

3 Inteligência Computacional

3.1 Introdução

A Inteligência Computacional envolve um conjunto de técnicas inspiradas na Natureza que visam o desenvolvimento de sistemas inteligentes. Estas técnicas imitam características do comportamento humano, tais como: aprendizado, raciocínio, evolução e adaptação. Entre elas estão as Redes Neurais, os Algoritmos Genéticos e a Lógica Nebulosa.

Os Sistemas Inteligentes têm aplicação nos mais diversos setores, incluindo: energético, econômico/comercial, seguros, telecomunicações, mercado de capitais, industrial, meio-ambiente e medicina, obtendo, em muitos casos, uma eficiência superior àquela obtida por métodos convencionais.

Este capítulo apresenta uma breve descrição das técnicas de Inteligência Computacional e a aplicação da técnica de redes neurais na avaliação do efeito de cargas concentradas em vigas de aço. Como mencionado nos capítulos anteriores, é grande a dificuldade de se obter uma fórmula de projeto precisa para este problema, devido à influência de diversos parâmetros. Adicionalmente a quantidade insuficiente de dados experimentais dificulta a validação de uma análise paramétrica completa e uma correta avaliação dos comportamentos resultantes.

Trabalhos anteriores (Fonseca et al., 1999), (Fonseca et al., 1999a), (Fonseca, 1999) mostraram a viabilidade de geração de novos dados a partir do treinamento das redes neurais de previsão com os dados experimentais existentes e a comparação com as fórmulas de projeto propostas por Roberts (2.9) e Bergfelt (2.7) conduziu a resultados significativos. Estes resultados também são apresentados neste capítulo.

3.2 Algoritmos Genéticos

Os Algoritmos Genéticos foram inspirados no princípio Darwiniano da evolução das espécies (sobrevivência dos mais aptos e mutações) e na genética.

São algoritmos probabilísticos, que fornecem um mecanismo de busca paralela e adaptativa. Este mecanismo é obtido a partir de uma população de indivíduos (várias soluções possíveis para o problema), representados por cromossomos (palavras binárias, vetores, matrizes etc), cada um associado a uma aptidão (avaliação da solução no problema), os quais são submetidos a um processo de evolução (seleção, reprodução, cruzamento e mutação) por vários ciclos.

O funcionamento do ciclo do algoritmo pode ser visto na Figura 3.1, onde deseja-se achar o valor máximo para a função $f(x) = x^2$, sendo x um número inteiro no intervalo de 0 a 63. A partir da população original, é calculada a aptidão dos pais que corresponde à probabilidade de procriação para gerar a próxima população, através de cruzamento e mutação. Com a nova população o ciclo se repete.

Os Algoritmos Genéticos consistem em uma técnica na qual os cromossomos são palavras binárias que codificam soluções para um problema. Algoritmos Genéticos que empregam outras estruturas e operadores heurísticos são também conhecidos como Programas Evolucionários.

Programação Genética é uma técnica automática de programação que propicia a evolução de programas de computadores que (aproximadamente) resolvem problemas.

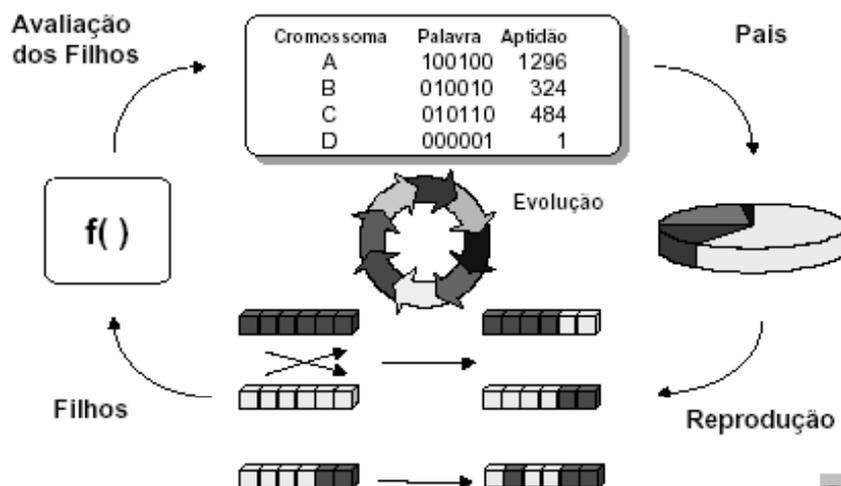


Figura 3.1 – Ciclo do algoritmo.

A técnica de algoritmos genéticos é muito útil em problemas de otimização, em casos onde se busca a síntese de um objeto (programa de computador, circuito

eletrônico, etc.), e quando a finalidade é simular um modelo que reproduza o comportamento de determinado fenômeno (*machine learning*). Nestes casos, geralmente é possível encontrar na programação tradicional um algoritmo que ofereça uma solução ótima ou aproximadamente ótima. No entanto, alguns desses algoritmos requerem informação auxiliar como, por exemplo, derivadas no caso de técnicas de gradiente, informação esta muitas vezes não disponível ou difícil de se obter. A Computação Evolucionária dispensa informações auxiliares e oferece algoritmos gerais (Algoritmos Genéticos, Programação Genética e Eletrônica Evolucionária) que são aplicados em problemas complexos, com grandes espaços de busca, de difícil modelagem, ou para os quais não há um algoritmo eficiente disponível. Alguns exemplos de utilização desta técnica são:

- Alocação de salas de aula;
- Otimização de percurso;
- Otimização de ligações de aço;
- Planejamento de tabelas de jogos;
- Controle de fluxo de caixa.

Na engenharia estrutural encontram-se alguns trabalhos que utilizam os algoritmos genéticos como ferramenta computacional. Entre eles pode-se citar os trabalhos de dimensionamento e configuração de estruturas (Lemonge & Barbosa, 2000), de análise paramétrica de materiais viscoelásticos (Barbosa et al., 2000), de projeto otimizado de galpões industriais (Ramasamy & Rajasekaran, 1996), etc.

3.3 Redes Neurais

As Redes Neurais foram inspiradas na estrutura do cérebro humano. Sabe-se que o cérebro é composto de um grande conjunto de neurônios altamente interconectados pelas sinapses. De forma análoga, as Redes Neurais Artificiais são compostas de elementos processadores com comportamento inspirado em um modelo simplificado do neurônio biológico, apresentando, deste modo, um número surpreendente de características humanas (Haykin, 1999), tais como:

- Aprendizado por experiência e apresentação de exemplos;
- Generalização de novos exemplos a partir dos já apresentados;

- Abstração, ou seja, podem extrair características essenciais de dados ruidosos;
- Associação entre padrões diferentes (Ex.: Pessoa – Local.).

O neurônio artificial é baseado no modelo do neurônio biológico (Wasserman, 1989). Os neurônios biológicos (Figura 3.2) são células com a capacidade de receber, processar e transmitir impulsos eletroquímicos através das sinapses que formam o sistema de comunicação cerebral. Cada neurônio recebe sinais de outros neurônios através de seus dendritos. Estes sinais de excitação ou inibição são acumulados no neurônio e quando a excitação atinge determinado limite, o neurônio é ativado e dispara sinais para os neurônios vizinhos através do axônio.

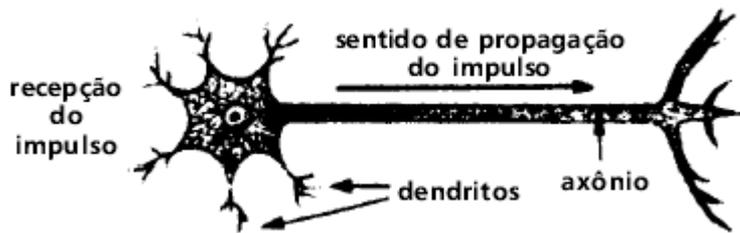


Figura 3.2 – Neurônio Biológico (Schalch., 2003).

Na rede neural artificial, o Elemento Processador j (Figura 3.3) recebe um vetor de entrada onde cada componente x_i é multiplicado por um peso sináptico w_{ij} . O nível de ativação do neurônio é uma função de ativação F (geralmente não-linear) da soma destas entradas ponderadas. Esta função gera então uma saída s_j (normalmente entre 0 e 1 ou -1 e 1) que é transmitida ao próximo neurônio.

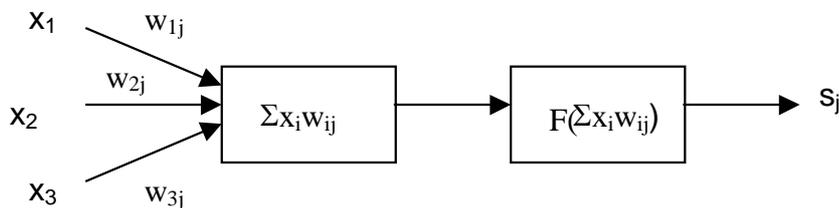


Figura 3.3 – Estrutura do Elemento Processador j .

As redes neurais têm uma estrutura organizada em camadas de elementos processadores conectadas entre si. Esta estrutura, na maioria das redes, é composta por uma camada de entrada, onde os vetores de entrada são fornecidos;

uma camada de saída, por onde os resultados da rede são apresentados; e uma ou mais camadas intermediárias ou escondidas. A organização interna das redes depende do tipo de algoritmo adotado.

As Redes Neurais aprendem a partir da apresentação de exemplos. Este aprendizado corresponde à determinação dos valores dos pesos sinápticos e pode ocorrer através de um treinamento supervisionado ou não-supervisionado. Um treinamento supervisionado implica na apresentação de vários vetores de entrada com componentes x_i e dos vetores de saída correspondentes desejados t_i até que o erro na saída seja inferior ao desejado. O treinamento não-supervisionado não requer este vetor de saída. O sistema extrai as características do conjunto de padrões, agrupando-os em classes. Desta forma, o treinamento não-supervisionado se aplica apenas a problemas de “clustering” (agrupamentos), enquanto o treinamento supervisionado é mais genérico, podendo ser empregado em diversos tipos de problema.

Devido a sua inspiração na funcionalidade e estrutura do cérebro, as Redes Neurais não se aplicam a problemas simples de cálculo, passíveis de resolução por qualquer programa convencional. Por suas características, elas são empregadas na resolução de problemas complexos, como previsão, aproximação de funções, reconhecimento de padrões e geração de novos resultados a partir de exemplos ruidosos. Alguns exemplos de utilização desta técnica são:

- Previsão de séries temporais (carga elétrica, precipitação pluviométrica, taxa de câmbio, etc.);
- Detecção e diagnóstico de falhas;
- Reconhecimento de imagens;
- Reconhecimento de voz;
- Reconhecimento de caracteres impressos (OCR);
- Detecção de fraude em cartões de crédito;
- Avaliação de risco;
- Determinação do perfil de consumidores.

Na engenharia estrutural já existem várias referências de trabalhos que utilizam Redes Neurais. Uma das pesquisas, desenvolvida por Bento et al., é a classificação e previsão de problemas na análise sísmica e resistência de estruturas

a terremotos (Bento & Ndumo,1996), (Bento et al., 1997). O mesmo pesquisador apresentou um estudo de viabilidade econômica de edifícios usando redes (Bento et al., 1996).

Outro trabalho interessante foi desenvolvido por Anderson et al. (1993) para avaliar o comportamento de ligações semi-rígidas viga-coluna no eixo de menor inércia. Uma série de ensaios foi executada, variando sistematicamente parâmetros significativos, e os resultados foram usados para treinar uma rede neural para a previsão do momento e da rotação transmitidos. Foram considerados aceitáveis erros de $\pm 10\%$ para o momento e $\pm 20\%$ para rotação. Três dados de um total de vinte e um ultrapassaram a tolerância de rotação e quatro em vinte e um ultrapassaram a tolerância de momento, equivalendo a 14,3% e 19,0% dos dados, respectivamente.

Outros exemplos da ampla utilização de redes neurais em problemas de engenharia estrutural podem ser encontrados em trabalhos de diversas linhas de pesquisa. Alguns exemplos são: identificação das cargas atuantes em asas de aeronaves através da relação carga-deformação (Cao et al., 1998); análise de vigas de concreto reforçadas (Flood et al., 2001); projeto otimizado de galpões industriais (Ramamamy & Rajasekaran, 1996); análise de uma estrutura espacial tri-dimensional deformada (Kerk & Yee, 2000); controle adaptativo descentralizado de um seguimento de ponte estaiada (Xu et al., 2001); entre outros.

3.3.1

A utilização das Redes Neurais para a Previsão da Carga Crítica

Na dissertação de mestrado (Fonseca, 1999) verificou-se que as redes neurais fornecem bons resultados na previsão da carga crítica, apesar de nunca terem sido empregadas para a avaliação dos três fenômenos separadamente. Nos trabalhos anteriores, as redes foram treinadas a partir dos dados experimentais existentes na literatura (Anexo A). Cada dado de treinamento foi composto pelos parâmetros geométricos apresentados na Figura 1.5, pelas tensões limite de escoamento da mesa σ_y^{fl} e alma σ_y^w e pela carga de ruína P_{ex} . Sabendo-se que a espessura da alma ao quadrado influencia diretamente o resultado, um novo parâmetro de entrada foi inserido (t_w^2). Apesar da rede ser capaz de estabelecer

esta relação apenas pelo fornecimento de t_w , o tempo e o número de dados necessários ao treinamento tendem a diminuir. Após alguns estudos, novos parâmetros de entrada também foram gerados a partir das fórmulas de Bergfelt (2.7) e de Roberts (2.9). Estes cinco novos parâmetros são apresentados na Tabela 3.1:

Tabela 3.1 – Parâmetros Combinados Utilizados no Treinamento.

a/h	t_w/h	t_f/t_w	c/h	$c t_w/a h$
-------	---------	-----------	-------	-------------

Foi utilizado o Software NeuralWorks Predict (Neuralware, 1997) para o treinamento e a comparação das diversas redes. Este programa permite a utilização de planilhas do Microsoft Excel para a entrada e saída de dados da rede e pode fazer uma seleção das variáveis de entrada. O nível de seleção, a ser definido pelo usuário, permite desde a utilização de todas as variáveis, até uma escolha rigorosa de um número reduzido de variáveis mais importantes.

Conforme explicado anteriormente, os tipos de função de ativação utilizados pelos elementos processadores fornecem saídas limitadas, geralmente entre 0 e 1. Deste modo, foi necessário efetuar uma normalização dos dados para permitir que os valores apresentados na entrada e saída da rede neural também se situem entre 0 e 1. De forma a especializar a rede neural com o objetivo de se obter um melhor desempenho, decidiu-se dividir os dados em 3 faixas. Estas 3 faixas permitiram uma melhor normalização durante o treinamento das redes e o modelo com três redes de previsão e uma de classificação é apresentado na Figura 3.4.

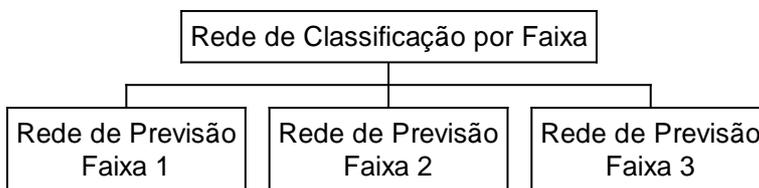


Figura 3.4 – Estrutura do Modelo adotado.

Os dados foram então divididos por faixa de carga. Estes conjuntos de dados serviram para o treinamento das três Redes de Previsão de Carga e da Rede de Classificação. Esta divisão, apesar de necessária, gerava uma quantidade muito pequena de dados por grupo de treinamento, como pode ser visto na Figura 3.5,

dificultando o aprendizado correto da influência de cada parâmetro de entrada na carga última das vigas. A solução foi adotar uma superposição de valores de treinamento nas redes de previsão. Esta superposição aumentou a quantidade de dados por faixa, viabilizando o aprendizado das redes (Tabela 3.2).

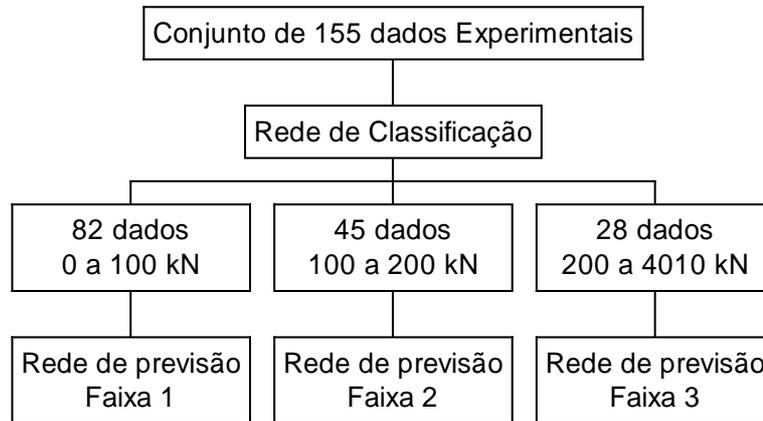


Figura 3.5– Modelo de Classificação e Previsão

Tabela 3.2 – Quantidade de Dados de Treinamento e Teste.

Faixas de Carga	Dados para Treinamento e Teste	Faixas de Carga com Superposição	Dados para Treinamento e Teste
0 a 100 kN	82	0 a 121,64 kN	89
100 a 200 kN	45	79,68 a 252,75kN	65
Maior que 200 kN	28	Maior que 150 kN	46

Foram descartados os dados que apresentavam espessura da alma menor que 1 milímetro, considerando as dificuldades de ensaio de vigas com esta característica e tendo em vista a complexidade do fenômeno medido experimentalmente. Assim, estes dados foram retirados do conjunto de treinamento e teste.

A razão dos resultados de treinamento e teste da rede da faixa 1 pelos resultados experimentais em comparação com a razão dos resultados da fórmula de Roberts (2.9) pelos resultados experimentais pode ser vista na Figura 3.6. A Figura 3.7 e a Figura 3.8 apresentam o mesmo tipo de comparação para as faixas 2 e 3, respectivamente. Cada rede forneceu os resultados para os dados classificados como sendo pertencentes à respectiva faixa.

Um resumo das características das 4 redes utilizadas na modelagem é apresentado na Tabela 3.3.

Pode-se verificar que todas as redes apresentaram erros inferiores a 15%, não tendo ultrapassado 5% para a grande maioria dos dados. A tolerância em torno de 20% permitiu uma melhor generalização, ajustando a função de forma a responder bem para todos os dados.

Apesar dos bons resultados encontrados nesta abordagem, a divisão das faixas foi baseada na carga última e não no comportamento estrutural. Desta forma, torna-se necessária uma nova abordagem, baseada na especialização das três redes para cada fenômeno específico (enrugamento, flambagem e plastificação). Isto tende a melhorar a previsão da carga última e a compreensão do problema, pois vigas com um mesmo valor de carga última podem representar fenômenos físicos diferentes.

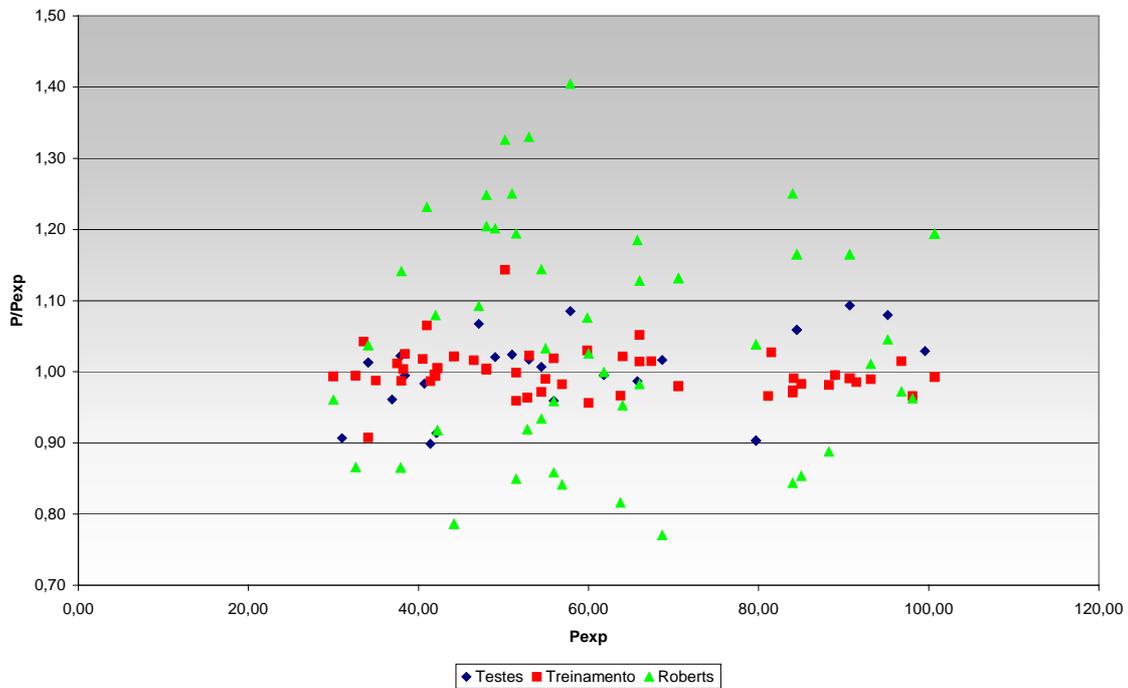


Figura 3.6 – Razão dos Resultados Previstos pela Rede da Faixa 1 e Pela equação (2.9) sobre o Resultado Experimental em função da Carga Experimental.

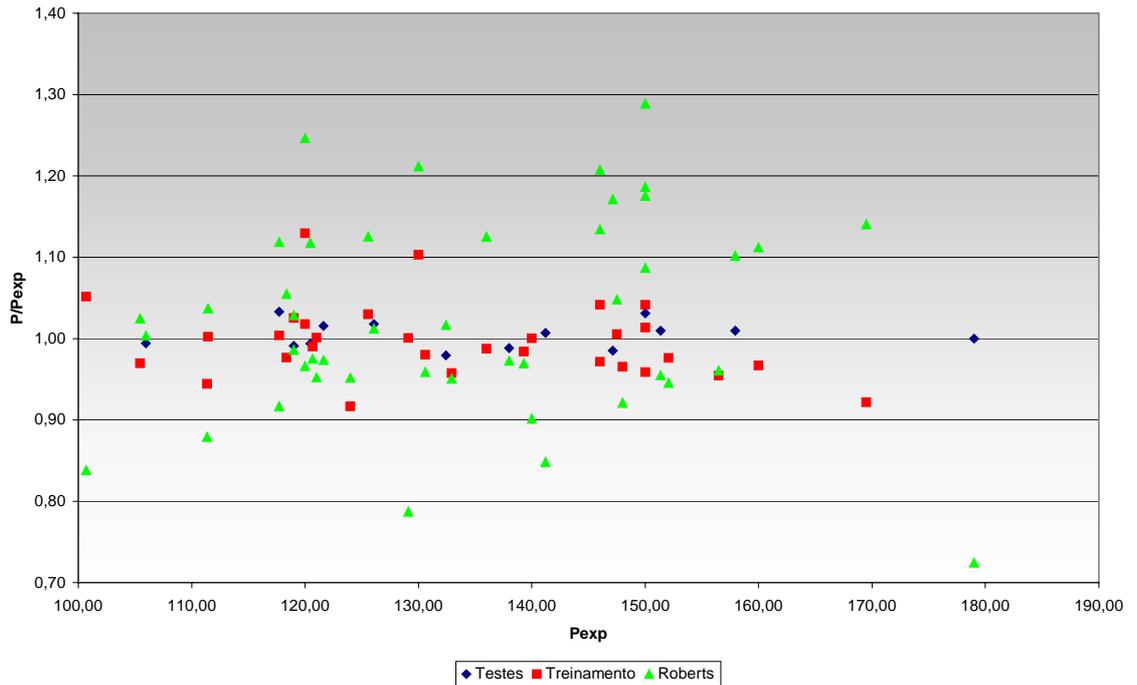


Figura 3.7 – Razão dos Resultados Previstos pela Rede da Faixa 2 e Pela equação (2.9) sobre o Resultado Experimental em função da Carga Experimental.

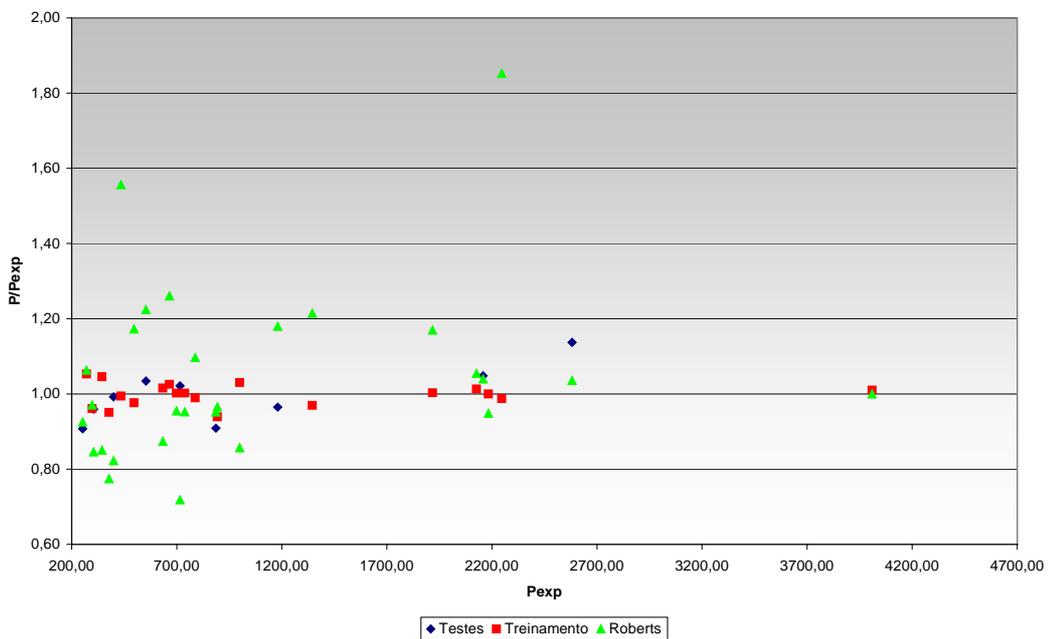


Figura 3.8 – Razão dos Resultados Previstos pela Rede da Faixa 3 e Pela equação (2.9) sobre o Resultado Experimental em função da Carga Experimental.

Tabela 3.3 – Características das Redes Selecionadas.

Rede Neural	Classificação	Faixa 1	Faixa 2	Faixa 3
Entradas	14	15	14	11
Processad. na Camada Escond.	0	16	16	8
Saídas	3	1	1	1
Tolerância (%)	5	20	20	17
Função de Ativação	Sigmóide	Sigmóide	Sigmóide	Sigmóide
Nível de Ruído	Dados limpos	Dados limpos	Dados limpos	Dados limpos
Transformação de Dados	Superficial	Superficial	Superficial	Superficial
Seleção de Variáveis	Nenhuma	Nenhuma	Nenhuma	Nenhuma
Busca da Rede	Exaustiva	Exaustiva	Exaustiva	Exaustiva
Erro percentual máximo	-	14,28	12,96	11,16
Desvio padrão	-	5,20	3,48	4,47
Erro Percentual médio absoluto	-	4,08	2,75	4,68
Dados com erro inferior a 5%	100%	69,6%	86,7%	60,7%
Dados com erro entre 5% e 10%	-	24,1%	8,9%	32,1%
Dados com erro entre 10% e 15%	-	6,3%	4,4%	7,2%
Percentual de Treinamento-Testes	70-30	70-30	70-30	70-30

3.4 Lógica Nebulosa

De acordo com Zadeh (1965), o princípio da incompatibilidade diz que: “Conforme a complexidade de um sistema aumenta, a nossa habilidade de fazer declarações precisas e significativas sobre o comportamento do sistema diminui, até alcançar um limite além do qual precisão e relevância se tornam características mutuamente exclusivas”. A lógica fuzzy fornece um método para reduzir e explicar a complexidade do sistema.

A Lógica Nebulosa tem como objetivo modelar o modo aproximado de raciocínio, tentando imitar a habilidade humana de tomar decisões racionais em um ambiente de incerteza e imprecisão. Deste modo, a Lógica Nebulosa é uma técnica inteligente que fornece um mecanismo para manipular informações imprecisas, como conceitos de esbeltez, compacidade, flexibilidade e rigidez.

Esta técnica permite inferir uma resposta aproximada para uma questão baseada em um conhecimento que é inexato, incompleto ou não totalmente confiável. Devido as suas características intrínsecas, a Lógica Nebulosa é capaz de incorporar tanto o conhecimento objetivo (a partir de dados numéricos) quanto o

conhecimento subjetivo (a partir de informações lingüísticas). Ao contrário de outras técnicas, esta se desenvolveu primeiro no mercado comercial para depois retornar às universidades.

Os primeiros fundamentos desta técnica surgiram em 1965, desenvolvidos por Lofti Zadeh. A idéia é de que os elementos pertençam a cada conjunto com diferentes graus de pertinência (ou de ativação, no caso de não estarem normalizados entre 0 e 1). Desta forma, nos conjuntos nebulosos, cada elemento pode estar associado a um ou mais conjuntos através de funções de pertinência a estes conjuntos. Estas funções representam matematicamente regras nebulosas (por exemplo, se a esbeltez da alma é baixa, então a alma é compacta). A Figura 3.9 mostra que vigas com esbeltez de alma entre 60 e 100 podem ser consideradas ao mesmo tempo de alma esbelta e de alma compacta, tendo no entanto uma pertinência maior ou menor a cada um destes conjuntos.

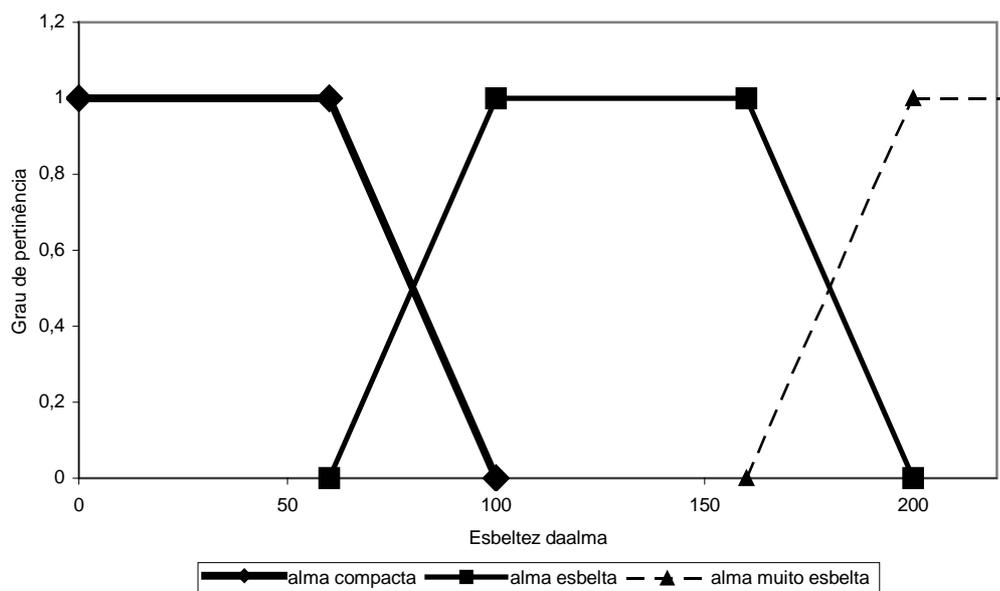


Figura 3.9 – Exemplo de conjuntos nebulosos e funções de pertinência.

Desta forma, a lógica nebulosa permite limites mais graduais para a classificação de vigas de almas esbeltas ou compactas ou a classificação de diferentes comportamentos. Exemplos de utilização desta técnica podem ser encontrados nas seguintes áreas:

- Controle de aeronaves, metrô, etc;
- Ajuste de imagens de TV;
- Ajuste de foco;

- Controles de temperatura, velocidade, posição, etc.

Na área estrutural, encontram-se os trabalhos de dano em estruturas mistas (Ramu & Johnson, 1995) e um modelo neuro-fuzzy para avaliação de construções (Barai & Nair, 2001), entre outros.

3.5 Modelos Neuro-fuzzy Hierárquicos

Os sistemas Neuro-Fuzzy estão entre os sistemas híbridos mais pesquisados na atualidade, por associarem as vantagens de duas técnicas de modelagem muito populares como as Redes Neurais e a Lógica Fuzzy. Estes sistemas combinam a capacidade de aprendizado das Redes Neurais com o poder de interpretação lingüístico dos sistemas de inferência fuzzy (Souza, 1987).

Os sistemas neuro-fuzzy tradicionais apresentam duas limitações: a primeira limitação ocorre em função da chamada explosão combinatorial das regras nebulosas. Por exemplo, num problema com oito variáveis de entrada e cada uma delas com seu universo de discurso subdividido em três conjuntos fuzzy (Ex.: alto, baixo e médio), pode-se chegar a um total de 6561 (3^8) regras. Suponha agora que se tenha 12 entradas. Usando-se a mesma divisão nos universos de discurso para cada variável de entrada, chega-se a um total de 531.441 (3^{12}) regras.

A segunda limitação presente nos sistemas neuro-fuzzy atuais é que alguns têm estrutura fixa, arbitrada a priori e invariável. Outros sistemas neuro-fuzzy, mais flexíveis, têm alguma capacidade de alterar a sua estrutura, permitindo que se altere o número de divisões no universo de discurso de algumas de suas variáveis de entrada e, conseqüentemente, o número de regras. Entretanto, esta habilidade é reduzida (Gonçalves, 2001).

Em função das limitações dos modelos neuro-fuzzy existentes, nesta tese optou-se por utilizar modelos neuro-fuzzy **hierárquicos**. Estes modelos possuem capacidade ilimitada de criar e expandir sua estrutura, reduzem a limitação quanto ao número de entradas, e são capazes de extrair regras de conhecimento a partir de um conjunto de dados. Sendo assim, o modelo adotado é o software desenvolvido no Laboratório do ICA da PUC-Rio, como dissertação de Mestrado do aluno de Sistemas de Computação Laércio Brito Gonçalves (2001). Este modelo está preparado para fazer uma classificação neuro-fuzzy, explicitando regras

hierárquicas (BSP – *Binary Space Partitioning*) geradas e o grau de ativação de cada dado aos diferentes fenômenos físicos, utilizando um método de particionamento recursivo que divide o espaço de entrada das variáveis x_n do sistema, sucessivamente, em duas regiões. Essa divisão recursiva do espaço de entrada pode ser representada por uma árvore binária (Figura 3.10) que ilustra as sucessivas sub-divisões do espaço de entrada, sendo cada uma das terminações finais uma regra do tipo: “Se x_{12} é baixo, x_{13} é baixo e x_3 é baixo, então o padrão pertence a classe 1”. A taxa de decomposição do programa é a variável que limita o número de partições e, conseqüentemente, o número de regras. O princípio da tarefa de classificação de padrões é descobrir relacionamentos entre os dados com a intenção de prever a classe de um padrão desconhecido.

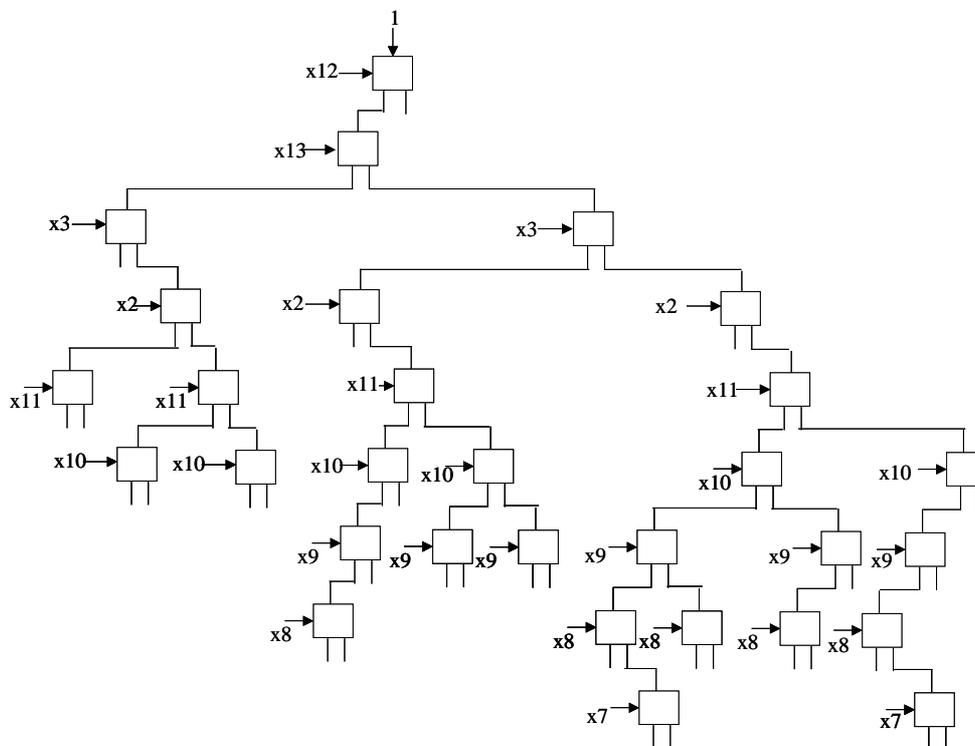


Figura 3.10 – Exemplo de estrutura em árvore do modelo NFHB-Invertido.

O programa aceita até 21 parâmetros como entrada e 7 classes diferentes como saída, atendendo às necessidades do problema estudado. Os resultados encontrados por este programa e apresentados em (Gonçalves, 2001) mostraram-se, na maioria dos casos, iguais ou superiores aos melhores resultados encontrados pelos outros modelos e algoritmos aos quais foram comparados. O desempenho do programa em relação ao tempo de processamento também se mostrou muito

bom, além da utilização deste programa ser simples e não exigir do usuário conhecimentos de programação.

O próximo capítulo apresenta a utilização de algoritmos genéticos na otimização de fórmulas de previsão da carga crítica e um sistema de previsão da carga e classificação dos dados de cargas concentradas em vigas de aço de acordo com o fenômeno físico, baseado em um modelo neuro-fuzzy hierárquico de classificação.

