – Dados para Teste do Acidente 1

Parâmetro	950 (segundos)	1000 (segundos)	1100 (segundos)
P2	113,7	98	90
P3	296	288	285
C4	0,47	0,46	0.8
C2	0,08	0,08	0.02
G4	69	71	68
G5	69	71	68
G2	56,7	38	33
G3	58,5	38	42
C5	96	95	94
G6	909	10	3
G7	903	16	9
N2	0	0	0
N4	0	0	0
N6	0	0	0
C6	0	86	84
C7	0	86	84
P5	28	21	35
R3	0	0	0
R4	4,94	4,94	4,94

Tabela 15 – Dados para Teste do Acidente 3

Parâmetro	950 (segundos)	1152 (segundos)	1374 (segundos)
P2	112,6	96	92
P3	295,6	283	277
C4	0,46	0,46	0,46
C2	0,02	0,02	0,02
G4	69	66	58
G5	69	66	58
G2	56,4	33	34,8
G3	58,2	45	58
C5	96	95	94
G6	854	8	3
G7	827	13	7
N2	0	0	0
N4	0	0	0
N6	2,4	0	0
C6	0,07	83	83
C7	0,05	83	83
P5	27,5	25	27
R3	0	0	0
R4	4,95	4,95	4,95

Tabela 16 – Dados para Teste do Acidente 5

Parâmetro	450 (segundos)	810 (segundos)	1028 (segundos)
P2	146	149	152
P3	290	285	282
C4	0,34	0,35	0,35
C2	0,04	0,04	0,04
G4	74	68	65
G5	74	68	65
G2	0	0	0
G3	0	0	0
C5	95	96	96
G6	31	26	24
G7	32	26	25
N2	0	0	0
N4	0	0	0
N6	0,3	0	0
C6	27	29	29
C7	27	29	29
P5	48	55	69
R3	0	0	0
R4	0	0	0

Tabela 17 – Dados para Teste do Acidente 7

Parâmetro	350 (segundos)	800 (segundos)	1300 (segundos)
P2	120	122	148
P3	277	277	277
C4	0,9	1,64	1,85
C2	0,2	0,2	0,2
G4	0	0	0
G5	27	41	59
G2	2,3	0,8	0,36
G3	56	37	31,6
C5	94	88	83
G6	148	25	32
G7	0	0	0
N2	0	0	0
N4	0	0	0
N6	0,06	0	0
C6	84	84	84
C7	84	84	84
P5	31	80	100
R3	0	0	0
R4	0	0	0

Os resultados, mostrados na Tabela 18, demonstram que as tarefas de segurança indicadas pelo sistema especialista simbólico estão coerentes com as indicadas pelo sistema especialista conexionista. Estes dados foram selecionados dos dados dos acidentes utilizados para treinamento da rede neural e, portanto, as saída devem ser iguais.

Tabela 18 – Resultados Obtidos com os Dados de Teste

Acidente/tempo (seg)	Tarefas de segurança Sistema Espec. Simbólico	Tarefas de Segurança Sist. Espec. Neuro-nebuloso
1/950	1, 2, 3, 4, 9, 10, 11, 20, 21	1, 2, 3, 4, 9, 10, 11, 20, 21
1/1000	1, 2, 3, 4, 9, 10, 11, 20, 21	1, 2, 3, 4, 9, 10, 11, 20, 21
1/1100	1, 2, 3, 4, 9, 10, 11, 20, 21	1, 2, 3, 4, 9, 10, 11, 20, 21
3/950	1, 2, 3, 4, 9, 10, 11, 21	1, 2, 3, 4, 9, 10, 11, 21
3/1152	1, 2, 3, 4, 9, 10, 11, 21	1, 2, 3, 4, 9, 10, 11, 21
3/1354	1, 2, 3, 4, 9, 10, 11, 21	1, 2, 3, 4, 9, 10, 11, 21
5/450	11, 20	11, 20
5/810	11, 20	11, 20
5/1028	11, 20	11, 20
7/350	1, 2, 3, 4, 11, 12, 13, 14, 19	1, 2, 3, 4, 11, 12, 13, 14, 19
7/800	1, 2, 3, 4, 11, 12, 13, 14, 19	1, 2, 3, 4, 11, 12, 13, 14, 19
7/1300	1, 2, 3, 4, 11, 12, 13, 14, 19	1, 2, 3, 4, 11, 12, 13, 14, 19

A Figura 27 mostra um exemplo de vetor de entrada de dados para o sistema. Os valores numéricos correspondem ao valor das variáveis de entrada no sistema em um determinado instante de tempo do acidente. Esses valores são associados às variáveis das condições das regras das tarefas de segurança na MEMÓRIA PERCEPTUAL.

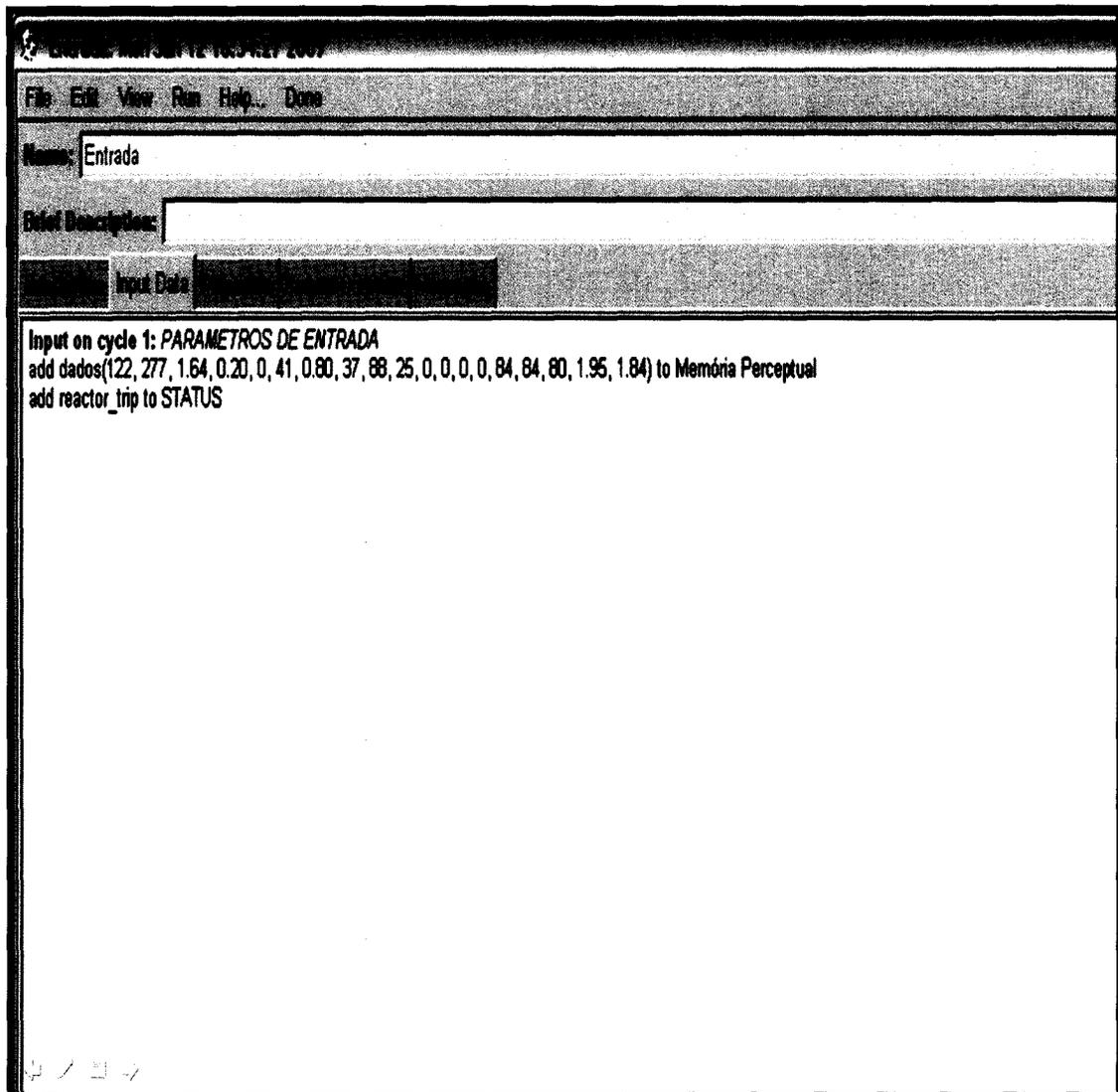


Figura 27 – Exemplo de um Vetor de Entrada

A Figura 28 mostra um exemplo de saída do módulo do sistema especialista clássico controlado por regras. As tarefas de segurança relacionadas são resultantes da ativação das regras pelos valores de entrada dos parâmetros no processador BASE DE REGRAS. Essa ativação é feita com a coincidência das condições das regras com os valores de entrada dos parâmetros.

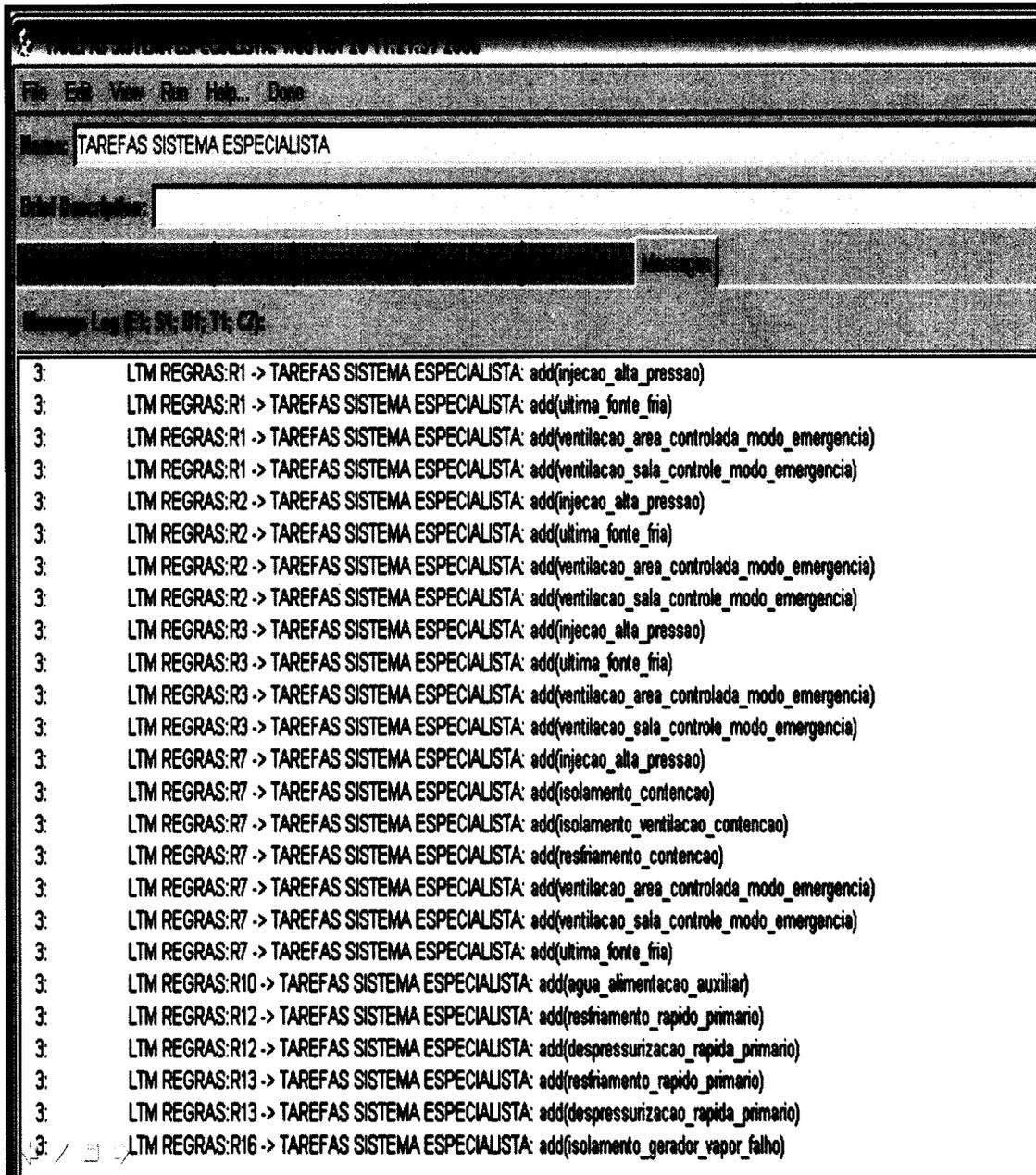


Figura 28 – Exemplo de uma Saída do Sistema Especialista Simbólico

A Figura 29 mostra um exemplo de saída do módulo do sistema especialista híbrido. As tarefas de segurança relacionadas são resultado final do processamento pelo sistema neuro-nebuloso dos valores de entrada dos parâmetros.

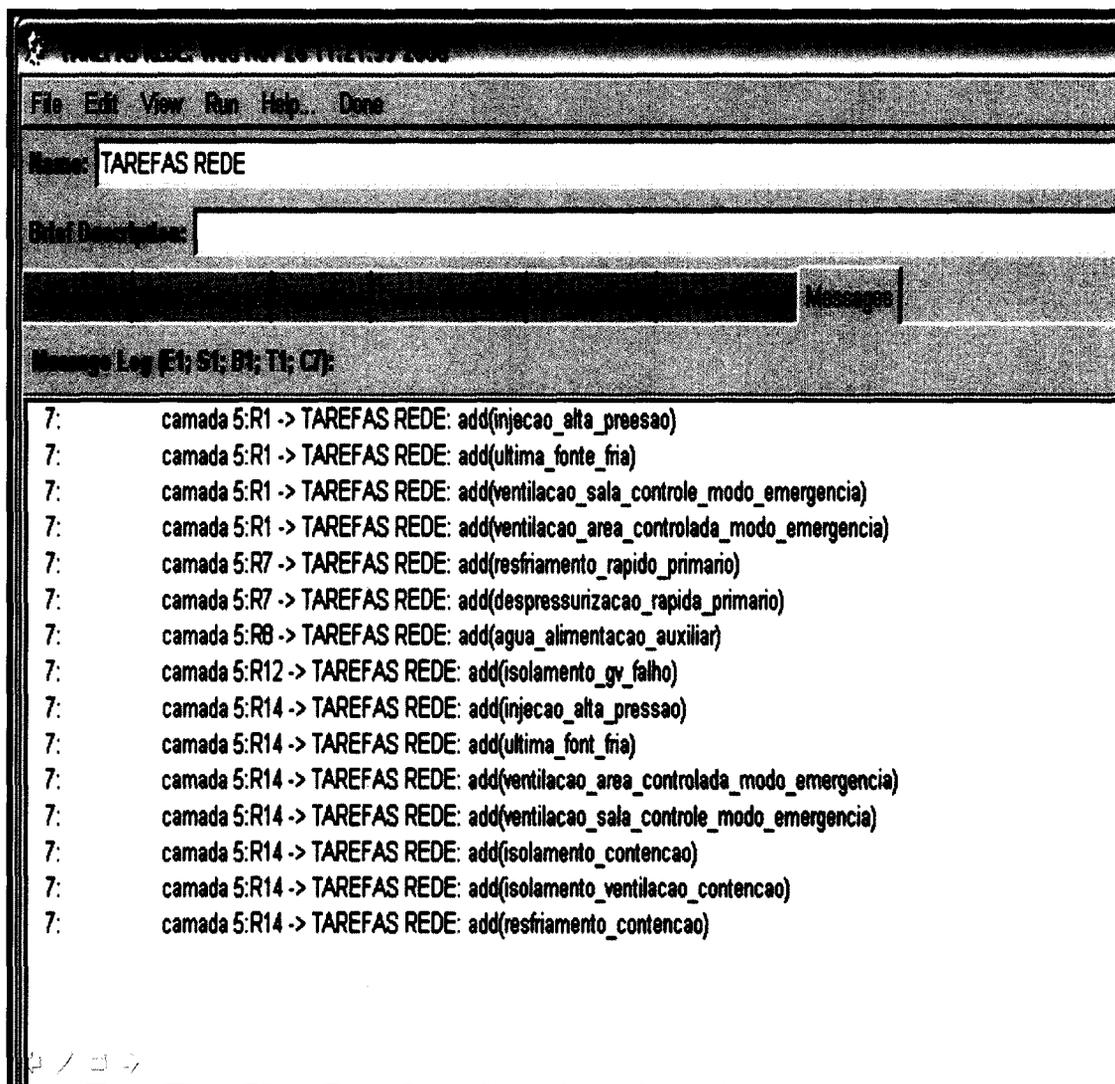


Figura 29 – Exemplo de uma Saída do Sistema Especialista Conexionista

5. CONCLUSÕES E RECOMENDAÇÕES

Esta pesquisa teve os seguintes objetivos:

1. Aplicação da simulação da cognição como um mecanismo principal no desenvolvimento de ferramentas de suporte à tomada de decisões, provendo um arcabouço, na forma de um modelo cognitivo simplificado, para a construção de um sistema de suporte. Esta abordagem tem o objetivo de facilitar a comunicação do usuário com o sistema na medida em que este tenta simular a cognição do usuário, por exemplo, apresentando na interface do sistema não somente a tarefa ativada, mas, também, o que gerou esta conclusão, ou seja, o conteúdo da Memória de Trabalho.
2. A proposição de um conceito de sistema de suporte ao operador de uma usina nuclear, que procurasse considerar os contextos familiares e não familiares possíveis de ocorrer durante o combate a um transitório operacional. Esses contextos definem o modo de tomada de decisão do operador como: controle por regras, também chamado conhecimento superficial, declarativo ou procedimental; ou controle baseado no conhecimento, também chamado de conhecimento profundo, respectivamente.
3. Considerar uma abordagem diferente daquela convencional, baseada no diagnóstico do acidente e em procedimentos contendo instruções sequenciais, para as ações do operador no combate ao transitório. Foi aqui adotado o conceito de Tarefas de Segurança, no qual as instruções sequenciais dos Procedimentos de Operação em Emergência foram agrupadas segundo seus objetivos, em subconjuntos que receberam esse nome. Um conjunto de tarefas de segurança forma, então, uma estratégia para combate ao transitório operacional. Este tratamento vai ao encontro do objetivo primordial da manutenção da consciência da situação, por parte do operador, durante os vários passos para o combate ao transitório.

4. A construção de um protótipo do sistema conceitual proposto, empregando-se ferramentas da inteligência artificial que teoricamente possuem características capazes de preencher as necessidades do sistema, especificamente a lógica nebulosa e as redes neurais artificiais.

O sistema foi desenvolvido na forma de um modelo cognitivo simplificado, em um ambiente específico para este fim, denominado COGENT. O sistema consiste de processadores e memórias de armazenamento que representam áreas funcionais, tais como Processador Perceptual, Memória de Longa Duração e Memória de Trabalho. A Memória de Longa Duração foi construída em dois módulos, sendo o primeiro para os contextos familiares e o segundo para os contextos não familiares à equipe de operação da Sala de Controle da usina.

Este modelo cognitivo permite que se realizem experimentos com o sistema, atribuindo-se propriedades às memórias e processadores, tais como uma limitação na capacidade de armazenamento de informação ou estabelecendo-se uma constante de decaimento para a informação armazenada no componente. Isto permite modelar características do operador no seu funcionamento.

A tarefa de tomada de decisão requerida para selecionar as ações de controle apropriadas, em adição às direções dos procedimentos formais, também envolve atividades cognitivas, comumente referidas como julgamento e raciocínio [43]. A necessidade de habilidades cognitivas é especialmente clara em situações onde procedimentos formais possam não existir ou possam não ser tão detalhados. Um exemplo de uma dessas situações é o gerenciamento de acidentes severos que envolvam o superaquecimento do núcleo do reator. Nestes casos, onde são disponibilizados apenas guias genéricos com as ações requeridas, o tratamento através de tarefas de segurança oferece vantagens.

A Tabela 7 mostra que os resultados obtidos, para os dados dos acidentes para os quais o sistema foi treinado, apresentaram uma taxa de acerto em relação ao conjunto de regras próxima de 100% e uma taxa de indicação errônea de tarefas próxima de zero. Verifica-se que, conforme explicado nas análises do capítulo anterior, um ajuste no valor da função de ativação da camada 5, individualmente para cada tarefa de segurança, levará os valores desta tabela para os valores ideais.

Nas Tabelas 9 e 11, que apresentam os resultados obtidos ao simular-se a indisponibilidade de dois parâmetros importantes, verificamos que as taxas de acertos em relação ao conjunto de regras mantiveram-se altas, em conformidade com os resultados apresentados na Tabela 7. As taxas de indicações errôneas de tarefas aumentaram em relação aos resultados da Tabela 7, indicando que a capacidade do sistema de obter resultados satisfatórios de dados com ruído não está ideal.

A Tabela 18 mostra os resultados obtidos nos testes do protótipo construído no COGENT, com linhas de dados de instantes de 4 acidentes escolhidos pela variedade de tarefas de segurança ativadas. Essas linhas de dados foram escolhidas aleatoriamente de instantes no início, meio e fim dos acidentes. Esses resultados demonstram que este conceito de sistema de suporte é viável e atende aos objetivos buscados.

Os resultados globais obtidos nos diversos testes aos quais o sistema foi submetido nos permitem concluir que a rede neuro-nebulosa é uma boa ferramenta para o desenvolvimento de sistemas especialistas híbridos. O grande problema para sua utilização está na introdução dos conhecimentos do especialista no domínio, na sua topologia, sem o que sua utilização é bastante limitada, se não impossível. Esta tarefa é trabalhosa e consome a maior parte do tempo necessário para o desenvolvimento do sistema. Entretanto, no decorrer do desenvolvimento do sistema, percebeu-se que à medida que se aumenta a interferência do conhecimento do especialista na rede, menor sua capacidade de obter resultados satisfatórios de dados com ruído.

No nosso caso, a interferência na topologia da rede que mais prejudicou a capacidade do sistema de obter resultados satisfatórios de dados com ruído, foi a necessidade de se limitar as conexões entre as camadas 2 e 3. Foi necessário fazer com que cada nodo de regra da camada 3 fosse conectado somente aos nodos da camada 2 referentes aos parâmetros que participam diretamente da regra. Isto foi necessário porque os parâmetros que não participam diretamente da regra se comportam diferentemente para cada tipo de acidente, dificultando tremendamente a que a rede produza suas próprias regras durante o treinamento supervisionado.

A Tabela 13 apresenta os resultados do teste do sistema híbrido neuro-nebuloso submetido às linhas de dados de instantes do acidente 8, nos quais este não foi treinado. Embora tenha sido utilizado apenas um acidente para esse teste, os resultados obtidos indicam que o sistema tem capacidade de generalização.

Para a continuação desta pesquisa seria de interesse estudar a rede neuro-nebulosa utilizada, com a intenção de melhorar sua capacidade de obter resultados satisfatórios de dados com ruído, fazendo com que a interferência do especialista no domínio fosse menos requerida. Isto poderia ser aplicado à interconexão das camadas 2 e 3, bem como das camadas 3 e 4.

O simulador gráfico que forneceu os dados de treinamento para o módulo conezionista é muito utilizado no treinamento dos operadores da usina Angra 1. Por esse motivo, o tempo disponibilizado para obtenção de linhas de dados de instantes dos acidentes para treinamento do sistema foi bastante reduzido. Apenas 14 dos 19 parâmetros selecionados foram efetivamente utilizados, levando à ativação de 12 das 20 tarefas de segurança selecionadas. Para continuação desta pesquisa poder-se-ia fazer o treinamento do sistema com um maior número de cenários, nos quais seja ativado um maior número de tarefas de segurança, para verificar se isto poderá melhorar significativamente a capacidade do sistema de obter resultados satisfatórios de dados com ruído.

6. REFERÊNCIAS

- [1] Comissão Nacional de Energia Nuclear. Norma CNEN-NE-1.01 – Licenciamento de Operadores de Reatores Nucleares, 1979.
- [2] United States Nuclear Regulatory Commission. NUREG-0578 - TMI-2 Lessons Learned Task Force Status Report and Short Term Recommendations, 1979.
- [3] Eletrobrás Termonuclear S/A. Final Safety Analysis Report – Central Nuclear Almirante Álvaro Alberto UNIT 1, 2007
- [4] Eletrobrás Termonuclear S/A. Manual de Operação da Usina, Unidade 1, 2008.
- [5] Cordon, O. et al., *Genetic Fuzzy Systems*. 1 ed., Singapore, World Scientific Publishing Co. Pte. Ltd. , 2001.
- [6] Hollnagel, E., *CREAM - Cognitive Reliability and Error Analysis Method*, 1 ed., Oxford, Elsevier Science Ltd. , 1998.
- [7] Eletrobrás Termonuclear S/A. Manual de Operação da Usina - Unidade 2, Volume 3, Procedimentos de Emergência, 2007.
- [8] Grant, S., “Aiding decisions by recognizing unexpected situations”. In: *5th European conference on cognitive science approaches to process control (CSAPC'95)*, pp. 358 – 367, Espoo, Finland. August 30 - September 1, 1995.
- [9] Kosko, B., *Fuzzy Engineering*, 1 ed., New Jersey, Prentice-Hall, Inc. , 1997.
- [10] Alvarenga, M.A.B., Martinez, A.S., Schirru, R., “Adaptive Vector Quantization Optimized by Genetic Algorithm for Real-Time Diagnosis Through Fuzzy Sets”. *Nuclear Technology*, vol. 120, no. 3, p. 188-197, 1997.
- [11] Uhrig, R.E., Tsoukalas, L.H., “Soft Computing Technologies in Nuclear Engineering Applications”. *Progress in Nuclear Energy* 34 p.13, 1999.
- [12] Hwang, B.C., “Intelligent Control for a Nuclear Power Plant Using Artificial Neural Networks”. In: *IEEE International Conference in Neural Networks*. pp. 2580, 1994.
- [13] Kim, K., Bartlett, E.B., “Error Prediction for a Nuclear Power Plant Fault Diagnostic Adviser Using Neural Networks”. *Nuclear Technology* 108 pp.283, 1994.
- [14] Nabeshima, K., et al., “Real-Time Nuclear Power Plant Monitoring with Neural Networks”. *Journal of Nuclear Science and Technology* vol. 35 no. 2, pp.93-100, 1998.

- [15] Uhrig, R.E., Hines, J.W., Nelson, W.R., "Integration of Artificial Intelligence Systems for Nuclear Power Plant Surveillance and Diagnostics" In: *Special Meeting on Instrumentation and Control of the Halden Research Center*, Lillhammer, Norway, March 28-21, 1998.
- [16] Marseguerra, M., et al., "Fuzzy Identification of Transients in Nuclear Power Plants". *Nuclear Engineering and Design* 225, p. 285-294, 2002.
- [17] Lee, M., et al., "An Expert System Development for Operating Procedure Monitoring of PWR Plants". In: *Computational Science and its Applications*, Springer Berlin/Heidelberg, pp. 483-490, 2005.
- [18] Lee, S.J., Seong, P.H., "A Dynamical Neural Network Based Accident Diagnosis Advisory System for Nuclear Power Plants". *Progress in Nuclear Energy* vol. 46, no. 3-4, pp. 268-281, 2005.
- [19] Santosh, T.V., et al., "Application of Artificial Neural Networks to Nuclear Power Plant Transient Diagnosis". *Reliability Engineering and System Safety* 92, pp. 1468-1472, 2006.
- [20] Shi, X., Xie, C., Wang, Y., "Nuclear Power Plant Fault Diagnosis Based on Genetic-RBF Neural Network". *Journal of Marine Science and Applications*, Vol. 5, no. 3, pp. 57-62, 2006.
- [21] Chen, Y., Narita, M., Yamada, T., "Nuclear Reactor Diagnostic System Using Genetic Algorithm (GA) Trained Neural Network". *Electrical Engineering in Japan*, Vol. 115- 5, pp. 88-99, 2007.
- [22] Mo, K., Lee, S.J., Seong, P.H., "A Dynamic Neural Network Aggregation Model for Transient Diagnosis in Nuclear Power Plants". *Progress in Nuclear Energy* 49, pp. 262-272, 2007.
- [23] Yang, Y.O., Chang, S.H., "A diagnostic Expert System for the Nuclear Power Plant Based on the Hybrid Knowledge Approach". *IEEE Transactions on Nuclear Science*, vol. 36-6, partes 1-2, 1989.
- [24] Cheon, S.W., Chang, S.H., Chung, H.Y., "Development Strategies of an Expert System for Multiple Alarm Processing and Diagnosis in Nuclear Power Plants". *IEEE Transactions on Nuclear Science*, vol. 40-1, pp. 21-30, 1993.
- [25] Varde, P.V., Sankar, S., Verma, A.K., "An Operator Support System for Research Reactor Operations and Fault Diagnosis Through a Connectionist Framework and PSA Based Knowledge Based Systems". *Reliability Engineering and Systems Safety*, vol. 60, no. 1, pp. 53-69, 1998.

- [26] Vinod, S.G., et al., "Symptom Based Diagnostic System for Nuclear Power Plant Operations Using Artificial Neural Networks". *Reliability Engineering and Systems Safety*, vol. 82, pp. 33-40, 2003.
- [27] Vinod, S.G., et al., "Neural Networks Based Operator Support System for Event Identification in NPPs". *BARC Newsletter* 251, 2004.
- [28] Lee, S.J., Seong, P.H., "Development of an Integrated Decision Support System to Aid Cognitive Activities of Operators". *Nuclear Engineering and Technology*, vol. 39, no. 6, 2007.
- [29] Soares, H.V., Schirru, R., Alvarenga, M.A.B., "On the Proposal of an Intelligent Support System with a Cognitive Architecture Based on Contextual Modules for the Operators of Nuclear Power Plants". *International Nuclear Atlantic Conference – INAC*, Santos, outubro, 2007.
- [30] Thompson, C.M, et al., "The Application of ATHEANA: A Technique for Human Error Analysis". *IEEE Sixth Annual Human Factors Meeting*, 1997.
- [31] Kasabov, N.K., *Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering*, 1 ed., Cambridge, The MIT Press, 1996.
- [32] Liow, R., Vidal, J., "A dual network expert system", In: *Anais do Congresso: IEEE International Joint Conference on Neural Networks*, Volume 2, pp.1670-1674, Singapura, Novembro, 1991.
- [33] Card, S. K., Moran, T.P., Newell, A., *The Psychology of Human-Computer Interaction*, Hillsdale, New Jersey, Lawrence Erlbaum Associates, 1983.
- [34] Card, S. K., Newell, A., "Cognitive Architectures" In: *Human Performance Models for Computer-Aided Engineering*, 176. J.L.Elkind, S.K.Card, J.Hochberg, B.M.Huey (Eds.) San Diego, CA Academic Press Inc., 1985.
- [35] Baron, S., "Approaches to Human Performance Modeling" In: *Sheldon Baron, Dana S. Kruser and Beverly M. Huey (Eds.) Quantitative Modeling of Human Performance in Complex, Dynamic Systems* , 14, 19, 162, 457. Washington D.C. National Academy Press, 1990.
- [36] Card, S. K., T.P., Newell, A., *The Psychology of Human-Computer Interaction*, Hillsdale, New Jersey. Lawrence Erlbaum Associates, 1983.
- [37] Cooper, R. P., *Modelling High-Level Cognitive Processes*, 1 ed. Mahwah, New Jersey, Lawrence Erlbaum Associates, Inc., 2002.
- [38] Jang, J. S. R., Sun, C.T., Mizutani, E., *Neuro Fuzzy and Soft Computing*, Upper Sadle River, New Jersey, Prentice Hall, 1997.

- [39] Kasabov, N. K., *Foundation of Neural Networks, Fuzzy Systems, and Knowledge Engineering*, 1 ed., Cambridge, Massachusetts, MIT Press, 1996.
- [40] Hassoun, M. H., *Fundamentals of Artificial Neural Networks*, 1 ed., Cambridge, The MIT Press, 1995.
- [41] Kosko, B., *Neural Networks and Fuzzy System*. Englewood Clifs, New Jersey. Prentice Hall, 1992.
- [42] Ward Systems Group, Inc., *Neuroshell 2 Manual* 2nd. Edition, 1996.
- [43] United States Nuclear Regulatory Commission. NUREG/CR-6126 – Cognitive Skill Training for Nuclear Power Plant Operational Decision Making, 1994.

APÊNDICE 1

REGRAS E CONDIÇÕES

1. INJEÇÃO DE SEGURANÇA DE ALTA PRESSÃO

CONDIÇÕES:

Pressão no Pressurizador < 123 Kgf/cm²

ou

Pressão ½ Geradores de Vapor < 42 kgf/cm²

ou

Pressão na contenção > 0,28 kgf/cm²

2. INJEÇÃO DO ACUMULADOR

CONDIÇÕES:

Sinal de Injeção de segurança

&

Pressão no Sistema de Refrigeração do Reator < 53 kgf/cm²

3. INJEÇÃO DE SEGURANÇA DE BAIXA PRESSÃO

CONDIÇÕES:

Sinal de Injeção de Segurança

&

Pressão no Sistema de Refrigeração do Reator < 9 kgf/cm²

4. INJEÇÃO DE ÁCIDO BÓRICO

CONDIÇÕES:

Sinal de desarme do reator requerido

&

(Fluxo de nêutrons na Faixa de Potencia > 5%)

APÊNDICE 1

Ou

Taxa de partida na Faixa Intermediária > 0

Ou

Taxa de partida na Faixa da Fonte > 0,2 DPM)

5. RECIRCULAÇÃO DO POÇO

CONDIÇÕES:

Sinal de injeção de segurança

&

Nível no Tanque de água de carregamento < 30%

&

Nível no poço da contenção > 1,5 m

6. ISOLAÇÃO DA CONTENÇÃO

CONDIÇÕES:

Pressão na contenção > 0,28 kgf/cm²

7. ISOLAÇÃO DA VENTILAÇÃO DA CONTENÇÃO

CONDIÇÕES:

Pressão na contenção > 0,28 kgf/cm²

8. RESFRIAMENTO DA CONTENÇÃO

CONDIÇÕES:

Pressão na contenção > 0,28 kgf/cm²

APÊNDICE 1

9. SPRAY DA CONTENÇÃO

CONDIÇÕES:

Pressão na contenção > 1,62 kgf/cm²

10. ÚLTIMA FONTE FRIA

CONDIÇÕES:

Sinal de injeção de segurança

11. SISTEMA DE VENTILAÇÃO DA SALA DE CONTROLE EM MODO DE EMERGÊNCIA

CONDIÇÕES:

Pressão no Pressurizador < 123 Kgf/cm²

ou

Pressão ½ Geradores de Vapor < 42 kgf/cm²

ou

Pressão na contenção > 0,28 kgf/cm²

12. SISTEMA DE VENTILAÇÃO DA ÁREA CONTROLADA EM MODO DE EMERGÊNCIA

CONDIÇÕES:

Pressão no Pressurizador < 123 Kgf/cm²

ou

Pressão ½ Geradores de Vapor < 42 kgf/cm²

ou

Pressão na contenção > 0,28 kgf/cm²

APÊNDICE 1

13. AGUA DE ALIMENTAÇÃO AUXILIAR

CONDIÇÕES:

Desarme do reator

&

Temperatura media do refrigerante > 170 C

14. ISOLAMENTO DA AGUA DE ALIMENTAÇÃO PRINCIPAL

CONDIÇÕES:

Nível ½ Geradores de Vapor > 92%

15. ISOLAMENTO DAS LINHAS DE VAPOR PRINCIPAL

CONDIÇÕES:

Injeção de segurança

&

(Vazão de vapor > 2035 ton/h

Ou

(Vazão de vapor > 740 ton/h

Ou

Temperatura media < 290 C))

Ou

Pressão na contenção > 1,05 kgf/cm²

16. RESFRIAMENTO RÁPIDO DO SISTEMA DE REFRIGERAÇÃO DO REATOR

CONDIÇÕES:

Injeção de segurança

&

Nível alto de radiação em 1 ou 2 Geradores de Vapor

APÊNDICE 1

17. DESPRESSURIZAÇÃO RÁPIDA DO SISTEMA DE REFRIGERAÇÃO DO REATOR

CONDIÇÕES:

Injeção de segurança

&

Resfriamento rápido do SRR concluído

18. ISOLAMENTO DE GERADOR DE VAPOR FALHO

CONDIÇÕES:

Pressão em 1 de 2 Geradores de Vapor < 2 kgf/cm²

Ou

Vazão de vapor > 2035 ton/h

19. RESFRIAMENTO DO SISTEMA DE REFRIGERAÇÃO DO REATOR POR DRENAGEM E ALIMENTAÇÃO

CONDIÇÕES:

Isolamento da linha de alimentação principal

&

(nível em 2 Geradores de Vapor < 3%

Ou

Pressão no pressurizador > 164 kgf/cm²)

APÊNDICE 2

PROGRAMAS “TREINASUP” E “TESTEDAREDEOUT7”

PROGRAMA “TREINASUP”

```
npar = 19;
ntasks = 21;
npartx = [4 4 2 2 2 2 4 4 2 2 2 2 2 2 4 4 4 2 2];
nparty = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];
nregras = 21;
eta = 0.1;
mm=7;
nam=28421+17373+29342+26091+21437+16059+31775+33075;
namot=0;
r=1.8;
alfa=0.10;
opcao=0;

% leitura de vetores e matrizes de conjuntos Gaussianos e conexões entre camadas
m2=[0.5 0.5 0.5 0.5 0.5 1 0.5 1 0.5 1 1 0.5 0.5 0.5 0.5 0.5 1 1 1; 0.7 0.7 2 2 5 5 1 0.5 3 10 10 2
5 2 0.7 0.7 0.5 5 5; 1 1 0 0 0 0 2 2 0 0 0 0 0 0 1 1 2 0 0; 2 2 0 0 0 0 5 5 0 0 0 0 0 0 2 2 3 0 0]'

for i = 1:npar
for j = 1:npartx(i)
if (j~=1)
V=abs(m2(i,j)-m2(i,j-1));
sigma2(i,j) = V/r;
end
end
end

for i = 1:npar
for j = 1:npartx(i)
if j==1
sigma2(i,1)=sigma2(i,2);
end
end
end
```

APÊNDICE 2

```
for j = 1:npar
for k = 1:npartx(j)
for i = 1:nregras
    w23(i,j,k) = 0.0;
end
end
end
```

```
for j=1:2
    w23(1,5,j)=1;
    w23(19,6,j)=1;
    w23(2,14,j)=1;
    w23(5,3,j)=1;
    w23(5,9,j)=1;
    w23(6,19,j)=1;
    w23(8,4,j)=1;
    w23(10,10,j)=1;
    w23(10,11,j)=1;
    w23(12,5,1)=1;
    w23(20,6,1)=1;
    w23(15,4,j)=1;
    w23(17,10,j)=1;
    w23(17,11,j)=1;
    w23(14,4,j)=1;
end
```

```
for j=1:4
    w23(13,15,j)=1;
    w23(13,16,j)=1;
    w23(13,7,j)=1;
    w23(13,8,j)=1;
    w23(11,7,j)=1;
    w23(11,8,j)=1;
    w23(9,7,j)=1;
    w23(9,8,j)=1;
    w23(13,2,j)=1;
    w23(7,2,j)=1;
    w23(10,2,j)=1;
```

APÊNDICE 2

```
w23(18,17,j)=1;
w23(11,1,j)=1;
w23(3,1,j)=1;
w23(4,1,j)=1;
end

w23(16,1,4)=1;
w23(21,1,1)=1;
w23(21,1,2)=1;
w23(16,1,3)=1;

for i = 1:ntasks
y(i)=0;
y1(i)=0;
for j = 1:nparty(i)
m4(i,j)=0.5+j;
sigma4(i,j)=0.2+j;
for k = 1:nregras
w34(i,j,k) = 1.0;
end
end
end

% leitura dos dados de treinamento nao-supervisionado nao-competitivo

for ncol=1:mm
'NUMERO DO ACIDENTE'
ncol
if
ncol==1,nome1='PRIM1';nome2='CAA1';nome3='GV1';nome4='PN1';nome5='RAD1';nome6='
POWER1';namo=28421;end
if
ncol==2,nome1='PRIM2';nome2='CAA2';nome3='GV2';nome4='PN2';nome5='RAD2';nome6='
POWER2';namo=17373;end
if
ncol==3,nome1='PRIM3';nome2='CAA3';nome3='GV3';nome4='PN3';nome5='RAD3';nome6='
POWER3';namo=29342;end
```

APÊNDICE 2

```
if
ncol==4,nome1='PRIM4';nome2='CAAA4';nome3='GV4';nome4='PN4';nome5='RAD4';nome6='
POWER4';namo=26091;end
if
ncol==5,nome1='PRIM5';nome2='CAAA5';nome3='GV5';nome4='PN5';nome5='RAD5';nome6='
POWER5';namo=21437;end
if
ncol==6,nome1='PRIM6';nome2='CAAA6';nome3='GV6';nome4='PN6';nome5='RAD6';nome6='
POWER6';namo=33084;end
if
ncol==7,nome1='PRIM7';nome2='CAAA7';nome3='GV7';nome4='PN7';nome5='RAD7';nome6='
POWER7';namo=31681;end
if
ncol==8,nome1='PRIM8';nome2='CAAA8';nome3='GV8';nome4='PN8';nome5='RAD8';nome6='
POWER8';namo=16059;end

%
% leitura dos dados de treinamento das amostras dos acidentes
%
eval(['load ',nome1,'.dat']);
PRIM=eval(nome1);
[k,n]=size(PRIM);
for i=1:k
for j=1:n
if PRIM(i,j)<0.0,PRIM(i,j)=0.0;end
end
end
eval(['load ',nome2,'.dat']);
CAAA=eval(nome2);
[k,n]=size(CAAA);
for i=1:k
for j=1:n
if CAAA(i,j)<0.0,CAAA(i,j)=0.0;end
end
end

if (ncol==6 | ncol==7)
for nn = 1:namo
CA(nn)=0;
```

APÊNDICE 2

```
end
for nn = 1:namo
CA(nn)=CAAA(nn,2);
CAAA(nn,2)=CAAA(nn,4);
CAAA(nn,4)=CA(nn);
end
end

eval(['load ',nome3,'.dat']);
GV=eval(nome3);
[k,n]=size(GV);
for i=1:k
for j=1:n
if GV(i,j)<0.0,GV(i,j)=0.0;end
end
end
eval(['load ',nome4,'.dat']);
PN=eval(nome4);
[k,n]=size(PN);
for i=1:k
for j=1:n
if PN(i,j)<0.0,PN(i,j)=0.0;end
end
end
eval(['load ',nome5,'.dat']);
RAD=eval(nome5);
[k,n]=size(RAD);
for i=1:k
for j=1:n
if RAD(i,j)<0.0,RAD(i,j)=0.0;end
end
end

eval(['load ',nome6,'.dat']);
POWER=eval(nome6);
[k,n]=size(POWER);
for i=1:k
for j=1:n
if POWER(i,j)<0.0,POWER(i,j)=0.0;end
```

APÊNDICE 2

```
end  
end
```

```
for i=1:35000  
for j=1:npar  
x1(i,j)=0.0;  
desl(i)=0.0;  
end  
end
```

```
for nn = 1:namo
```

```
x1(nn,1)=PRIM(nn,2)/100;  
% pressao do primario
```

```
x1(nn,2)=PRIM(nn,3)/200;  
% temperatura media do primario
```

```
x1(nn,3)=CAAA(nn,4);  
% nivel do poco da contencao
```

```
x1(nn,4)=CAAA(nn,2)/0.2;  
% pressao da contencao
```

```
x1(nn,5)=GV(nn,4)/50;  
% pressao do GV1
```

```
x1(nn,6)=GV(nn,5)/50;  
% pressao do GV2
```

```
x1(nn,7)=GV(nn,2)/50;  
% nivel do GV1
```

```
x1(nn,8)=GV(nn,3)/50;  
% nivel do GV2
```

```
x1(nn,9)=CAAA(nn,5)/30;  
% nivel do TAR
```

APÊNDICE 2

```
x1(nn,10)=GV(nn,6)/10;
% vazao de vapor no GV1

x1(nn,11)=GV(nn,7)/10;
% vazao de vapor no GV2

x1(nn,12)=PN(nn,2)/0.2;
% taxa de partida da faixa da fonte

x1(nn,13)=POWER(nn,6)/200;
% taxa de partida da faixa intermediaria

x1(nn,14)=PN(nn,6)/10;
% fluxo de neutrons da faixa de potencia

x1(nn,15)=CAAA(nn,6)/50;
% vazao da agua da alimentacao auxiliar do GV1

x1(nn,16)=CAAA(nn,7)/50;
% vazao da agua de alimentacao auxiliar do GV2

x1(nn,17)=PRIM(nn,5)/10;
% grau de sub-resfriamento

x1(nn,18)=RAD(nn,3)/5;
% nivel de radiacao do secundario A

x1(nn,19)=RAD(nn,4)/5;
% nivel de radiacao do secundario B

end

% aqui começa o loop de treinamento de um cenario especifico
IS=0;
ILVP=0;
TRIP=0;
RADI=0;
for nn=1:namo
namot=namot+1;
```

APÊNDICE 2

```
% teste de saida - identificação das tarefas de segurança por regras
if (RAD(nn,3)-1.95424>0 | RAD(nn,4)-1.845>0)
    RADI=1;
end

for i=1:ntasks
    y(i)=0;
    y1(i)=0;
end

if nn>700
if (POWER(nn-600,6)==0 & (ncol==1 | ncol==2 | ncol==3 | ncol==4 | ncol==5))
    % desligamento do reator
    TRIP=1;
    desl(nn)=1;
end
end

if nn>700
if (POWER(nn-600,6)<30 & (ncol==6 | ncol==7 | ncol==8))
    % desligamento do reator
    TRIP=1;
    desl(nn)=1;
end
end

if desl(nn)==1

if (GV(nn,4)<42)
    % pressão do GV1
    y(1) = 1;
    % injeção de segurança de alta pressão e última fonte fria
    IS = 1;
end

if (GV(nn,5)<42)
    % pressão do GV2
```

APÊNDICE 2

```
y(19) = 1;
% injeção de segurança de alta pressão e ultima fonte fria
IS = 1;
end

if (CAAA(nn,2)>=0.17)
% pressão da contenção
IS=1;
y(14)=1;
% realinhamento dos sistemas de ventilação
% injeção de segurança de alta pressão e ultima fonte fria
% isolamento da contenção, isolamento da ventilação da contenção e resfriamento da
contenção
end

if (PRIM(nn,2)<123)
% pressao do primario
y(16)=1;
IS=1;
% injeção de segurança de alta pressão e ultima fonte fria
% realinhamento dos sistemas de ventilação
end

if ((PN(nn,6)>5 | PN(nn,4)>0 | PN(nn,2)>0.2) & TRIP==1)
% fluxo de neutrons FP, taxa de partida F1 e taxa de partida FF
y(2) = 1;
% injeção de ácido bórico
end

if (IS==1 & PRIM(nn,2)<52)
% pressão do primário
y(3) = 1;
% injeção do acumulador
end

if (IS==1 & PRIM(nn,2)<9)
% pressão do primário
y(4) = 1;
% injeção de segurança de baixa pressão
```

APÊNDICE 2

end

if (IS==1 & CAAA(nn,5)<=35 & CAAA(nn,4)>=1.50)

% nível do TAR, nível do poço da contenção

y(5) = 1;

% recirculação do poço da contenção

end

if (IS==1 & RADII==1)

% nível de radiação no secundário

y(6) = 1;

% resfriamento rápido do primário e despressurização rápida do primário

end

if (PRIM(nn,3)>170)

% temperatura média do primário

y(7) = 1;

% água de alimentação auxiliar

end

if (CAAA(nn,2)>1.60)

% pressão da contenção

y(8) = 1;

% spray da contenção

% desligamento das BRRs

end

if (GV(nn,2)>92)

% nível do GV1 e do GV2

y(9) = 1;

% isolamento da água de alimentação

end

if (GV(nn,3)>92)

% nível do GV1 e do GV2

y(9) = 1;

% isolamento da água de alimentação

end

APÊNDICE 2

```
if (IS==1 & ((GV(nn,6)>2030 | GV(nn,7)>2030) | ((GV(nn,6)>740 | GV(nn,7)>740) &
(PRIM(nn,3)<290))))
% vazao de vapor no GV1 e GV2, temperatura media do primário e pressão da contenção
y(10) = 1;
% isolamento das linhas de vapor principal
ILVP=1;
end

if CAAA(nn,2)>1.05
% pressão da contenção
y(15)=1;
ILVP=1;
% isolamento das linhas de vapor principal
end

if (ILVP==1 & ((GV(nn,2)<3 & GV(nn,3)<3) | PRIM(nn,2)>164))
% nível do GV1 e do GV2, pressão do primário
Y(11) = 1;
% resfriamento do primário por drenagem e alimentação
end

if (GV(nn,4)<2)
% pressões de vapor dos GVs
y(12) = 1;
% isolamento do gerador de vapor falho
end

if (GV(nn,5)<2)
% pressões de vapor dos GVs
y(20) = 1;
% isolamento do gerador de vapor falho
end

if (GV(nn,6)>2035 | GV(nn,7)>2035)
% vazoes de vapor dos GVs
y(17) = 1;
% isolamento do gerador de vapor falho
end
```

APÊNDICE 2

```
if (((GV(nn,2)<4 & GV(nn,3)<4) & (CAAA(nn,6)+CAAA(nn,7)<60) & (PRIM(nn,2)>GV(nn,4) |  
PRIM(nn,2)>GV(nn,5)) & PRIM(nn,3)>170 & TRIP==1))
```

```
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos  
GVs, temperatura media do primário, pressão da contenção e grau de sub-resfriamento
```

```
y(13) = 1;
```

```
% desligamento das BRRs
```

```
end
```

```
if (IS==1 & PRIM(nn,5)<20)
```

```
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos  
GVs, temperatura media do primário, pressão da contenção e grau de sub-resfriamento
```

```
y(18) = 1;
```

```
% desligamento das BRRs
```

```
end
```

```
if (PRIM(nn,2)<95)
```

```
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos  
GVs, temperatura media do primário, pressão da contenção e grau de sub-resfriamento
```

```
y(21) = 1;
```

```
% desligamento das BRRs
```

```
end
```

```
% treinamento competitivo não-supervisionado nao-competitivo
```

```
% camada 2
```

```
for kk=1:ntasks
```

```
for i = 1:npar
```

```
for j = 1:npartx(i)
```

```
mod1(j) = abs(x1(nn,i) - m2(i,j));
```

```
end
```

```
[V,k] = min(mod1);
```

```
if y(kk)==1 & w23(kk,i,k)>0
```

```
m2(i,k) = m2(i,k) + alfa * (x1(nn,i)-m2(i,k));
```

```
end
```

```
end
```

```
end
```

APÊNDICE 2

```
% camada 4

for i = 1:ntasks
    if y(i)==1
        for j = 1:nparty(i)
            mod3(j) = abs(y(i) - m4(i,j));
        end
        [V,k] = min(mod3);
        m4(i,k) = m4(i,k) + alfa * (y(i) - m4(i,k));
    end
end

end
end
end

for i = 1:npar
    for j = 1:npartx(i)
        if (j~=1)
            V=abs(m2(i,j)-m2(i,j-1));
            sigma2(i,j) = V/r;
        end
    end
end

for i = 1:npar
    for j = 1:npartx(i)
        if j==1
            sigma2(i,1)=sigma2(i,2);
        end
    end
end

for i = 1:ntasks
    for j = 1:nparty(i)
        if (j~=1)
```

APÊNDICE 2

```
V=abs(m4(i,j)-m4(i,j-1));
sigma4(i,j) = V/r;
end
if j==1
    sigma4(i,j)=1.5;
end

end

end

save treinasup m2 sigma2 m4 sigma4;

for i = 1:ntasks
for j = 1:nparty(i)
m4n(i,j)=m4(i,j);
sigma4n(i,j)=sigma4(i,j);
erro4(i,j)=0;
end
end

m2
sigma2
m4;
sigma4;

load treinasup m2 sigma2 m4 sigma4;

for i = 1:ntasks
for j = 1:nparty(i)
for k = 1:nregras
w34(i,j,k) = 0.0;
end
end
end
%w34
```

APÊNDICE 2

```
% leitura dos dados de treinamento nao-supervisionado competitivo

for ncol=1:mm
    ncol
    if
        ncol==1,nome1='PRIM1';nome2='CAA1';nome3='GV1';nome4='PN1';nome5='RAD1';nome6='
        POWER1';namo=28421;end
    if
        ncol==2,nome1='PRIM2';nome2='CAA2';nome3='GV2';nome4='PN2';nome5='RAD2';nome6='
        POWER2';namo=17373;end
    if
        ncol==3,nome1='PRIM3';nome2='CAA3';nome3='GV3';nome4='PN3';nome5='RAD3';nome6='
        POWER3';namo=29342;end
    if
        ncol==4,nome1='PRIM4';nome2='CAA4';nome3='GV4';nome4='PN4';nome5='RAD4';nome6='
        POWER4';namo=26091;end
    if
        ncol==5,nome1='PRIM5';nome2='CAA5';nome3='GV5';nome4='PN5';nome5='RAD5';nome6='
        POWER5';namo=21437;end
    if
        ncol==6,nome1='PRIM6';nome2='CAA6';nome3='GV6';nome4='PN6';nome5='RAD6';nome6='
        POWER6';namo=33084;end
    if
        ncol==7,nome1='PRIM7';nome2='CAA7';nome3='GV7';nome4='PN7';nome5='RAD7';nome6='
        POWER7';namo=31681;end
    if
        ncol==8,nome1='PRIM8';nome2='CAA8';nome3='GV8';nome4='PN8';nome5='RAD8';nome6='
        POWER8';namo=16059;end

%
% leitura dos dados de treinamento das amostras dos acidentes
%
eval(['load ',nome1,'.dat']);
PRIM=eval(nome1);
[k,n]=size(PRIM);
for i=1:k
    for j=1:n
        if PRIM(i,j)<0.0,PRIM(i,j)=0.0;end
```

APÊNDICE 2

```
end
end
eval(['load ',nome2,'.dat']);
CAAA=eval(nome2);
[k,n]=size(CAAA);
for i=1:k
for j=1:n
if CAAA(i,j)<0.0,CAAA(i,j)=0.0;end
end
end
```

```
if (ncol==6 | ncol==7)
for nn = 1:namo
CA(nn)=0;
end
for nn = 1:namo
CA(nn)=CAAA(nn,2);
CAAA(nn,2)=CAAA(nn,4);
CAAA(nn,4)=CA(nn);
end
end
```

```
eval(['load ',nome3,'.dat']);
GV=eval(nome3);
[k,n]=size(GV);
for i=1:k
for j=1:n
if GV(i,j)<0.0,GV(i,j)=0.0;end
end
end
eval(['load ',nome4,'.dat']);
PN=eval(nome4);
[k,n]=size(PN);
for i=1:k
for j=1:n
if PN(i,j)<0.0,PN(i,j)=0.0;end
end
end
```

APÊNDICE 2

```
eval(['load ',nome5,'.dat']);
RAD=eval(nome5);
[k,n]=size(RAD);
for i=1:k
for j=1:n
if RAD(i,j)<0.0,RAD(i,j)=0.0;end
end
end

eval(['load ',nome6,'.dat']);
POWER=eval(nome6);
[k,n]=size(POWER);
for i=1:k
for j=1:n
if POWER(i,j)<0.0,POWER(i,j)=0.0;end
end
end

for i=1:35000
for j=1:npar
x1(i,j)=0.0;
desl(i)=0.0;
end
end

for nn = 1:namo

x1(nn,1)=PRIM(nn,2)/100;
% pressao do primario

x1(nn,2)=PRIM(nn,3)/200;
% temperatura media do primario

x1(nn,3)=CAAA(nn,4);
% nivel do poco da contencao

x1(nn,4)=CAAA(nn,2)/0.2;
% pressao da contencao
```

APÊNDICE 2

$x1(nn,5)=GV(nn,4)/50;$
% pressao do GV1

$x1(nn,6)=GV(nn,5)/50;$
% pressao do GV2

$x1(nn,7)=GV(nn,2)/50;$
% nivel do GV1

$x1(nn,8)=GV(nn,3)/50;$
% nivel do GV2

$x1(nn,9)=CAAA(nn,5)/30;$
% nivel do TAR

$x1(nn,10)=GV(nn,6)/10;$
% vazao de vapor no GV1

$x1(nn,11)=GV(nn,7)/10;$
% vazao de vapor no GV2

$x1(nn,12)=PN(nn,2)/0.2;$
% taxa de partida da faixa da fonte

$x1(nn,13)=POWER(nn,6)/200;$
% taxa de partida da faixa intermediaria

$x1(nn,14)=PN(nn,6)/10;$
% fluxo de neutrons da faixa de potencia

$x1(nn,15)=CAAA(nn,6)/50;$
% vazao da agua da alimentacao auxiliar do GV1

$x1(nn,16)=CAAA(nn,7)/50;$
% vazao da agua de alimentacao auxiliar do GV2

$x1(nn,17)=PRIM(nn,5)/10;$
% grau de sub-resfriamento

APÊNDICE 2

```
x1(nn,18)=RAD(nn,3)/5;
% nivel de radiacao do secundario A

x1(nn,19)=RAD(nn,4)/5;
% nivel de radiacao do secundario B

end

% aqui começa o loop de treinamento de um cenario especifico
IS=0;
ILVP=0;
TRIP=0;
RADI=0;
for nn=1:namo
namot=namot+1;

% teste de saida - identificação das tarefas de segurança por regras

if (RAD(nn,3)-1.95424>0 | RAD(nn,4)-1.84510>0)
    RADI=1;
end

for i=1:ntasks
y(i)=0;
y1(i)=0;
end

if nn>700
if (POWER(nn-600,6)==0 & (ncol==1 | ncol==2 | ncol==3 | ncol==4 | ncol==5))
    % desligamento do reator
    TRIP=1;
    desl(nn)=1;
end
end

if nn>700
if (POWER(nn-600,6)<30 & (ncol==6 | ncol==7 | ncol==8))
```

APÊNDICE 2

```
% desligamento do reator
TRIP=1;
desl(nn)=1;
end
end

if desl(nn)==1

if (GV(nn,4)<42)
% pressão do GV1
y(1) = 1;
% injeção de segurança de alta pressão e ultima fonte fria
IS = 1;
end

if (GV(nn,5)<42)
% pressão do GV2
y(19) = 1;
% injeção de segurança de alta pressão e ultima fonte fria
IS = 1;
end

if (CAAA(nn,2)>=0.17)
% pressão da contenção
IS=1;
y(14)=1;
% realinhamento dos sistemas de ventilação
% injeção de segurança de alta pressão e ultima fonte fria
% isolamento da contenção, isolamento da ventilação da contenção e resfriamento da
contenção
end

if (PRIM(nn,2)<123)
% pressao do primario
y(16)=1;
IS=1;
% injeção de segurança de alta pressão e ultima fonte fria
% realinhamento dos sistemas de ventilação
end
```

APÊNDICE 2

```
if ((PN(nn,6)>5 | PN(nn,4)>0 | PN(nn,2)>0.2) & TRIP==1)
% fluxo de neutrons FP, taxa de partida F1 e taxa de partida FF
y(2) = 1;
% injeção de ácido bórico
end

if (IS==1 & PRIM(nn,2)<52)
% pressão do primário
y(3) = 1;
% injeção do acumulador
end

if (IS==1 & PRIM(nn,2)<9)
% pressão do primário
y(4) = 1;
% injeção de segurança de baixa pressão
end

if (IS==1 & CAAA(nn,5)<=35 & CAAA(nn,4)>=1.50)
% nível do TAR, nível do poço da contenção
y(5) = 1;
% recirculação do poço da contenção
end

if (IS==1 & RADl==1)
% nível de radiação no secundario
y(6) = 1;
% resfriamento rápido do primário e despressurização rápida do primário
end

if (PRIM(nn,3)>170)
% temperatura média do primário
y(7) = 1;
% água de alimentação auxiliar
end

if (CAAA(nn,2)>1.60)
% pressão da contenção
```

APÊNDICE 2

```
y(8) = 1;
% spray da contenção
% desligamento das BRRs
end

if (GV(nn,2)>92)
% nível do GV1 e do GV2
y(9) = 1;
% isolamento da água de alimentação
end

if (GV(nn,3)>92)
% nível do GV1 e do GV2
y(9) = 1;
% isolamento da água de alimentação
end

if (IS==1 & ((GV(nn,6)>2030 | GV(nn,7)>2030) | ((GV(nn,6)>740 | GV(nn,7)>740) &
(PRIM(nn,3)<290))))
% vazao de vapor no GV1 e GV2, temperatura media do primário e pressão da contenção
y(10) = 1;
% isolamento das linhas de vapor principal
ILVP=1;
end

if CAAA(nn,2)>1.05
% pressão da contenção
y(15)=1;
ILVP=1;
% isolamento das linhas de vapor principal
end

if (ILVP==1 & ((GV(nn,2)<3 & GV(nn,3)<3) | PRIM(nn,2)>164))
% nível do GV1 e do GV2, pressão do primário
Y(11) = 1;
% resfriamento do primário por drenagem e alimentação
end

if (GV(nn,4)<2)
```

APÊNDICE 2

```
% pressoes de vapor dos GVs
y(12) = 1;
% isolamento do gerador de vapor falho
end

if (GV(nn,5)<2)
% pressoes de vapor dos GVs
y(20) = 1;
% isolamento do gerador de vapor falho
end

if (GV(nn,6)>2035 | GV(nn,7)>2035)
% vazoes de vapor dos GVs
y(17) = 1;
% isolamento do gerador de vapor falho
end

if (((GV(nn,2)<4 & GV(nn,3)<4) & (CAAA(nn,6)+CAAA(nn,7)<60) & (PRIM(nn,2)>GV(nn,4) |
PRIM(nn,2)>GV(nn,5)) & PRIM(nn,3)>170 & TRIP==1))
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, pressão da contenção e grau de sub-resfriamento
y(13) = 1;
% desligamento das BRRs
end

if (IS==1 & PRIM(nn,5)<20)
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, pressão da contenção e grau de sub-resfriamento
y(18) = 1;
% desligamento das BRRs
end

if (PRIM(nn,2)<95)
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, pressão da contenção e grau de sub-resfriamento
y(21) = 1;
% desligamento das BRRs
end
```

APÊNDICE 2

```
% treinamento nao-supervisionado competitivo
```

```
% camada 2 - saída dos neurônios
```

```
for i = 1:npar
for j = 1:npartx(i)
f2(i,j) = (((x1(nn,i) - m2(i,j)) / sigma2(i,j))^2);
a2(i,j) = exp(-f2(i,j));
end
end
```

```
% camada 3 - saída dos neurônios
```

```
for i = 1:nregras
a3(i)=0;
for j=1:npar
for k=1:npartx(j)
a34(j,k)=0.0;
end
end
end
```

```
for i = 1:nregras
% a33(i,j)=0;
a33(i)=1.0;
cont=0;
for j = 1:npar
for k=1:npartx(j)
a34(j,k)=a2(j,k)*w23(i,j,k);
end
[Y,k]=max(a34(j,:));
if (a34(j,k)>0.00 & w23(i,j,k)~=0)
a33(i) = a33(i) * a34(j,k) * w23(i,j,k);
cont=cont+1;
end
end
```

APÊNDICE 2

```
if (cont~=0)
    a3(i)=a33(i);
end

end

% camada 4 - saida dos neuronios

for i = 1:ntasks
    for j = 1:nparty(i)
        f4(i,j) = (((y(i) - m4(i,j)) / sigma4(i,j))^2);
        a4(i,j) = exp(-f4(i,j));
    end
end

% camada 3 - camada 4

for i = 1:nregras
    if y(i)==1
        for j = 1:ntasks
            if y(j)==1
                for k = 1:nparty(j)
                    w34(j,k,i) = w34(j,k,i) + a4(j,k) * (-w34(j,k,i) + a3(i));
                end
            end
        end
    end
end

w34
end
end
end

save w34n w34;
```

APÊNDICE 2

```
% fim do treinamento nao-supervisionado
%comeco do treinamento supervisionado
% leitura dos dados de treinamento

load treinasup m2 sigma2 m4 sigma4;
load w34n w34;
m2
sigma2
m4;
sigma4;

for i=1:ntasks
    for j=1:nparty(i)
        m4n(i,j)=0;
        sigma4n(i,j)=0;
        if sigma4(i,j)==0
            sigma4(i,j)=1.5;
        end
    end
end

for ncol=1:mm
    ncol
    if
        ncol==1,nome1='PRIM1';nome2='CAAA1';nome3='GV1';nome4='PN1';nome5='RAD1';nome6='
        POWER1';namo=28421;end
    if
        ncol==2,nome1='PRIM2';nome2='CAAA2';nome3='GV2';nome4='PN2';nome5='RAD2';nome6='
        POWER2';namo=17373;end
    if
        ncol==3,nome1='PRIM3';nome2='CAAA3';nome3='GV3';nome4='PN3';nome5='RAD3';nome6='
        POWER3';namo=29342;end
    if
        ncol==4,nome1='PRIM4';nome2='CAAA4';nome3='GV4';nome4='PN4';nome5='RAD4';nome6='
        POWER4';namo=26091;end
```

APÊNDICE 2

```
if
ncol==5,nome1='PRIM5';nome2='CAA5';nome3='GV5';nome4='PN5';nome5='RAD5';nome6='
POWER5';namo=21437;end
if
ncol==6,nome1='PRIM6';nome2='CAA6';nome3='GV6';nome4='PN6';nome5='RAD6';nome6='
POWER6';namo=33084;end
if
ncol==7,nome1='PRIM7';nome2='CAA7';nome3='GV7';nome4='PN7';nome5='RAD7';nome6='
POWER7';namo=31681;end
if
ncol==8,nome1='PRIM8';nome2='CAA8';nome3='GV8';nome4='PN8';nome5='RAD8';nome6='
POWER8';namo=16059;end

%
% leitura dos dados de treinamento das amostras dos acidentes
%
eval(['load ',nome1,'.dat']);
PRIM=eval(nome1);
[k,n]=size(PRIM);
for i=1:k
for j=1:n
if PRIM(i,j)<0.0,PRIM(i,j)=0.0;end
end
end
eval(['load ',nome2,'.dat']);
CAA=eval(nome2);
[k,n]=size(CAA);
for i=1:k
for j=1:n
if CAA(i,j)<0.0,CAA(i,j)=0.0;end
end
end

if (ncol==6 | ncol==7)
for nn = 1:namo
CA(nn)=0;
end
for nn = 1:namo
CA(nn)=CAA(nn,2);
```

APÊNDICE 2

```
CAAA(nn,2)=CAAA(nn,4);
CAAA(nn,4)=CA(nn);
end
end

eval(['load ',nome3,'.dat']);
GV=eval(nome3);
[k,n]=size(GV);
for i=1:k
for j=1:n
if GV(i,j)<0.0,GV(i,j)=0.0;end
end
end
eval(['load ',nome4,'.dat']);
PN=eval(nome4);
[k,n]=size(PN);
for i=1:k
for j=1:n
if PN(i,j)<0.0,PN(i,j)=0.0;end
end
end
eval(['load ',nome5,'.dat']);
RAD=eval(nome5);
[k,n]=size(RAD);
for i=1:k
for j=1:n
if RAD(i,j)<0.0,RAD(i,j)=0.0;end
end
end
eval(['load ',nome6,'.dat']);
POWER=eval(nome6);
[k,n]=size(POWER);
for i=1:k
for j=1:n
if POWER(i,j)<0.0,POWER(i,j)=0.0;end
end
end
```

APÊNDICE 2

```
for i=1:35000
for j=1:npar
x1(i,j)=0.0;
desl(i)=0.0;
end
end

errt=1;
if errt==1

for nn = 1:namo

x1(nn,1)=PRIM(nn,2)/100;
% pressao do primario

x1(nn,2)=PRIM(nn,3)/200;
% temperatura media do primario

x1(nn,3)=CAAA(nn,4);
% nivel do poco da contencao

x1(nn,4)=CAAA(nn,2)/0.2;
% pressao da contencao

x1(nn,5)=GV(nn,4)/50;
% pressao do GV1

x1(nn,6)=GV(nn,5)/50;
% pressao do GV2

x1(nn,7)=GV(nn,2)/50;
% nivel do GV1

x1(nn,8)=GV(nn,3)/50;
% nivel do GV2

x1(nn,9)=CAAA(nn,5)/30;
% nivel do TAR
```

APÊNDICE 2

x1(nn,10)=GV(nn,6)/10;
% vazao de vapor no GV1

x1(nn,11)=GV(nn,7)/10;
% vazao de vapor no GV2

x1(nn,12)=PN(nn,2)/0.2;
% taxa de partida da faixa da fonte

x1(nn,13)=POWER(nn,6)/200;
% taxa de partida da faixa intermediaria

x1(nn,14)=PN(nn,6)/10;
% fluxo de neutrons da faixa de potencia

x1(nn,15)=CAAA(nn,6)/50;
% vazao da agua da alimentacao auxiliar do GV1

x1(nn,16)=CAAA(nn,7)/50;
% vazao da agua de alimentacao auxiliar do GV2

x1(nn,17)=PRIM(nn,5)/10;
% grau de sub-resfriamento

x1(nn,18)=RAD(nn,3)/5;
% nivel de radiacao do secundario A

x1(nn,19)=RAD(nn,4)/5;
% nivel de radiacao do secundario B

end

% aqui começa o loop de treinamento de um cenario especifico

IS=0;

ILVP=0;

TRIP=0;

APÊNDICE 2

```
RADI=0;
conta=0;
for nn=1:namo

    namot=namot+1;

    % teste de saida - identificação das tarefas de segurança por regras

    if (RAD(nn,3)-1.95424>0 | RAD(nn,4)-1.84510>0)
        RADI=1;
    end

    for i=1:ntasks
        y(i)=0;
        y1(i)=0;
    end

    if nn>700
        if (POWER(nn-600,6)==0 & (ncol==1 | ncol==2 | ncol==3 | ncol==4 | ncol==5))
            % desligamento do reator
            TRIP=1;
            desl(nn)=1;
        end
    end

    if nn>700
        if (POWER(nn-600,6)<30 & (ncol==6 | ncol==7 | ncol==8))
            % desligamento do reator
            TRIP=1;
            desl(nn)=1;
        end
    end

    if desl(nn)==1
        conta=conta+1;

        if (GV(nn,4)<42)
```

APÊNDICE 2

```
% pressão do GV1
y(1) = 1;
% injeção de segurança de alta pressão e ultima fonte fria
IS = 1;
end

if (GV(nn,5)<42)
% pressão do GV2
y(19) = 1;
% injeção de segurança de alta pressão e ultima fonte fria
IS = 1;
end

if (CAAA(nn,2)>=0.17)
% pressão da contenção
IS=1;
y(14)=1;
% realinhamento dos sistemas de ventilação
% injeção de segurança de alta pressão e ultima fonte fria
% isolamento da contenção, isolamento da ventilação da contenção e resfriamento da
contenção
end

if (PRIM(nn,2)<123)
% pressao do primario
y(16)=1;
IS=1;
% injeção de segurança de alta pressão e ultima fonte fria
% realinhamento dos sistemas de ventilação
end

if ((PN(nn,6)>5 | PN(nn,4)>0 | PN(nn,2)>0.2) & TRIP==1)
% fluxo de neutrons FP, taxa de partida F1 e taxa de partida FF
y(2) = 1;
% injeção de ácido bórico
end

if (IS==1 & PRIM(nn,2)<52)
% pressão do primário
```

APÊNDICE 2

```
y(3) = 1;
% injeção do acumulador
end

if (IS==1 & PRIM(nn,2)<9)
% pressão do primário
y(4) = 1;
% injeção de segurança de baixa pressão
end

if (IS==1 & CAAA(nn,5)<=35 & CAAA(nn,4)>=1.50)
% nível do TAR, nível do poço da contenção
y(5) = 1;
% recirculação do poço da contenção
end

if (IS==1 & RADi==1)
% nível de radiação no secundario
y(6) = 1;
% resfriamento rápido do primário e despressurização rápida do primário
end

if (PRIM(nn,3)>170)
% temperatura média do primário
y(7) = 1;
% água de alimentação auxiliar
end

if (CAAA(nn,2)>1.60)
% pressão da contenção
y(8) = 1;
% spray da contenção
% desligamento das BRRs
end

if (GV(nn,2)>92)
% nível do GV1 e do GV2
y(9) = 1;
% isolamento da água de alimentação
```

APÊNDICE 2

end

if (GV(nn,3)>92)

% nível do GV1 e do GV2

y(9) = 1;

% isolamento da água de alimentação

end

if (IS==1 & ((GV(nn,6)>2030 | GV(nn,7)>2030) | ((GV(nn,6)>740 | GV(nn,7)>740) &
(PRIM(nn,3)<290))))

% vazão de vapor no GV1 e GV2, temperatura média do primário e pressão da contenção

y(10) = 1;

% isolamento das linhas de vapor principal

ILVP=1;

end

if CAAA(nn,2)>1.05

% pressão da contenção

y(15)=1;

ILVP=1;

% isolamento das linhas de vapor principal

end

if (ILVP==1 & ((GV(nn,2)<3 & GV(nn,3)<3) | PRIM(nn,2)>164))

% nível do GV1 e do GV2, pressão do primário

Y(11) = 1;

% resfriamento do primário por drenagem e alimentação

end

if (GV(nn,4)<2)

% pressões de vapor dos GVs

y(12) = 1;

% isolamento do gerador de vapor falho

end

if (GV(nn,5)<2)

% pressões de vapor dos GVs

y(20) = 1;

% isolamento do gerador de vapor falho

APÊNDICE 2

```
end

if (GV(nn,6)>2035 | GV(nn,7)>2035)
% vazoes de vapor dos GVs
y(17) = 1;
% isolamento do gerador de vapor falho
end

if (((GV(nn,2)<4 & GV(nn,3)<4) & (CAAA(nn,6)+CAAA(nn,7)<60) & (PRIM(nn,2)>GV(nn,4) |
PRIM(nn,2)>GV(nn,5)) & PRIM(nn,3)>170 & TRIP==1))
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, pressão da contenção e grau de sub-resfriamento
y(13) = 1;
% desligamento das BRRs
end

if (IS==1 & PRIM(nn,5)<20)
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, pressão da contenção e grau de sub-resfriamento
y(18) = 1;
% desligamento das BRRs
end

if (PRIM(nn,2)<95)
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, pressão da contenção e grau de sub-resfriamento
y(21) = 1;
% desligamento das BRRs
end

% treinamento supervisionado on-line

% calculo do erro na saida

% camada 2 - saida dos neurônios

for i = 1:npar
for j = 1:npartx(i)
f2(i,j) = (((x1(nn,i) - m2(i,j)) / sigma2(i,j))^2);
```

APÊNDICE 2

```
a2(i,j) = exp(-f2(i,j));
end
end

% camada 3 - saída dos neurônios

for i = 1:nregras
    a3(i)=0;
    for j=1:npar
        for k=1:npartx(j)
            a34(j,k)=0.0;
        end
    end
end

for i = 1:nregras
    % a33(i,j)=0;
    a33(i)=1.0;
    cont=0;
    for j = 1:npar
        for k=1:npartx(j)
            a34(j,k)=a2(j,k)*w23(i,j,k);
        end
        [Y,k]=max(a34(j,:));
        if (a34(j,k)>0.00 & w23(i,j,k)~=0)
            a33(i) = a33(i) * a34(j,k) * w23(i,j,k);
            cont=cont+1;
        end
    end
end

if (cont~=0)
    a3(i)=a33(i);
end

end
```

APÊNDICE 2

```
% camada 4 - saída dos neurônios
```

```
for i = 1:ntasks
for k = 1:nparty(i)
f4(i,k)=0.0;
for j = 1:nregras
    if i==j
f4(i,k) = f4(i,k) + a3(j);
end
end
a4(i,k) = min(1,f4(i,k));
end
end
```

```
% camada 5 - saída de neurônios
```

```
errt=0;
```

```
for i = 1:ntasks
    soma1(i)=0;
    soma2(i)=0;
end
```

```
for i = 1:ntasks
    % if erro(i)>0 & conta>0
for j = 1:nparty(i)
f5(i,j) = (((f4(i,j) - m4(i,j)) / sigma4(i,j))^2);
%a5(i,j) = exp(-f5(i,j));
a5(i,j)=a4(i,j);
end
%end
end
```

```
a5;
```

```
for j = 1:ntasks
```

APÊNDICE 2

```
if nparty(j)~=1
soma1(j) = 0;
soma2(j) = 0;
for i = 1:nparty(j)
soma1(j) = soma1(j) + a5(j,i) * sigma4(j,i);
soma2(j) = soma2(j) + a5(j,i) * sigma4(j,i) * m4(j,i);
end

if soma1(j)~=0
y1(j) = soma2(j)/soma1(j);
else
y1(j)=0;
end

end

if nparty(j)==1

y1(j)=a5(j,1);
end

if norm(w34(:,j))==0
y1(j)=0;
end

erro(j) = y(j) - y1(j);

if (ncol==2 | ncol==4 | ncol==5 | ncol==6 | ncol==7)

erro(16)=0;
end

if (ncol==2 | ncol==4 | ncol==5 | ncol==6 | ncol==7)

erro(21)=0;
end

if (ncol==2 | ncol==4 | ncol==5 | ncol==1 | ncol==3 | (nn>775 & ncol==6) | ncol==7)
```

APÊNDICE 2

```
    erro(1)=0;
end

if (ncol==2 | ncol==4 | ncol==5 | ncol==6 | ncol==7)
    erro(6)=0;
end

if (ncol~=5)
    erro(13)=0;
end

if (ncol==2 | ncol==3 | ncol==5 | ncol==7)

    erro(18)=0;
end

if abs(erro(j))>0.21, errt=1; end

end
erro

% treinamento com retropropagação do erro

% cálculo dos parâmetros da camada 4

for i = 1:ntasks
    for j = 1:nparty(i)

        if nparty(i)~=1
            if soma1(i)~=0 & a5(i,j)~=0
                m4n(i,j) = m4(i,j) + (eta * erro(i)) * (sigma4(i,j) * a5(i,j) / soma1(i));
                parte1=(m4(i,j) * a5(i,j) * soma1(i) - soma2(i) * a5(i,j)) / (soma1(i)^2);
                if parte1~=0 & a5(i,j)~=0
                    sigma4n(i,j) = sigma4(i,j) + (eta * erro(i) * parte1);
                end
            end
        end
    end
end
if a4(i,j)==0 | parte1==0
```

APÊNDICE 2

```
m4n(i,j) = m4(i,j) + (eta * erro(i));
sigma4n(i,j) = sigma4(i,j) + (eta * erro(i));
end
end

if nparty(i)==1
m4n(i,j) = m4(i,j) + (eta * erro(i));
sigma4n(i,j) = sigma4(i,j) + (eta * erro(i));
end

end
end

% cálculo do erro na camada 4

for i = 1:ntasks
    for j = 1:nparty(i)
        erro4(i,j)=0.0;
    end
end

for i = 1:ntasks
    for j = 1:nparty(i)

        if nparty(i)~=1
            if soma1(i)~=0 & a4(i,j)~=0
                parte2=((m4(i,j) * sigma4(i,j) * soma1(i)) - (sigma4(i,j) * soma2(i))) / (soma1(i)^2);
            if parte2~=0 & a4(i,j)~=0
                erro4(i,j) = (erro(i) * parte2);
            end
            end

        if a4(i,j)==0 | parte2==0
            erro4(i,j)=erro(i);
        end

    end

end
```

APÊNDICE 2

```
if nparty(i)==1
erro4(i,j)=erro(i);
end

end
end

erro4;
a4;

for k=1:ntasks
for i=1:nparty(k)
m4(k,i) = m4n(k,i);
sigma4(k,i) = sigma4n(k,i);
end
end

% cálculo do erro da camada 3

for i = 1:nregras
erro3(i) = 0;
for j = 1:ntasks
for k=1:nparty(j)
if i==j
erro3(i) = erro3(i) + erro4(j,k);
end
end
end
end

erro3;
a3;

% cálculo do erro na camada 2

for i = 1:npar
for j = 1:npartx(i)
```

APÊNDICE 2

```
erro2(i,j)=0.0;
end
end

for i=1:nregras
    for j=1:npar
        pos(i,j)=0;
        a271(i,j)=0;
        for k=1:npartx(j)
            a27(i,j,k)=0;
        end
    end
end
end

for i=1:nregras
    for j=1:npar
        for k=1:npartx(j)
            a27(i,j,k)=a2(j,k)*w23(i,j,k);
        end
    end
end
end

for k = 1:nregras
    for i=1:npar
        [C,KI]=max(a27(k,i,:));
        a271(k,i)=C;
        pos(k,i)=KI;
    end
end
end

for k = 1:nregras
    for i=1:npar
        if a271(k,i)==0
            a271(k,i)=10000;
        end
    end
end
end
```

APÊNDICE 2

```
for k = 1:nregras
    [V,KL]=min(a271(k,:));
    KC=pos(k,KL);
    if (w23(k,KL,KC)~=0)
        erro2(KL,KC) = erro2(KL,KC) - erro3(k);
    end
end

erro2;

% cálculo dos parâmetros na camada 2

for i = 1:npar
    for j = 1:npartx(i)
        f2(i,j) = (((x1(nn,i) - m2(i,j)) / sigma2(i,j))^2);
        a22(i,j) = exp(-f2(i,j));
    end
end

for i = 1:npar
    for j = 1:npartx(i)
        sigma2(i,j) = sigma2(i,j) - (eta * erro2(i,j) * a22(i,j) * (2 * ((x1(nn,i) - m2(i,j))^2)) / (sigma2(i,j)^3));
        m2(i,j) = m2(i,j) - (eta * erro2(i,j) * a22(i,j) * (2 * (x1(nn,i) - m2(i,j))) / (sigma2(i,j)^2));
    end
end

m2;
sigma2;

end
end
% fim das amostras do cenario
save w34t;
'FIM'
end
% treinamento do proximo cenario - ncol=ncol+1
end
% fim do treinamento dos cenarios
```

APÊNDICE 2

```
save treisup m2 sigma2 m4 sigma4 /ascii;  
save w34;
```

PROGRAMA "TESTEDAREDEOUT7"

```
cogent=0  
if cogent==0  
npar = 19;  
ntasks = 21;  
npartx = [4 4 2 2 2 2 4 4 2 2 2 2 2 2 4 4 4 2 2];  
nparty = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];  
nregras = 21;  
eta = 0.1;  
mm=8;  
nam=28421+17373+29342+26091+21437+16059+31781+33084;  
namot=0;  
r=1.8;  
alfa=0.10;  
ivar=0  
ivari=5
```

% leitura de vetores e matrizes de conjuntos Gaussianos e conexões entre camadas

```
for j = 1:npar  
for k = 1:npartx(j)  
for i = 1:nregras  
w23(i,j,k) = 0.0;  
end  
end  
end
```

```
for j=1:2  
w23(1,5,j)=1;  
w23(19,6,j)=1;  
w23(2,14,j)=1;  
w23(5,3,j)=1;
```

APÊNDICE 2

```
w23(5,9,j)=1;
w23(6,19,j)=1;
w23(8,4,j)=1;
w23(10,10,j)=1;
w23(10,11,j)=1;
w23(12,5,1)=1;
w23(20,6,1)=1;
w23(15,4,j)=1;
w23(17,10,j)=1;
w23(17,11,j)=1;
w23(14,4,j)=1;
end
```

```
for j=1:4
    w23(13,15,j)=1;
    w23(13,16,j)=1;
    w23(13,7,j)=1;
    w23(13,8,j)=1;
    w23(11,7,j)=1;
    w23(11,8,j)=1;
    w23(9,7,j)=1;
    w23(9,8,j)=1;
    w23(13,2,j)=1;
    w23(7,2,j)=1;
    w23(10,2,j)=1;
    w23(18,17,j)=1;
    w23(11,1,j)=1;
    w23(3,1,j)=1;
    w23(4,1,j)=1;
end
```

```
w23(16,1,4)=1;
w23(21,1,1)=1;
w23(21,1,2)=1;
w23(16,1,3)=1;
```

```
load w34 m2 sigma2 m4 sigma4 w34;
```

APÊNDICE 2

```
m2
sigma2
m4;
sigma4;
```

```
m2(6,1)=m2(5,1);
m2(6,2)=m2(5,2);
sigma2(6,1)=sigma2(5,1);
sigma2(6,2)=sigma2(5,2);
```

```
m2
sigma2
m4;
sigma4;
```

```
% teste de identificacao das tarefas de seguranca
```

```
for ncol=1:mm
```

```
    'NUMERO DO ACIDENTE'
```

```
    ncol
```

```
IP=0;
```

```
IP1=0;
```

```
namot=0;
```

```
namot1=0;
```

```
if
```

```
ncol==1,nome1='PRIM1';nome2='CAAA1';nome3='GV1';nome4='PN1';nome5='RAD1';nome6='
POWER1';namo=28421;end
```

```
if
```

```
ncol==2,nome1='PRIM2';nome2='CAAA2';nome3='GV2';nome4='PN2';nome5='RAD2';nome6='
POWER2';namo=17373;end
```

```
if
```

```
ncol==3,nome1='PRIM3';nome2='CAAA3';nome3='GV3';nome4='PN3';nome5='RAD3';nome6='
POWER3';namo=29342;end
```

APÊNDICE 2

```
if
ncol==4,nome1='PRIM4';nome2='CAA4';nome3='GV4';nome4='PN4';nome5='RAD4';nome6='
POWER4';namo=26091;end
if
ncol==5,nome1='PRIM5';nome2='CAA5';nome3='GV5';nome4='PN5';nome5='RAD5';nome6='
POWER5';namo=21437;end
if
ncol==6,nome1='PRIM6';nome2='CAA6';nome3='GV6';nome4='PN6';nome5='RAD6';nome6='
POWER6';namo=33084;end
if
ncol==7,nome1='PRIM7';nome2='CAA7';nome3='GV7';nome4='PN7';nome5='RAD7';nome6='
POWER7';namo=31681;end
if
ncol==8,nome1='PRIM8';nome2='CAA8';nome3='GV8';nome4='PN8';nome5='RAD8';nome6='
POWER8';namo=16059;end

%
% leitura dos dados de treinamento das amostras dos acidentes
%
eval(['load ',nome1,'.dat']);
PRIM=eval(nome1);
[k,n]=size(PRIM);
for i=1:k
for j=1:n
if PRIM(i,j)<0.0,PRIM(i,j)=0.0;end
end
end
eval(['load ',nome2,'.dat']);
CAA=eval(nome2);
[k,n]=size(CAA);
for i=1:k
for j=1:n
if CAA(i,j)<0.0,CAA(i,j)=0.0;end
end
end

if (ncol==6 | ncol==7)
for nn = 1:namo
CA(nn)=0;
```

APÊNDICE 2

```
end
for nn = 1:namo
CA(nn)=CAAA(nn,2);
CAAA(nn,2)=CAAA(nn,4);
CAAA(nn,4)=CA(nn);
end
end

eval(['load ',nome3,'.dat']);
GV=eval(nome3);
[k,n]=size(GV);
for i=1:k
for j=1:n
if GV(i,j)<0.0,GV(i,j)=0.0;end
end
end
eval(['load ',nome4,'.dat']);
PN=eval(nome4);
[k,n]=size(PN);
for i=1:k
for j=1:n
if PN(i,j)<0.0,PN(i,j)=0.0;end
end
end
eval(['load ',nome5,'.dat']);
RAD=eval(nome5);
[k,n]=size(RAD);
for i=1:k
for j=1:n
if RAD(i,j)<0.0,RAD(i,j)=0.0;end
end
end

eval(['load ',nome6,'.dat']);
POWER=eval(nome6);
[k,n]=size(POWER);
for i=1:k
for j=1:n
```

APÊNDICE 2

```
if POWER(i,j)<0.0,POWER(i,j)=0.0;end  
end  
end
```

```
for i=1:namo  
for j=1:npar  
x1(i,j)=0;  
desl(i)=0;  
end  
end
```

```
mult = [100 200 1 0.1 10 50 10 50 30 10 10 0.2 200 10 50 50 10 1 1];
```

```
for nn = 1:namo
```

```
if ivari==1  
PRIM(nn,2)=(123+95)/2;  
end
```

```
x1(nn,1)=PRIM(nn,2)/100;  
% pressao do primario
```

```
x1(nn,2)=PRIM(nn,3)/200;  
% temperatura media do primario
```

```
x1(nn,3)=CAAA(nn,4);  
% nivel do poco da contencao
```

```
x1(nn,4)=CAAA(nn,2)/0.2;  
% pressao da contencao
```

```
if ivari==5  
GV(nn,4)=(41+4)/2;  
end
```

```
x1(nn,5)=GV(nn,4)/50;  
% pressao do GV1
```

APÊNDICE 2

$x1(nn,6)=GV(nn,5)/50;$

% pressao do GV2

$x1(nn,7)=GV(nn,2)/50;$

% nivel do GV1

$x1(nn,8)=GV(nn,3)/50;$

% nivel do GV2

$x1(nn,9)=CAAA(nn,5)/30;$

% nivel do TAR

$x1(nn,10)=GV(nn,6)/10;$

% vazao de vapor no GV1

$x1(nn,11)=GV(nn,7)/10;$

% vazao de vapor no GV2

$x1(nn,12)=PN(nn,2)/0.2;$

% taxa de partida da faixa da fonte

$x1(nn,13)=POWER(nn,6)/200;$

% taxa de partida da faixa intermediaria

$x1(nn,14)=PN(nn,6)/10;$

% fluxo de neutrons da faixa de potencia

$x1(nn,15)=CAAA(nn,6)/50;$

% vazao da agua da alimentacao auxiliar do GV1

$x1(nn,16)=CAAA(nn,7)/50;$

% vazao da agua de alimentacao auxiliar do GV2

$x1(nn,17)=PRIM(nn,5)/10;$

% grau de sub-resfriamento

$x1(nn,18)=RAD(nn,3)/5;$

% nivel de radiacao do secundario A

APÊNDICE 2

```
x1(nn,19)=RAD(nn,4)/5;
% nivel de radiacao do secundario B
end

for i=1:ntasks
numtasks(i)=0;
numtasks1(i)=0;
numtasks2(i)=0;
numtasks3(i)=0;
numtasks4(i)=0;
end

for i=1:npar
numvar(i)=0;
numin(i)=0;
end

% aqui começa o loop de treinamento de um cenario especifico
IS=0;
ILVP=0;
TRIP=0;
RADI=0;

for nn=1:namo
    interv(nn)=0;
    interv1(nn)=0;
    interv2(nn)=0;

% teste de saida - identificação das tarefas de segurança por regras

if (RAD(nn,3)-1.95424>0 | RAD(nn,4)-1.84510>0)
    RADI=1;
end

for i=1:ntasks
y(i)=0;
y1(i)=0;
end
```

APÊNDICE 2

```
if nn>700
if (POWER(nn-600,6)==0 & (ncol==1 | ncol==2 | ncol==3 | ncol==4 | ncol==5))
    % desligamento do reator
    TRIP=1;
    desl(nn)=1;
end
end
```

```
if nn>700
if (POWER(nn-600,6)<30 & (ncol==6 | ncol==7 | ncol==8))
    % desligamento do reator
    TRIP=1;
    desl(nn)=1;
end
end
```

```
if desl(nn)==1
```

```
if (GV(nn,4)<42)
% pressão do GV1
y(1) = 1;
% injeção de segurança de alta pressão e ultima fonte fria
IS = 1;
numvar(5)=numvar(5)+1;
numtasks2(1)=numtasks2(1)+1;
end
```

```
if (GV(nn,5)<42)
% pressão do GV2
y(19) = 1;
% injeção de segurança de alta pressão e ultima fonte fria
IS = 1;
numvar(6)=numvar(6)+1;
numtasks2(19)=numtasks2(19)+1;
end
```

```
if (CAAA(nn,2)>=0.17)
% pressão da contenção
numvar(4)=numvar(4)+1;
```

APÊNDICE 2

```
numtasks2(14)=numtasks2(14)+1;
IS=1;
y(14)=1;
% realinhamento dos sistemas de ventilação
% injeção de segurança de alta pressão e ultima fonte fria
% isolamento da contenção, isolamento da ventilação da contenção e resfriamento da
contenção
end
```

```
if (PRIM(nn,2)<123)
% pressao do primario
y(16)=1;
IS=1;
numvar(1)=numvar(1)+1;
numtasks2(16)=numtasks2(16)+1;
% injeção de segurança de alta pressão e ultima fonte fria
% realinhamento dos sistemas de ventilação
end
```

```
if ((PN(nn,6)>5 | PN(nn,4)>0 | PN(nn,2)>0.2) & TRIP==1)
% fluxo de neutrons FP, taxa de partida F1 e taxa de partida FF
y(2) = 1;
numvar(14)=numvar(14)+1;
numvar(13)=numvar(13)+1;
numvar(12)=numvar(12)+1;
numtasks2(2)=numtasks2(2)+1;
% injeção de ácido bórico
end
```

```
if (IS==1 & PRIM(nn,2)<52)
% pressão do primário
y(3) = 1;
numvar(1)=numvar(1)+1;
numtasks2(3)=numtasks2(3)+1;
% injeção do acumulador
end
```

```
if (IS==1 & PRIM(nn,2)<9)
% pressão do primário
```

APÊNDICE 2

```
y(4) = 1;
numvar(1)=numvar(1)+1;
numtasks2(4)=numtasks2(4)+1;
% injeção de segurança de baixa pressão
end

if (IS==1 & CAAA(nn,5)<=35 & CAAA(nn,4)>=1.50)
% nível do TAR, nível do poço da contenção
y(5) = 1;
numvar(9)=numvar(9)+1;
numvar(3)=numvar(3)+1;
numtasks2(5)=numtasks2(5)+1;
% recirculação do poço da contenção
end

if (IS==1 & RAD==1)
% nível de radiação no secundario
y(6) = 1;
numvar(18)=numvar(18)+1;
numvar(19)=numvar(19)+1;
numtasks2(6)=numtasks2(6)+1;
% resfriamento rápido do primário e despressurização rápida do primário
end

if (PRIM(nn,3)>170)
% temperatura média do primário
y(7) = 1;
numvar(2)=numvar(2)+1;
numtasks2(7)=numtasks2(7)+1;
% água de alimentação auxiliar
end

if (CAAA(nn,2)>1.6)
% pressão da contenção
y(8) = 1;
numvar(4)=numvar(4)+1;
numtasks2(8)=numtasks2(8)+1;
% spray da contenção
% desligamento das BRRs
```

APÊNDICE 2

end

```
if (GV(nn,2)>92 | GV(nn,3)>92)
% nível do GV1 e do GV2
y(9) = 1;
numvar(7)=numvar(7)+1;
numvar(8)=numvar(8)+1;
numtasks2(9)=numtasks2(9)+1;
% isolamento da água de alimentação
end
```

```
if (IS==1 & (((GV(nn,6)>2030 | GV(nn,7)>2030) | ((GV(nn,6)>740 | GV(nn,7)>740) &
(PRIM(nn,3)<290 & PRIM(nn,3)<170))))))
% vazao de vapor no GV1 e GV2, temperatura media do primário e pressão da contenção
y(10) = 1;
numvar(4)=numvar(4)+1;
numvar(2)=numvar(2)+1;
numvar(10)=numvar(10)+1;
numvar(11)=numvar(11)+1;
numtasks2(10)=numtasks2(10)+1;
% isolamento das linhas de vapor principal
ILVP=1;
end
```

```
if (CAAA(nn,2)>1.05)
% pressão da contenção
y(15) = 1;
numvar(4)=numvar(4)+1;
numtasks2(15)=numtasks2(15)+1;
% isolamento das linhas de vapor principal
ILVP=1;
end
```

```
if (ILVP==1 & ((GV(nn,2)<3 & GV(nn,3)<3) | PRIM(nn,2)>164))
% nível do GV1 e do GV2, pressão do primário
Y(11) = 1;
numvar(7)=numvar(7)+1;
numvar(8)=numvar(8)+1;
numvar(1)=numvar(1)+1;
```

APÊNDICE 2

```
numtasks2(11)=numtasks2(11)+1;
% resfriamento do primário por drenagem e alimentação
end

if (GV(nn,4)<2)
% pressoes de vapor dos GVs
y(12) = 1;
numvar(5)=numvar(5)+1;
numvar(6)=numvar(6)+1;
numtasks2(12)=numtasks2(12)+1;
% isolamento do gerador de vapor falho
end

if (GV(nn,5)<2)
% pressoes de vapor dos GVs
y(20) = 1;
numvar(5)=numvar(5)+1;
numvar(6)=numvar(6)+1;
numtasks2(20)=numtasks2(20)+1;
% isolamento do gerador de vapor falho
end

if (GV(nn,6)>2035 | GV(nn,7)>2035)
% vazoes de vapor dos GVs
y(17) = 1;
numvar(10)=numvar(10)+1;
numvar(11)=numvar(11)+1;
numtasks2(17)=numtasks2(17)+1;
% isolamento do gerador de vapor falho
end

if (((GV(nn,2)<4 & GV(nn,3)<4) & (CAAA(nn,6)+CAAA(nn,7)<60) & (PRIM(nn,2)>GV(nn,4) |
PRIM(nn,2)>GV(nn,5)) & PRIM(nn,3)>170 & TRIP==1))
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, grau de subresfriamento e pressão da contenção
y(13) = 1;
numvar(7)=numvar(7)+1;
numvar(8)=numvar(8)+1;
```

APÊNDICE 2

```
numvar(15)=numvar(15)+1;
numvar(16)=numvar(16)+1;
numvar(1)=numvar(1)+1;
numvar(17)=numvar(17)+1;
numvar(2)=numvar(2)+1;
numvar(5)=numvar(5)+1;
numvar(6)=numvar(6)+1;
numtasks2(13)=numtasks2(13)+1;
% desligamento das BRRs
end

if (IS==1 & PRIM(nn,5)<20)
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, grau de subresfriamento e pressão da contenção
y(18) = 1;
numvar(17)=numvar(17)+1;
numtasks2(18)=numtasks2(18)+1;
% desligamento das BRRs
end

if (PRIM(nn,2)<95)
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, grau de subresfriamento e pressão da contenção
y(21) = 1;
numvar(1)=numvar(1)+1;
numtasks2(21)=numtasks2(21)+1;
% desligamento das BRRs
end

% treinamento supervisionado on-line

% calculo do erro na saida

% camada 2 - saída dos neurônios

for i = 1:npar
for j = 1:npartx(i)
f2(i,j) = (((x1(nn,i) - m2(i,j)) / sigma2(i,j))^2);
```

APÊNDICE 2

```
a2(i,j) = exp(-f2(i,j));  
end  
end
```

```
if (ivar~=0)  
    for j=1:npartx(i)  
        a2(ivar,j)=0;  
    end  
end
```

% camada 3 - saída dos neurônios

```
[V,k]=min(a2);
```

```
for i=1:npar  
    for j=1:npartx(i)  
        if (V==a2(i,j))  
            numin(i)=numin(i)+1;  
        end  
    end  
end
```

```
for i = 1:nregras  
    a3(i)=0;  
    for j=1:npar  
        for k=1:npartx(j)  
            a34(j,k)=0.0;  
        end  
    end  
end
```

```
for i = 1:nregras  
    % a33(i,j)=0;  
    a33(i)=1.0;  
    cont=0;  
    for j = 1:npar  
        for k=1:npartx(j)
```

APÊNDICE 2

```
    a34(j,k)=a2(j,k)*w23(i,j,k);
end
    [Y,k]=max(a34(j,:));
    if (a34(j,k)>0.00 & w23(i,j,k)~=0)
    a33(i) = a33(i) * a34(j,k) * w23(i,j,k);
    cont=cont+1;
end
end

if (cont~=0)
    a3(i)=a33(i);
end

end

% camada 4 - saída dos neurônios

for i = 1:ntasks
for k = 1:nparty(i)
f4(i,k)=0.0;
for j = 1:nregras
    if i==j
        if a3(j)==0 & ivar~=0
            for ij=1:ntasks
                if ij~=6 & (ij==16)
f4(i,k) = f4(i,k) + a3(ij)*w34(ij,k,i);
                end
            end
        else
f4(i,k) = f4(i,k) + a3(j);
        end
    end
end
end
a4(i,k) = min(1,f4(i,k));
end
end
```

APÊNDICE 2

```
% camada 5 - saída de neurônios

for i = 1:ntasks
for j = 1:nparty(i)
f5(i,j) = (((f4(i,j) - m4(i,j)) / sigma4(i,j))^2);
a5(i,j)=a4(i,j);
end
end

a5;

for j = 1:ntasks

if nparty(j)==1 & desl(nn)==1
    y1(j)=a5(j,1);
end

if norm(w34(:,j))==0
    y1(j)=0;
end

erro(j) = y(j) - y1(j);

yy2(nn,j,ncol)=y1(j);

if abs(erro(j))>0.21,errt=1;end

erro;

end

for j = 1:ntasks
if ((y1(j)<0.79) & (y(j)==1) & (desl(nn)==1))
IP=IP+1;
interv(nn)=interv(nn)+1;
numtasks(j)=numtasks(j)+1;
end
if ((y1(j)>0.79) & (y(j)==1) & (desl(nn)==1))
```

APÊNDICE 2

```
    interv1(nn)=interv1(nn)+1;
    numtasks3(j)=numtasks3(j)+1;
end
if ((y1(j)>0.79) & (y(j)==0) & (desl(nn)==1))
    IP1=IP1+1;
    numtasks1(j)=numtasks1(j)+1;
    interv2(nn)=interv2(nn)+1;
end
if (y(j)==1 & desl(nn)==1)
    namot=namot+1;
end
if (y(j)~=1 & desl(nn)==1)
    namot1=namot1+1;
end

end

end

end

% fim do teste de todas as amostras

% apresentar os resultados

'TAXA DE ACERTOS COM RELACAO AS TAREFAS QUE PRECISAM SER IDENTIFICADAS
(DEVEM EXISTIR)'
IPT=(namot-IP)*100/(namot)

'TAXA DE ACERTOS COM RELACAO AS TAREFAS QUE NAO DEVEM SER
IDENTIFICADAS (NAO DEVEM EXISTIR)'
IPT1=(namot1-IP1)*100/(namot1)

'NUMERO DE VEZES QUE TAREFAS PRECISAM SER IDENTIFICADAS, OU SEJA, ELAS
EXISTEM'
namot
'NUMERO DE VEZES QUE TAREFAS NAO PRECISAM SER IDENTIFICADAS, OU SEJA,
ELAS NAO EXISTEM'
namot1
'NUMERO TOTAL DE VEZES QUE A TAREFA CORRETA NAO E IDENTIFICADA'
```

APÊNDICE 2

IP

'NUMERO TOTAL DE VEZES QUE UMA TAREFA E INCORRETAMENTE IDENTIFICADA'

IP1

```
for j = 1:namo/1200
```

```
    nerros(j)=0;
```

```
    nerros1(j)=0;
```

```
    nerros2(j)=0;
```

```
    for k = 1:1200
```

```
        nerros(j)=nerros(j)+interv((j-1)*1200+k);
```

```
        nerros1(j)=nerros1(j)+interv1((j-1)*1200+k);
```

```
        nerros2(j)=nerros2(j)+interv2((j-1)*1200+k);
```

```
    end
```

```
end
```

```
k=0;
```

```
for j = 1:namo
```

```
    if interv1(j)>=2
```

```
        k=k+1;
```

```
    fim(k,1)=ncol;
```

```
    fim(k,2)=j;
```

```
    fim(k,3)=interv1(j);
```

```
    for i=1:npar
```

```
        fim1(k,i)=x1(k,i)*mult(i);
```

```
    end
```

```
    end
```

```
end
```

```
if (ncol==1) save testefim1 fim fim1 /ascii;end
```

```
if (ncol==2) save testefim2 fim fim1 /ascii;end
```

```
if (ncol==3) save testefim3 fim fim1 /ascii;end
```

```
if (ncol==3) save testefim4 fim fim1 /ascii;end
```

```
if (ncol==5) save testefim5 fim fim1 /ascii;end
```

```
if (ncol==6) save testefim6 fim fim1 /ascii;end
```

```
if (ncol==7) save testefim7 fim fim1 /ascii;end
```

```
if (ncol==8) save testefim8 fim fim1 /ascii;end
```

'NUMERO DE ERROS POR INTERVALO COM RELACAO A TAREFAS QUE PRECISAM SER IDENTIFICADAS (DEVEM EXISTIR)'

APÊNDICE 2

```
nerros
'NUMERO DE ACERTOS POR INTERVALO COM RELACAO A TAREFAS QUE PRECISAM
SER IDENTIFICADAS (DEVEM EXISTIR)'
nerros1
'NUMERO DE VEZES QUE UMA TAREFA ESPECIFICA NAO E IDENTIFICADA
CORRETAMENTE'
numtasks
'NUMERO DE VEZES QUE A VARIABEL E USADA NAS REGRAS ATIVADAS'
numvar
'NUMERO DE VEZES QUE A VARIABEL CONTRIBUI PARA O MINIMO NA FUNCAO LOGICA
AND NA ENTRADA DA CAMADA 3'
numin
'NUMERO DE VEZES QUE A TAREFA CORRETA E ATIVADA'
numtasks3
'NUMERO DE VEZES QUE O CONJUNTO DE REGRAS DISPARA DETERMINADA REGRA'
numtasks2
'NUMERO DE VEZES QUE UMA TAREFA ESPECIFICA E IDENTIFICADA ERRADA'
numtasks1
'NUMERO DE ERROS POR INTERVALO COM RELACAO A TAREFAS QUE NAO PRECISAM
SER IDENTIFICADAS (DEVEM EXISTIR)'
nerros2

end
end

if cogent==1

npar = 19;
ntasks = 21;
npartx = [2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2];
nparty = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];
nregras = 21;
eta = 0.1;
mm=8;
nam=28421+17373+29342+26091+21437+16059+31781+33084;
namot=0;
r=1.8;
```

APÊNDICE 2

```
alfa=0.10;
```

```
ivar=0
```

```
% leitura de vetores e matrizes de conjuntos Gaussianos e conexões entre camadas
```

```
for j = 1:npar
```

```
for k = 1:npartx(j)
```

```
for i = 1:nregras
```

```
    w23(i,j,k) = 0.0;
```

```
end
```

```
end
```

```
end
```

```
for j=1:npartx(1)
```

```
    w23(1,1,j)=0;
```

```
    w23(1,5,j)=1;
```

```
    w23(19,6,j)=1;
```

```
    w23(1,4,j)=0;
```

```
    w23(2,1,j)=0;
```

```
    w23(2,4,j)=0;
```

```
    w23(2,5,j)=0;
```

```
    w23(2,6,j)=0;
```

```
    w23(2,7,j)=0;
```

```
    w23(2,8,j)=0;
```

```
    w23(2,10,j)=0;
```

```
    w23(2,11,j)=0;
```

```
    w23(2,12,j)=0;
```

```
    w23(2,13,j)=0;
```

```
    w23(2,14,j)=1;
```

```
    w23(3,1,j)=1;
```

```
    w23(4,1,j)=1;
```

```
    w23(5,3,j)=1;
```

```
    w23(5,9,j)=1;
```

```
    w23(6,1,j)=0;
```

```
    w23(6,4,j)=0;
```

```
    w23(6,5,j)=0;
```

```
    w23(6,6,j)=0;
```

APÊNDICE 2

```
w23(6,18,j)=0;
w23(6,19,2)=1;
w23(7,2,j)=1;
w23(8,4,j)=1;
w23(9,7,2)=1;
w23(9,8,2)=1;
w23(10,10,j)=1;
w23(10,11,j)=1;
w23(10,2,j)=1;
w23(10,4,j)=0;
w23(11,1,2)=1;
w23(11,7,1)=1;
w23(11,8,1)=1;
w23(12,5,1)=1;
w23(20,6,1)=1;
w23(12,10,j)=0;
w23(12,11,j)=0;
w23(13,4,j)=0;
w23(13,15,j)=1;
w23(13,16,j)=1;
w23(13,7,1)=1;
w23(13,8,1)=1;
w23(13,5,j)=0;
w23(13,1,j)=0;
w23(13,6,j)=0;
w23(13,2,j)=1;
w23(13,17,j)=0;
w23(14,4,j)=1;
w23(15,4,j)=1;
w23(16,1,2)=1;
w23(17,10,j)=1;
w23(17,11,j)=1;
w23(18,17,j)=1;
w23(18,1,j)=0;
w23(21,1,1)=1;
end

load w34 m2 sigma2 m4 sigma4 w34;
```

APÊNDICE 2

```
m2
sigma2
m4
sigma4

% teste de identificacao das tarefas de segurancia

IP=0;
IP1=0;
namot=0;
namot1=0;
namo=12;

for i=1:namo
for j=1:npar
x1(i,j)=0;
end
end

mult = [100 200 1 0.1 10 50 10 50 30 10 10 0.2 200 10 50 50 10 1 1];

load coge.dat;

for nn=1:namo

PRIM(nn,2) = coge(1,nn);
% pressao do primario

PRIM(nn,3) = coge(2,nn);
% temperatura media do primario

CAAA(nn,4) = coge(3,nn);
% nivel do poco da contencao

CAAA(nn,2) = coge(4,nn);
% pressao da contencao

GV(nn,4) = coge(5,nn);
```

APÊNDICE 2

% pressao do GV1

GV(nn,5) = coge(6,nn);

% pressao do GV2

GV(nn,2) = coge(7,nn);

% nivel do GV1

GV(nn,3) = coge(8,nn);

% nivel do GV2

CAAA(nn,5) = coge(9,nn);

% nivel do TAR

GV(nn,6) = coge(10,nn);

% vazao de vapor no GV1

GV(nn,7) = coge(11,nn);

% vazao de vapor no GV2

PN(nn,2) = coge(12,nn);

% taxa de partida da faixa da fonte

POWER(nn,6) = coge(13,nn);

% taxa de partida da faixa intermediaria

PN(nn,6) = coge(14,nn);

% fluxo de neutrons da faixa de potencia

CAAA(nn,6) = coge(15,nn);

% vazao da agua da alimentacao auxiliar do GV1

CAAA(nn,7) = coge(16,nn);

% vazao da agua de alimentacao auxiliar do GV2

PRIM(nn,5) = coge(17,nn);

% grau de sub-resfriamento

RAD(nn,3) = coge(18,nn);

APÊNDICE 2

% nivel de radiacao do secundario A

$RAD(nn,4) = coge(19,nn);$

% nivel de radiacao do secundario B

$x1(nn,1)=PRIM(nn,2)/100;$

% pressao do primario

$x1(nn,2)=PRIM(nn,3)/200;$

% temperatura media do primario

$x1(nn,3)=CAAA(nn,4);$

% nivel do poco da contencao

$x1(nn,4)=CAAA(nn,2)/0.2;$

% pressao da contencao

$x1(nn,5)=GV(nn,4)/50;$

% pressao do GV1

$x1(nn,6)=GV(nn,5)/50;$

% pressao do GV2

$x1(nn,7)=GV(nn,2)/50;$

% nivel do GV1

$x1(nn,8)=GV(nn,3)/50;$

% nivel do GV2

$x1(nn,9)=CAAA(nn,5)/30;$

% nivel do TAR

$x1(nn,10)=GV(nn,6)/10;$

% vazao de vapor no GV1

$x1(nn,11)=GV(nn,7)/10;$

% vazao de vapor no GV2

APÊNDICE 2

```
x1(nn,12)=PN(nn,2)/0.2;
```

```
% taxa de partida da faixa da fonte
```

```
x1(nn,13)=POWER(nn,6)/200;
```

```
% taxa de partida da faixa intermediaria
```

```
x1(nn,14)=PN(nn,6)/10;
```

```
% fluxo de neutrons da faixa de potencia
```

```
x1(nn,15)=CAAA(nn,6)/50;
```

```
% vazao da agua da alimentacao auxiliar do GV1
```

```
x1(nn,16)=CAAA(nn,7)/50;
```

```
% vazao da agua de alimentacao auxiliar do GV2
```

```
x1(nn,17)=PRIM(nn,5)/10;
```

```
% grau de sub-resfriamento
```

```
x1(nn,18)=RAD(nn,3)/5;
```

```
% nivel de radiacao do secundario A
```

```
x1(nn,19)=RAD(nn,4)/5;
```

```
% nivel de radiacao do secundario B
```

```
for i=1:ntasks
```

```
numtasks(i)=0;
```

```
numtasks1(i)=0;
```

```
numtasks2(i)=0;
```

```
numtasks3(i)=0;
```

```
end
```

```
for i=1:npar
```

```
numvar(i)=0;
```

```
numin(i)=0;
```

```
end
```

```
% aqui começa o loop de treinamento de um cenario específico
```

APÊNDICE 2

```
IS=0;
ILVP=0;
TRIP=0;
RADI=0;

% teste de saida - identificação das tarefas de segurança por regras

if (RAD(nn,3)-1.95424>0 | RAD(nn,4)-1.84510>0)
    RADI=1;
end

interv(nn)=0;
interv1(nn)=0;
interv2(nn)=0;

for i=1:ntasks
y(i)=0;
y1(i)=0;
end

TRIP=1;
desl(nn)=1;

if (GV(nn,4)<42)
% pressão do GV1
y(1) = 1;
% injeção de segurança de alta pressão e ultima fonte fria
IS = 1;
numvar(5)=numvar(5)+1;
numtasks2(1)=numtasks2(1)+1;
end

if (GV(nn,5)<42)
% pressão do GV2
y(19) = 1;
```

APÊNDICE 2

```
% injeção de segurança de alta pressão e ultima fonte fria
IS = 1;
numvar(6)=numvar(6)+1;
numtasks2(19)=numtasks2(19)+1;
end

if (CAAA(nn,2)>=0.17)
% pressão da contenção
numvar(4)=numvar(4)+1;
numtasks2(14)=numtasks2(14)+1;
IS=1;
y(14)=1;
% realinhamento dos sistemas de ventilação
% injeção de segurança de alta pressão e ultima fonte fria
% isolamento da contenção, isolamento da ventilação da contenção e resfriamento da
contenção
end

if (PRIM(nn,2)<123)
% pressao do primario
y(16)=1;
IS=1;
numvar(1)=numvar(1)+1;
numtasks2(16)=numtasks2(16)+1;
% injeção de segurança de alta pressão e ultima fonte fria
% realinhamento dos sistemas de ventilação
end

if ((PN(nn,6)>5 | PN(nn,4)>0 | PN(nn,2)>0.2) & TRIP==1)
% fluxo de neutrons FP, taxa de partida F1 e taxa de partida FF
y(2) = 1;
numvar(14)=numvar(14)+1;
numvar(13)=numvar(13)+1;
numvar(12)=numvar(12)+1;
numtasks2(2)=numtasks2(2)+1;
% injeção de ácido bórico
end

if (IS==1 & PRIM(nn,2)<52)
```

APÊNDICE 2

```
% pressão do primário
y(3) = 1;
numvar(1)=numvar(1)+1;
numtasks2(3)=numtasks2(3)+1;
% injeção do acumulador
end

if (IS==1 & PRIM(nn,2)<9)
% pressão do primário
y(4) = 1;
numvar(1)=numvar(1)+1;
numtasks2(4)=numtasks2(4)+1;
% injeção de segurança de baixa pressão
end

if (IS==1 & CAAA(nn,5)<=35 & CAAA(nn,4)>=1.50)
% nível do TAR, nível do poço da contenção
y(5) = 1;
numvar(9)=numvar(9)+1;
numvar(3)=numvar(3)+1;
numtasks2(5)=numtasks2(5)+1;
% recirculação do poço da contenção
end

if (IS==1 & RADl==1)
% nível de radiação no secundario
y(6) = 1;
numvar(18)=numvar(18)+1;
numvar(19)=numvar(19)+1;
numtasks2(6)=numtasks2(6)+1;
% resfriamento rápido do primário e despressurização rápida do primário
end

if (PRIM(nn,3)>170)
% temperatura média do primário
y(7) = 1;
numvar(2)=numvar(2)+1;
numtasks2(7)=numtasks2(7)+1;
% água de alimentação auxiliar
```

APÊNDICE 2

end

if (CAAA(nn,2)>1.6)

% pressão da contenção

y(8) = 1;

numvar(4)=numvar(4)+1;

numtasks2(8)=numtasks2(8)+1;

% spray da contenção

% desligamento das BRRs

end

if (GV(nn,2)>92 | GV(nn,3)>92)

% nível do GV1 e do GV2

y(9) = 1;

numvar(7)=numvar(7)+1;

numvar(8)=numvar(8)+1;

numtasks2(9)=numtasks2(9)+1;

% isolamento da água de alimentação

end

if (IS==1 & (((GV(nn,6)>2030 | GV(nn,7)>2030) | ((GV(nn,6)>740 | GV(nn,7)>740) &
(PRIM(nn,3)<290))))))

% vazão de vapor no GV1 e GV2, temperatura média do primário e pressão da contenção

y(10) = 1;

numvar(4)=numvar(4)+1;

numvar(2)=numvar(2)+1;

numvar(10)=numvar(10)+1;

numvar(11)=numvar(11)+1;

numtasks2(10)=numtasks2(10)+1;

% isolamento das linhas de vapor principal

ILVP=1;

end

if (CAAA(nn,2)>1.05)

% pressão da contenção

y(15) = 1;

numvar(4)=numvar(4)+1;

numtasks2(15)=numtasks2(15)+1;

% isolamento das linhas de vapor principal

APÊNDICE 2

```
ILVP=1;
end

if (ILVP==1 & ((GV(nn,2)<3 & GV(nn,3)<3) | PRIM(nn,2)>164))
% nível do GV1 e do GV2, pressão do primário
Y(11) = 1;
numvar(7)=numvar(7)+1;
numvar(8)=numvar(8)+1;
numvar(1)=numvar(1)+1;
numtasks2(11)=numtasks2(11)+1;
% resfriamento do primário por drenagem e alimentação
end

if (GV(nn,4)<2)
% pressões de vapor dos GVs
y(12) = 1;
numvar(5)=numvar(5)+1;
numvar(6)=numvar(6)+1;
numtasks2(12)=numtasks2(12)+1;
% isolamento do gerador de vapor falho
end

if (GV(nn,5)<2)
% pressões de vapor dos GVs
y(20) = 1;
numvar(5)=numvar(5)+1;
numvar(6)=numvar(6)+1;
numtasks2(20)=numtasks2(20)+1;
% isolamento do gerador de vapor falho
end

if (GV(nn,6)>2035 | GV(nn,7)>2035)
% vazões de vapor dos GVs
y(17) = 1;
numvar(10)=numvar(10)+1;
numvar(11)=numvar(11)+1;
numtasks2(17)=numtasks2(17)+1;
% isolamento do gerador de vapor falho
end
```

APÊNDICE 2

```
if (((GV(nn,2)<4 & GV(nn,3)<4) & (CAAA(nn,6)+CAAA(nn,7)<60) & (PRIM(nn,2)>GV(nn,4) |
PRIM(nn,2)>GV(nn,5)) & PRIM(nn,3)>170 & TRIP==1))
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, grau de subresfriamento e pressão da contenção
y(13) = 1;
end

if (IS==1 & PRIM(nn,5)<20)
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, grau de subresfriamento e pressão da contenção
y(18) = 1;
numvar(17)=numvar(17)+1;
numtasks2(18)=numtasks2(18)+1;
% desligamento das BRRs
end

if (PRIM(nn,2)<95)
% nível dos GVs, vazao de água de alimentação auxiliar, pressão do primário, pressão dos
GVs, temperatura media do primário, grau de subresfriamento e pressão da contenção
y(21) = 1;
numvar(1)=numvar(1)+1;
numtasks2(21)=numtasks2(21)+1;
% desligamento das BRRs
end

% treinamento supervisionado on-line

% calculo do erro na saida

% camada 2 - saída dos neurônios

for i = 1:npar
for j = 1:npartx(i)
f2(i,j) = (((x1(nn,i) - m2(i,j)) / sigma2(i,j))^2);
a2(i,j) = exp(-f2(i,j));
end
```

APÊNDICE 2

```
end
```

```
if (ivar~=0)
    for j=1:npartx(i)
        a2(ivar,j)=0;
    end
end
```

```
% camada 3 - saída dos neurônios
```

```
[V,k]=min(a2);
```

```
for i=1:npar
    for j=1:npartx(i)
        if (V==a2(i,j))
            numin(i)=numin(i)+1;
        end
    end
end
```

```
for i = 1:nregras
    a3(i)=0;
    for j=1:npar
        for k=1:npartx(j)
            a34(j,k)=0.0;
        end
    end
end
```

```
for i = 1:nregras
    a33(i)=1.0;
    cont=0;
    for j = 1:npar
        for k=1:npartx(j)
            a34(j,k)=a2(j,k)*w23(i,j,k);
        end
    end
    [Y,k]=max(a34(j,:));
```

APÊNDICE 2

```
if (a34(j,k)>0.00 & w23(i,j,k)~=0)
    a33(i) = a33(i) * a34(j,k) * w23(i,j,k);
    cont=cont+1;
end
end
```

```
if (cont~=0)
    a3(i)=a33(i);
end
```

```
end
```

```
% camada 4 - saída dos neurônios
```

```
for i = 1:ntasks
    for k = 1:nparty(i)
        f4(i,k)=0.0;
        for j = 1:nregras
            if i==j
                if a3(j)==0 & ivar~=0
                    for ij=1:ntasks
                        f4(i,k) = f4(i,k) + a3(ij)*w34(ij,k,i);
                    end
                else
                    f4(i,k) = f4(i,k) + a3(j);
                end
            end
        end
        a4(i,k) = min(1,f4(i,k));
    end
end
```

```
a2;
a3;
a4;
```

APÊNDICE 2

```
% camada 5 - saída de neurônios
for i = 1:ntasks
for j = 1:nparty(i)
f5(i,j) = (((f4(i,j) - m4(i,j)) / sigma4(i,j))^2);
a5(i,j)=a4(i,j);
end
end
f4;
a5;

for j = 1:ntasks
if nparty(j)~=1

soma1(j) = 0;
soma2(j) = 0;
for i = 1:nparty(j)
soma1(j) = soma1(j) + a5(j,i) * sigma4(j,i);
soma2(j) = soma2(j) + a5(j,i) * sigma4(j,i) * m4(j,i);
end

if soma1(j)~=0
y1(j) = soma2(j)/soma1(j);
end

end

if nparty(j)==1 & desl(nn)==1
y1(j)=a5(j,1);
end

if norm(w34(:,j))==0
y1(j)=0;
end

erro(j) = y(j) - y1(j);

if abs(erro(j))>0.5,errt=1;end
```

APÊNDICE 2

```
erro;

end

for j = 1:ntasks
if ((y1(j)<0.8) & (y(j)==1) & (desl(nn)==1))
IP=IP+1;
interv(nn)=interv(nn)+1;
numtasks(j)=numtasks(j)+1;
end
if ((y1(j)>0.8) & (y(j)==1) & (desl(nn)==1))
    interv1(nn)=interv1(nn)+1;
    numtasks3(j)=numtasks3(j)+1;
end
if ((y1(j)>0.8) & (y(j)==0) & (desl(nn)==1))
    IP1=IP1+1;
    numtasks1(j)=numtasks1(j)+1;
    interv2(nn)=interv2(nn)+1;
end
if (y(j)==1 & desl(nn)==1)
    namot=namot+1;
end
if (y(j)~=1 & desl(nn)==1)
    namot1=namot1+1;
end

end

% fim do teste de todas as amostras
nn
% apresentar os resultados
'NUMERO DE VEZES QUE UMA TAREFA ESPECIFICA NAO E IDENTIFICADA
CORRETAMENTE'
numtasks
%'NUMERO DE VEZES QUE A VARIAVEL E USADA NAS REGRAS ATIVADAS'
%numvar
%'NUMERO DE VEZES QUE A VARIAVEL CONTRIBUI PARA O MINIMO NA FUNCAO
LOGICA AND NA ENTRADA DA CAMADA 3'
%numin
```

APÊNDICE 2

```
'NUMERO DE VEZES QUE A TAREFA CORRETA E ATIVADA'  
numtasks3  
'NUMERO DE VEZES QUE O CONJUNTO DE REGRAS DISPARA DETERMINADA REGRA'  
numtasks2  
'NUMERO DE VEZES QUE UMA TAREFA ESPECIFICA E IDENTIFICADA ERRADA'  
numtasks1  
y1  
y  
  
if (nn==1), save teste1 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==2), save teste2 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==3), save teste3 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==4), save teste4 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==5), save teste5 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==6), save teste6 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==7), save teste7 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==8), save teste8 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==9), save teste9 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==10), save teste10 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==11), save teste11 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
if (nn==12), save teste12 numtasks2 numtasks3 numtasks numtasks1 /ascii;end  
  
end  
  
end
```

APENDICE 3

Resultados do Programa TESTEDAREDEOUT7 para os acidentes 2 a 8

Tabela A1 – Tarefas Ativadas no Acidente 2

Tarefa	TCR	TCA	TNI	TIA
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	10131	10131	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	0	0	0	0

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A2 – Tarefas Ativadas no Acidente 3

Tarefa	TCR	TCA	TNI	TIA
1	9992	9992	0	0
2	9992	9992	0	0
3	9992	9992	0	0
4	9992	9992	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	9992	9992	0	0
10	9992	9992	0	0
11	9992	9992	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	4825	4825	0	5167
21	0	0	0	0

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A3 – Tarefas Ativadas no Acidente 4

Tarefa	TCR	TCA	TNI	TIA
1	2124	2124	0	92
2	2124	2124	0	92
3	2124	2124	0	92
4	2124	2124	0	92
5	7880	1312	6652	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	9918	9918	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	1212	0	1212	0
21	0	0	0	2616

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A4 – Tarefas Ativadas no Acidente 5

Tarefa	TCR	TCA	TNI	TIA
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	13543	13543	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	13543	13543	0	0
21	0	0	0	0

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A5 – Tarefas Ativadas no Acidente 6

Tarefa	TCR	TCA	TNI	TIA
1	28830	28830	0	0
2	28830	28830	0	0
3	28830	28830	0	0
4	28830	28830	0	0
5	1508	0	1508	0
6	0	0	0	0
7	0	0	0	0
8	1529	1529	0	1592
9	28830	28830	0	0
10	28830	28830	0	0
11	28830	28830	0	0
12	28830	28830	0	0
13	28830	28830	0	0
14	28830	28830	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	2941	1517	1444	0
21	0	0	0	0

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A6 – Tarefas Ativadas no Acidente 7

Tarefa	TCR	TCA	TNI	TIA
1	27316	27316	0	272
2	27316	27316	0	272
3	27316	27316	0	272
4	27316	27316	0	272
5	28	0	28	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	27316	27316	0	0
12	27316	27316	0	0
13	27316	27316	0	0
14	27316	27316	0	0
15	0	0	0	0
16	0	0	0	0
17	109	0	109	0
18	0	0	0	0
19	26605	26615	0	312
20	0	0	0	0
21	0	0	0	13266

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A7 – Tarefas Ativadas no Acidente 8

Tarefa	TCR	TCA	TNI	TIA
1	9586	9586	0	0
2	9586	9586	0	0
3	9586	9586	0	0
4	9586	9586	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	7557	7225	0	0
10	7557	7225	0	0
11	9586	9586	0	0
12	0	0	0	5023
13	0	0	0	5023
14	0	0	0	5023
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	4003	4003	0	5671
21	0	0	0	0

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A8 – Tarefas Ativadas no Acidente 2 com o Valor da Pressão do Sistema de Refrigeração do Reator Fixo em 109 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	10131	10131	0	0
2	10131	10131	0	0
3	10131	10131	0	0
4	10131	10131	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	9026	9026	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	0	0	0	0

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A9 – Tarefas Ativadas no Acidente 3 com o Valor da Pressão do Sistema de Refrigeração do Reator Fixo em 109 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	9992	9992	0	0
2	9992	9992	0	0
3	9992	9992	0	0
4	9992	9992	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	9992	9992	0	0
10	9992	9992	0	0
11	9992	9992	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	9992	9992	0	0

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A10 – Tarefas Ativadas no Acidente 4 com o Valor da Pressão do Sistema de Refrigeração do Reator Fixo em 109 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	9918	9918	0	0
2	9918	9918	0	0
3	9918	9918	0	0
4	9918	9918	0	0
5	7880	1228	6652	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	9918	9918	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	1212	3048	1212	0
21	0	0	0	9918

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A11 – Tarefas Ativadas no Acidente 5 com o Valor da Pressão do Sistema de Refrigeração do Reator Fixo em 109 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	13543	13543	0	0
2	13543	13543	0	0
3	13543	13543	0	0
4	13543	13543	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	13543	13543	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	13543	13543	0	0
21	0	0	0	13543

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A12 – Tarefas Ativadas no Acidente 6 com o Valor da Pressão do Sistema de Refrigeração do Reator Fixo em 109 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	13845	8161	5684	0
2	13845	8161	5684	0
3	13845	8161	5684	0
4	13845	8161	5684	0
5	1508	0	1508	0
6	0	0	0	0
7	0	0	0	0
8	1529	1529	0	4512
9	28830	28830	0	0
10	28830	28830	0	0
11	28830	28830	0	0
12	28830	28830	0	5023
13	28830	28830	0	5023
14	28830	28830	0	5023
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	2941	1497	1444	8161
20	0	0	0	0
21	0	0	0	28830

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A13 – Tarefas Ativadas no Acidente 7 com o Valor da Pressão do Sistema de Refrigeração do Reator Fixo em 109 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	27386	27386	0	0
2	27386	27386	0	0
3	27386	27386	0	0
4	27386	27386	0	0
5	28	0	28	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	27386	27386	0	0
12	27386	27386	0	0
13	27386	27386	0	0
14	27386	27386	0	0
15	0	0	0	0
16	0	0	0	0
17	109	0	109	0
18	0	0	0	0
19	26605	26605	105	0
20	0	0	0	0
21	0	0	0	27386

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A14 – Tarefas Ativadas no Acidente 8 com o Valor da Pressão do Sistema de Refrigeração do Reator Fixo em 109 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	9586	9586	0	0
2	9586	9586	0	0
3	9586	9586	0	0
4	9586	9586	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	9586	9586	0	0
12	7557	7557	0	672
13	7557	7557	0	672
14	7557	7557	0	672
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	0	0	0	9586

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A15 – Tarefas Ativadas no Acidente 2 com o Valor da Pressão do Gerador de Vapor Fixo em 22 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	10131	10131	0	0
2	10131	10131	0	0
3	10131	10131	0	0
4	10131	10131	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	10131	10131	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	10131
20	0	0	0	0
21	0	0	0	0

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A15 – Tarefas Ativadas no Acidente 3 com o Valor da Pressão do Gerador de Vapor Fixo em 22 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	9992	9992	0	0
2	9992	9992	0	0
3	9992	9992	0	0
4	9992	9992	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	9992	9992	0	0
10	9992	9992	0	0
11	9992	9992	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	9992
20	0	0	0	0
21	4825	4825	0	5167

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A15 – Tarefas Ativadas no Acidente 4 com o Valor da Pressão do Gerador de Vapor Fixo em 22 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	9918	9918	148	0
2	9918	9918	0	0
3	9918	9918	0	0
4	9918	9918	0	0
5	7880	1228	6652	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	9918	9918	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	9918
20	1212	0	1212	0
21	0	0	0	2626

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A15 – Tarefas Ativadas no Acidente 5 com o Valor da Pressão do Gerador de Vapor Fixo em 22 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	13543	13543	0	0
2	13543	13543	0	0
3	13543	13543	0	0
4	13543	13543	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	13543	13543	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	13543
20	13543	13543	0	0
21	0	0	0	0

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A15 – Tarefas Ativadas no Acidente 6 com o Valor da Pressão do Gerador de Vapor Fixo em 22 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	28830	28830	0	0
2	28830	28830	0	0
3	28830	28830	0	0
4	28830	28830	0	0
5	1508	0	1508	0
6	0	0	0	0
7	0	0	0	0
8	1529	1529	0	4512
9	28830	28830	0	0
10	28830	28830	0	0
11	28830	28830	0	0
12	28830	28830	0	0
13	28830	28830	0	0
14	28830	28830	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	28830
20	2941	1497	1544	0
21	0	0	0	28830

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A15 – Tarefas Ativadas no Acidente 7 com o Valor da Pressão do Gerador de Vapor Fixo em 22 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	27386	27386	0	0
2	27386	27386	0	0
3	27386	27386	0	0
4	27386	27386	0	0
5	0	0	28	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	27386	27386	0	0
12	27386	27386	0	0
13	27386	27386	0	0
14	27386	27386	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	109	0
18	0	0	0	0
19	105	0	105	0
20	0	0	0	0
21	0	0	0	13118

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.

APENDICE 3

Tabela A15 – Tarefas Ativadas no Acidente 8 com o Valor da Pressão do Gerador de Vapor Fixo em 22 kgf/cm²

Tarefa	TCR	TCA	TNI	TIA
1	9586	9586	0	0
2	9586	9586	0	0
3	9586	9586	0	0
4	9586	9586	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	9586	9586	0	0
12	7557	7557	0	672
13	7557	7557	0	672
14	7557	7557	0	672
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	9586
20	0	0	0	0
21	4003	4003	0	5583

TCR - Número de vezes que a tarefa é ativada pelo conjunto de regras.

TCA - Número de vezes que a tarefa é corretamente ativada pela rede.

TNI - Número de vezes que a tarefa não é ativada pela rede embora tenha sido ativada pela base de regras.

TIA - Número de vezes que a tarefa é ativada pela rede, não tendo sido ativada pela base de regras.